TIBCO Spotfire® Server and Environment - Installation and Administration

Software Release 10.10 LTS (10.10.0)

TIBCO®

Contents

Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

ANY SOFTWARE ITEM IDENTIFIED AS THIRD PARTY LIBRARY IS AVAILABLE UNDER SEPARATE SOFTWARE LICENSE TERMS AND IS NOT PART OF A TIBCO PRODUCT. AS SUCH, THESE SOFTWARE ITEMS ARE NOT COVERED BY THE TERMS OF YOUR AGREEMENT WITH TIBCO, INCLUDING ANY TERMS CONCERNING SUPPORT, MAINTENANCE, WARRANTIES, AND INDEMNITIES. DOWNLOAD AND USE OF THESE ITEMS IS SOLELY AT YOUR OWN DISCRETION AND SUBJECT TO THE LICENSE TERMS APPLICABLE TO THEM. BY PROCEEDING TO DOWNLOAD, INSTALL OR USE ANY OF THESE ITEMS, YOU ACKNOWLEDGE THE FOREGOING DISTINCTIONS BETWEEN THESE ITEMS AND TIBCO PRODUCTS.

TIBCO, the TIBCO logo, the TIBCO O logo, TIBCO Spotfire, TIBCO Spotfire Analyst, TIBCO Spotfire Automation Services, TIBCO Spotfire Server, TIBCO Spotfire Web Player, TIBCO Spotfire Developer, TIBCO Enterprise Message Service, TIBCO Enterprise Runtime for R, TIBCO Enterprise Runtime for R - Server Edition, TERR, TERR Server Edition, TIBCO Hawk, and TIBCO Spotfire Statistics Services are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. Please see the readme.txt file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (https://www.tibco.com/patents) for details.

TIBCO Documentation and Support Services

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit https://docs.tibco.com.

TIBCO Spotfire Documentation

Documentation for Spotfire Server and related products is available on the Spotfire Server Product Documentation page.

The following documents relevant for this product can be found on the Spotfire Server Documentation site:

- *TIBCO Spotfire® Server and Environment - Quick Start*
- *TIBCO Spotfire® Server and Environment - Installation and Administration*
- *TIBCO Spotfire® Server and Environment Security*
- *TIBCO Spotfire® Server Release Notes*
- *TIBCO Spotfire® Business Author and TIBCO Spotfire® Consumer Release Notes*
- *TIBCO Spotfire® Business Author and Consumer User's Guide*
- *TIBCO Spotfire® Cobranding*
- *TIBCO Spotfire® Qualification Installation and Configuration Manual*
- *TIBCO Spotfire® Qualification User's Guide*
- *Deploying and Using a TIBCO Spotfire® Language Pack*
- *TIBCO Spotfire® Automation Services User's Guide*
- *TIBCO Drivers® - Connecting to an ODBC Data Source Using Spotfire® Analyst*
- *TIBCO Spotfire® Automation Services API Reference*
- *TIBCO Spotfire® Automation Services REST API Reference*
- *TIBCO Spotfire® Server Information Services API Reference*
- *TIBCO Spotfire® Server Library REST API Reference*
- *TIBCO Spotfire® Server Platform API Reference*
- *TIBCO Spotfire® Server Web Services API Reference*
- *TIBCO Spotfire® Server License Agreement*

Release Version Support

Some release versions of TIBCO Spotfire products are designated as long-term support (LTS) versions. LTS versions are typically supported for up to 36 months from release. Defect corrections will typically be delivered in a new release version and as hotfixes or service packs to one or more LTS versions. See also https://docs.tibco.com/pub/spotfire/general/LTS/spotfire_LTS_releases.htm.

How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, visit http://www.tibco.com/services/support.
- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support portal at https://support.tibco.com.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to https://support.tibco.com. If you do not have a user name, you can request one by clicking Register on the website.

System Requirements for Spotfire Products

For information about the system requirements for Spotfire products, visit http://spotfi.re/sr.

How to join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the TIBCO Ideas Portal. For a free registration, go to https://community.tibco.com.

For quick access to TIBCO Spotfire content, see https://community.tibco.com/products/spotfire.

Getting started

New TIBCO Spotfire® administrators can begin by learning how a Spotfire® implementation is put together and how it works, and then use the *Quick Start Guide* to begin setting up a pre-production environment. For experienced Spotfire administrators, the Release Notes describe new features and other changes.

Any updates to this documentation will be available at https://docs.tibco.com. To view the latest version of this documentation, click the question mark in the upper-right corner of a Spotfire screen (if your implementation allows access to the internet), or go to https://docs.tibco.com/products/tibco-spotfire-server.

Experienced Spotfire administrators:

- To get started, see Upgrading Spotfire.

New Spotfire administrators:

- The *Quick Start Guide* provides a short introduction to the Spotfire environment, and then takes you through the required steps for a simple Spotfire configuration: TIBCO Spotfire® Server on one computer, the TIBCO Spotfire® Analyst client on another, the node manager installed, the TIBCO Spotfire® Web Player, TIBCO Spotfire® Automation Services (if purchased), and the TIBCO® Enterprise Runtime for R - Server Edition (TERR service) or Spotfire Service for Python (if purchased) available on all network computers, and user authentication through the Spotfire database.

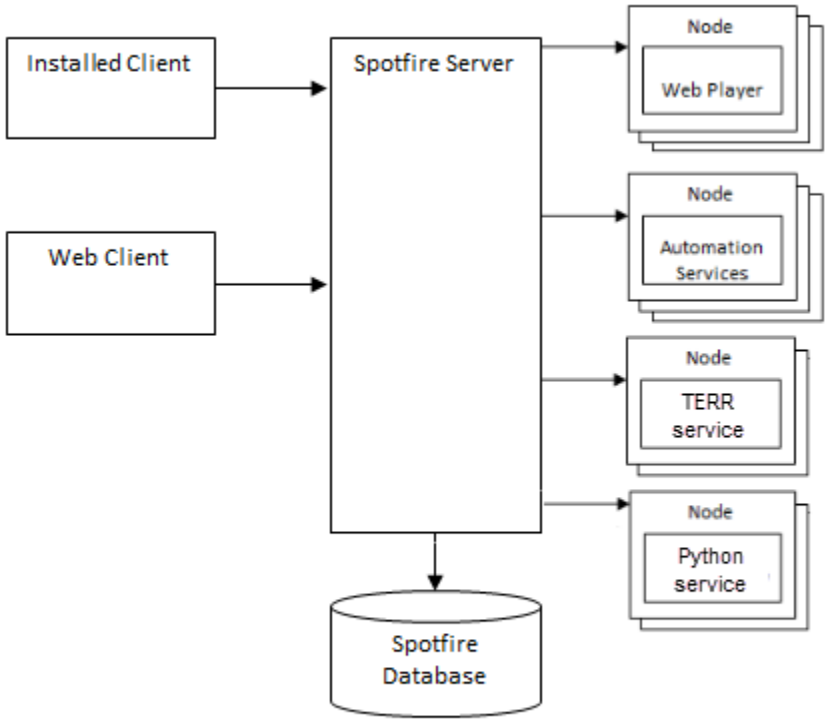
  

  You can also use the *Quick Start Guide* to complete the initial installation for a more complex implementation. In most cases it is recommended that you have a working basic installation before you add additional servers, load balancers, authentication methods, and so on.

  To begin installation, see either the Quick Start Help or the Quick Start Guide at https://docs.tibco.com/products/tibco-spotfire-server.

Introduction to the TIBCO Spotfire environment

The TIBCO Spotfire® environment is installed and configured to enable users to analyze their data in the Spotfire® clients.



Spotfire Server introduction

Spotfire Server, a Tomcat web application that runs on Windows and Linux operating systems, is the administrative center of any Spotfire environment.

In addition to providing the tools for configuring and administering the Spotfire environment, Spotfire Server facilitates the services that make it possible for users to access, blend, and visualize their data, creating analyses that provide actionable insight. The server also enables sharing of the prepared analyses to consumers.

These are the six functional areas and main functions of Spotfire Server:

Functional area	Function
Library services	Provides centralized storage of Spotfire analysis files and metadata. The library items reside in the Spotfire database.
User services	Provides user authentication and role-based authorization.
Audit services	Provides centralized collection of action logs.
Deployment services	Delivers client product upgrades and hotfixes.

Functional area	Function
Information Services	Provides a centralized point of data access and metadata management for relational data sources. The following functions are provided by Information Services: • Information links, which provide one way to access external data sources. (See also Accessing Data from External Data Sources in the Spotfire Analyst User's Guide for more data access options.) • Network input/output (I/O).
Client connections and routing services	Provides access point for all client connections. Routes clients to the appropriate service instance, based on smart default routing or configured routing rules. Continually gathers information about the state of all service instances.

Spotfire database introduction

Spotfire Server requires access to a Spotfire database.

The Spotfire database stores the information that Spotfire Server needs to control the Spotfire environment, including users, groups, licenses, preferences, shared analyses, and system configuration data.

You must have a database server up and running, preferably on a dedicated computer, before installing Spotfire Server. The Spotfire environment supports a couple of different database systems. For details on which database versions are supported, see the TIBCO Spotfire Server System Requirements.

Nodes and services introduction

Install nodes in the environment to enable the use of Spotfire web clients, Spotfire Automation Services, and the TERR service or Spotfire Service for Python.

The installed client, called Spotfire Analyst, can be used together with the Spotfire Server directly. To enable use of Spotfire web clients, Spotfire Automation Services, and the TERR service or Spotfire Service for Python, one or more nodes must also be configured, preferably on dedicated computers.

For each node, the administrator installs and enables services with a specified capability. Each Windows node can have services with the Spotfire Web Player capability, the Spotfire Automation Services capability, and the TERR service or Spotfire Service for Python capability.

The TERR service and Spotfire Service for Python are currently the only capabilities that are available for Linux nodes.

The Web Player service enables users to perform analyses in a web browser, Automation Services can be used to automate multi-step tasks, and the TERR service or Spotfire Service for Python can be used for additional calculations and advanced analytics. The capabilities of the enabled services determine the functionality that the node provides to Spotfire end users, through the Spotfire Server. For failover and performance purposes, multiple service instances can be added on each node.

You can scale your Spotfire environment by adding or removing nodes and service instances.

Spotfire clients introduction

Spotfire end users connect to Spotfire Server using either an installed client or a web client.

Spotfire Analyst is a fully-featured client for working with data sources and creating complex analyses. It is installed on a user's local computer. See the Spotfire Analyst User's Guide for more information.

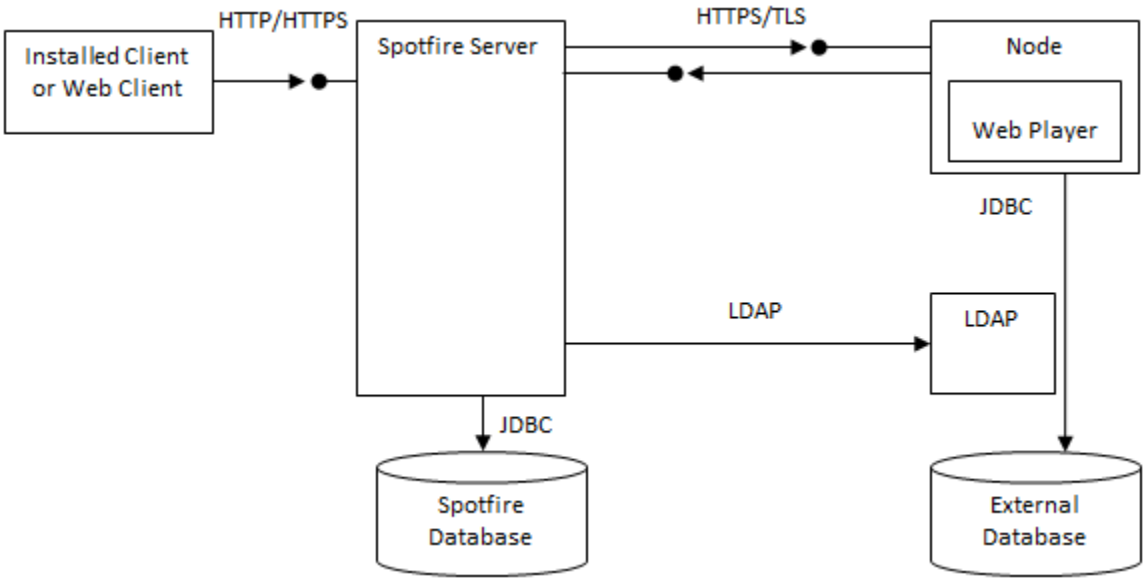
To make it possible to view or create interactive analyses in a web browser, a Web Player service must be installed on the server. Depending on which licenses a user has, the web client will have different capabilities. With a Consumer license, users can view and interact with analyses that others have created for them. With the Business Author license, it is also possible to create and edit analyses in the web client, even though not all of the functionality from the installed client is available. See the Spotfire Business Author and Consumer User's Guide for more information about which features are available in the web clients.

Environment communication introduction

All back-end communication in a Spotfire environment is secured by HTTPS/TLS, complying with current security standards and industry best practices.

Spotfire Servers listen to incoming traffic from installed clients and web clients on one HTTP or HTTPS port, the front-end communication port.

Spotfire Servers listen to traffic from services on the nodes on another HTTPS port, the back-end communication port.



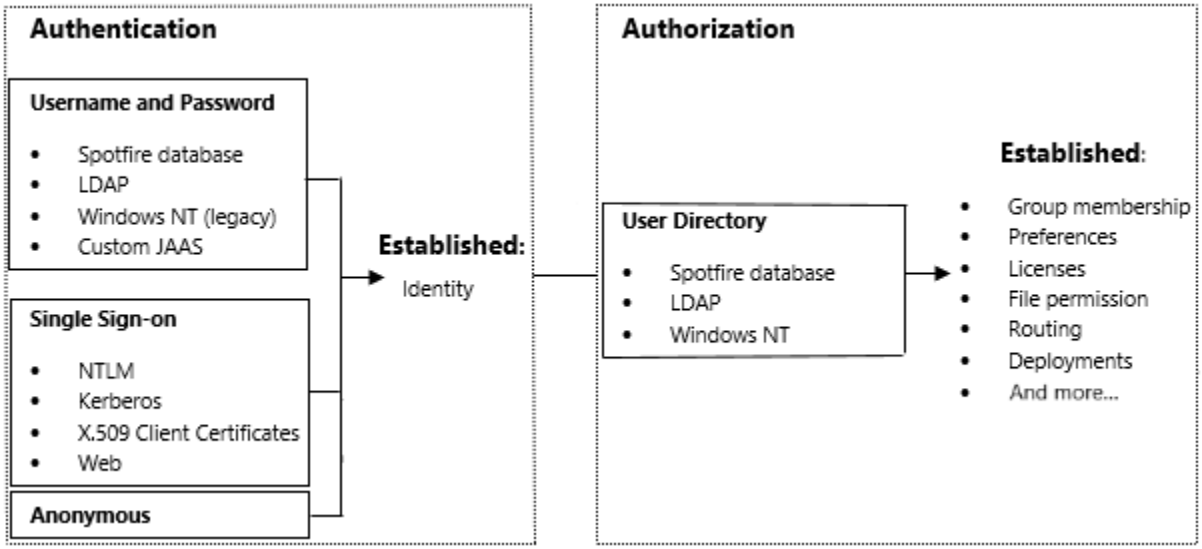
The secured back-end communication is based on certificates. After an administrator has approved the new server or node, the certificates are issued automatically. Without a certificate, a server or a service on a node cannot make requests to, or receive requests from, other entities, except for when requiring a certificate.

After being installed, a node performs a join request to a specific, unencrypted HTTP Spotfire Server port that only handles registration requests. The node remains untrusted until the administrator approves the request by trusting the node. The Spotfire Server start page provides the tools to add nodes to the environment by explicitly trusting them, thereby issuing the certificates. When the node receives its certificate, it can send encrypted communication over the HTTPS/TLS ports and with this it can start to send more than registration requests.

Authentication and user directory introduction

Installed clients, as well as web clients, connect to the Spotfire Server. When users of either client log in to a Spotfire Server, two things happen before they get access: authentication and authorization.

Authentication is the process of validating the identity of a user. After their identity is validated, the user is authorized in the user directory. Authorizing users determines their access rights within the Spotfire environment—in other words, what they are allowed to do.



If username and password are used for authentication, users are checked against the internal Spotfire user directory, a custom Java Authentication and Authorization Service module, or—the most common option—an external LDAP directory. Spotfire has built-in support for Microsoft Active Directory and the Directory Server product family, which includes Oracle Directory Server, Sun Java Directory Server, and Sun ONE Directory Server. Other LDAP servers can also be used.

For single sign-on, Spotfire supports NTLM, Kerberos, X.509 Certificates, and web authentication.

For anonymous authentication, a preconfigured Spotfire user identity is used to authenticate with the Spotfire Server.

Regardless of how the user is authenticated, the process of authorization is the same. The server checks the Spotfire user directory to determine which licenses, preferences, and permissions have been set for the user.

Optionally, the user and group accounts in the Spotfire user directory can be synchronized with an external LDAP directory. Spotfire supports the same LDAP servers for directory synchronization as it does for authentication.

For more information, see User authentication.

Users introduction

The authentication method of your Spotfire environment determines how users are added to the Spotfire database and where they are administered.

If your Spotfire implementation is configured for authentication towards the Spotfire database, the administrator adds and administers user accounts directly in the database by using the Spotfire Server administration interface.

If your implementation uses an external user directory such as LDAP, user accounts are added and administered in that context rather than in the server. Changes are copied to the Spotfire database during synchronization.

All users are organized in groups. Any user who is entered into the system automatically becomes a member of the Everyone group; this group cannot be deleted and will always contain all registered users. Licenses, which control access to Spotfire features, are assigned to groups, never to individual users.

In addition to the Everyone group, a user can belong to any number of groups, and has access to all of the license features that are enabled for those groups.

Groups and licenses introduction

Licenses provide access to the features of Spotfire. The administrator sets licenses for *groups*, thereby determining what the group members have permission to do within the Spotfire environment.

At installation Spotfire includes a number of groups that correspond to common user roles, such as Deployment Administrator or Scheduled Updates User. The administrator enables appropriate licenses for these groups, and adds to them the groups and users that perform these tasks.

Administrators often import their groups and users by synchronizing with an external user directory, but it is also possible to import users and groups from a file or create them manually. Administrators then build a hierarchy of groups to meet company requirements.

Groups can be created as subgroups, in which case group members inherit access to the licenses of all their parent groups in addition to any licenses specifically set for their own group.

Through careful planning, administrators are able to provide users with appropriate access to the Spotfire environment, within a group structure designed for easy maintenance. For more information, see How licenses work and the License and feature reference.

Preferences introduction

Preferences customize the default settings in Spotfire clients for members of a selected group. For example, an administrator may set a default color scheme for visualizations, or data optimization options.

Preferences are set in the Administration Manager in Spotfire Analyst. See the Administration Manager User's Guide for details on preference administration.

Deployment introduction

To deploy Spotfire software, the administrator places software packages in a deployment area on Spotfire Server, and assigns the deployment area to particular user groups.

If a new deployment is available when a user logs in to a Spotfire client, the software packages are downloaded from the Spotfire Server to the client.

Deployments are required for the following tasks:

- Setting up a new Spotfire environment.
- Installing a product upgrade, extension, or hotfix provided by Spotfire.
- Installing a custom tool or extension.

Administrators can create multiple deployment areas, such as "Production" and "Staging". This allows administrators to test new deployments before rolling them out to the entire client base, or to maintain different deployments for different groups of users.

Spotfire library introduction

The Spotfire database contains the Spotfire library. The library is accessible to Spotfire Analyst, and web clients through the Spotfire Server, allowing users to easily share and reuse their work.

The library stores Spotfire analyses, Spotfire binary data format files (SBDF), data functions, shared data connections (or connection data sources) created with Spotfire connectors, information links and data sources used by Information Services, and visualization color schemes.

The library is organized into hierarchical folders, which are also used to control access to folder content. The administrator creates the folder structure, and assigns groups with the appropriate read and write permissions to the folders.

Routing introduction

Spotfire provides routing capabilities within the environment.

No load balancer is required between Spotfire Server and the nodes because the default routing capability of Spotfire Server features built-in load balancing, enabling non-opened analyses to be loaded by the least utilized Web Player service instance. After an analysis is opened in a client, all subsequent requests by the user for the session are forwarded to the instance that was used for the initialization; thus Spotfire Server routing maintains analysis session affinity.

Default routing also improves capacity utilization by forwarding requests for a specific analysis file to the instance or instances of Spotfire Web Player where the file is already open, thereby serving multiple users with the same service instance. Analysis data is also shared between users, so additional users accessing the analysis file will have a low impact on performance.

In addition to the default routing, administrators can create resource pools and assign Spotfire Web Player instances to them. The resource pools abstraction enables default routing to be altered by specific routing rules. Rules can be specified for users, groups, or specific analysis files, and are defined and applied in priority order, similar to mail sorting rules. Rules can be sorted, enabled, disabled, and re-mapped to a different resource pool.

There are three health status codes for Web Player instances, used to better route traffic among the instances: Available (or OK), Strained, and Exhausted. The status codes are calculated from the CPU and memory usage on the node that is running the service instance. The current status can be observed on the diagnostics pages.

It is expected that a service instance that is frequently busy, and has high CPU or memory usage, would remain in the Strained state for long periods of time.

Also, administrators can attach schedules to routing rules that apply to analysis files, effectively turning a routing rule into a scheduled update. Thereby, the administrator can have the analysis pre-loaded on selected instances in a resource pool, and have the analysis refreshed at specified intervals.

Data sources introduction

The Spotfire environment provides several ways for clients to access data. The most common ones are: opening a local file, using a Spotfire connector, or connecting through the information services function of Spotfire Server. Users can combine data from multiple sources in a single Spotfire analysis.

Connectors

Spotfire connectors are a way for installed clients and service instances to connect directly to external data sources. Spotfire includes a range of connectors, each tailored specifically for a corresponding type of data source. Depending on the connector, users can choose to import the raw data into the memory of the local computer, or to only retrieve aggregated results and push queries to the data source based on the data currently required.

- Learn more about connectors

Information Services

Information Services is an option for connecting to JDBC-compliant external data sources. When you use Information Services to access data, the Spotfire Server makes the connection to data source on behalf of the client, using information links saved in the Spotfire library. The raw data sets are loaded into the memory of the server.

On the Spotfire Server, you must have installed a JDBC driver and a data source template for the type of data source you want to connect to. Some drivers and templates are included in the Spotfire Server installation.

- Learn more about Information Services

Logging introduction

In addition to the configurable logs for the Spotfire Server, the nodes, and the service instances, the Action Logs and System Monitoring feature helps administrators keep an eye on the health of their Spotfire environment.

The action logs collect information about system events that are sent through a web service from Spotfire Analyst, Spotfire Automation Services, and the Spotfire Web Player to the Spotfire Server. These event logs, along with those from the Spotfire Server itself, can be saved either to files or in a database.

System monitoring takes periodic snapshots of key metrics on the Spotfire Server and the Spotfire Web Player services, and stores this information in the same location as the action logs. The logs can then be analyzed in a Spotfire client.

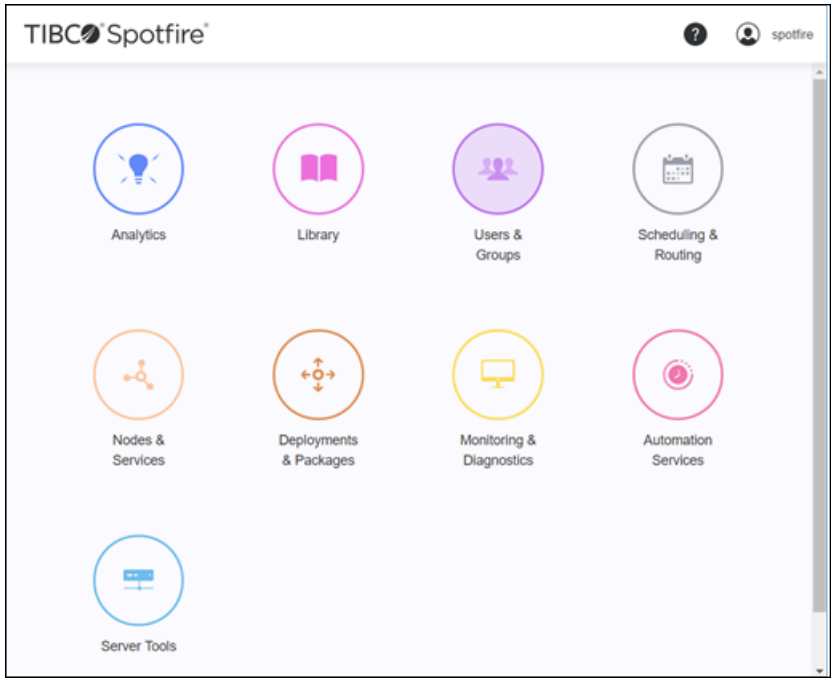
For the TERR service and Spotfire Service for Python, JMX is the only way to capture logs. for more information, see

- "Monitoring the TERR service using JMX" in TIBCO® Enterprise Runtime for R - Server Edition.
- "Monitoring the Spotfire Service for Python" in TIBCO® Spotfire Service for Python.

Administrators have many options for how to configure this feature, including which events and system statistics should be logged, from which hosts logging information should be collected, and how the logs are pruned or archived. For more information, see Monitoring and diagnostics.

Administration interface introduction

The Spotfire Server start page provides access to most administrative tasks and diagnostic information on your Spotfire environment.



- In **Analytics** you can create new analyses, and view and edit analyses that are in the Spotfire library.
- In **Library** you can manage the Spotfire library, including organizing, moving, renaming, and deleting library items.

- In **Users & Groups** you can manage users and groups, including creating users and groups, assigning licenses and adding members to groups, and changing user names and passwords.
- In **Scheduling & Routing** you can schedule updates and monitor their status, date, and time, and create routing rules applicable to groups, users, or specific analysis files.
- In **Nodes & Services** you can review the servers and services setup, add new nodes, services, and service instances, upgrade or rollback existing ones, and create resource pools for routing rules.
- In **Deployments & Packages** you can manage products, upgrades, extensions, and hotfixes by creating or altering deployment areas, adding distributions and packages, and so forth.
- In **Monitoring & Diagnostics** you can monitor the system status, set logging levels, review logs, troubleshoot and download troubleshooting bundle, create memory dumps, and more.
- In **Automation Services** you can schedule Automation Services jobs to run periodically, and view the resulting activity.
- In **Server Tools** you can download the configuration tool for Spotfire Server.

Assigning preferences, and exporting and importing users and groups, take place in the installed Spotfire Analyst client.

Example scenario

This is an example scenario of what happens in the Spotfire environment when a user opens an analysis in a web client.

1. The Spotfire web client user receives an email with a link to an analysis that contains interesting information.
2. When the link is opened, an ordinary http (or https) connection is set up from the browser to Spotfire Server. Because the environment is configured for username and password authentication, a login dialog appears.
3. If the username and password are correct, the user also needs to be listed in the user directory. Spotfire Server compares the credentials towards the Spotfire database for verification.
4. A check is made to see that the user has the license privileges to see the analysis, which is stored in the library.
5. The analysis is not already loaded on any Web Player service instance, so the routing logic of Spotfire Server selects the least utilized instance to load the analysis. The request is forwarded to this instance.
6. The Web Player service instance loads the analysis from the library.
7. Data in an analysis can be linked or embedded. This analysis contains linked data, loaded through information services. A request for the data goes back from the Web Player service instance to a Spotfire Server.
8. After the analysis and its data are loaded, Spotfire Server acts as a proxy between the web browser and the Web Player service instance.
9. The user finds the analysis interesting and wants to add an extra visualization. Because the user has the Business Author license, the menu options to do so are visible.
10. After the user has updated and saved the analysis, the user can send a link to interested parties.

Basic installation process for Spotfire

To get Spotfire up and running in a simple configuration, follow these steps. The resulting simple installation includes the following: the server on one computer, a few Spotfire Web Player instances available for other computers, the Spotfire Analyst client on another computer, and the user directory in the Spotfire database.

Prerequisites

A database server must be up and running, preferably on a dedicated computer. The Spotfire environment supports a couple of different database systems.

To view the complete system requirements, go to http://spotfi.re/sr.

1. Download installation software.
2. Download hotfixes
3. Collect the required information.
4. Set up the Spotfire database
5. Run the Spotfire Server installer.
6. Apply hotfixes.
7. Create the bootstrap.xml file.
8. Create and save a basic Spotfire Server configuration.
9. Create an administrator user.
10. Start Spotfire Server.
11. Deploy client software packages to Spotfire Server.
12. Install a node manager.
13. Trust the node.
14. Install Spotfire Web Player instances.
15. **Optional:** Install Spotfire Automation Services instances.
16. **Optional:** Install the TERR service instance. For details, see TIBCO® Enterprise Runtime for R - Server Edition.
17. **Optional:** Install the Spotfire Service for Python. For details, see TIBCO® Spotfire Service for Python.
18. Install Spotfire Analyst.

Alternatively, you can use the command line after step 5 above (see Manually creating a simple configuration) or run a script that invokes multiple commands (see Scripting a configuration).

Installation and configuration

Spotfire Server requires that the preparation, installation, database configuration, and server configuration happen in a specific order. Make sure that you follow the steps as described.

See Basic installation process for Spotfire for the required sequence.

Preparation

Prepare to install Spotfire Server by downloading the required software from the TIBCO eDelivery and Support websites, recording the required system properties, and setting up the Spotfire database on your database server.

Make sure that your system fulfills the requirements listed on the TIBCO Spotfire Server System Requirements page.

If you are upgrading, first read Upgrading Spotfire.

Downloading installation software

The first step in installing or updating Spotfire Server is to download the required software to the computer that will run the server.

Prerequisites

You must have access to the required software on the TIBCO eDelivery website. If you do not have access, contact your sales representative.

Procedure

1. On the TIBCO eDelivery website, go to the TIBCO Spotfire Server page.
2. At the bottom of the page, click **Download**, and sign in to the site if required.
3. On the server download page, select the latest version and your platform, and select the license agreement check box.

Some Spotfire releases are designated long-term support (LTS) versions; for information about how LTS versions differ from mainstream (non-LTS) versions, see https://docs.tibco.com/pub/spotfire/general/LTS/spotfire_LTS_releases.htm.

4. Under **Installation Method**, do one of the following:

   - To download the entire product, including language packs and developer software, select **Full Product with Download Manager**, click **Download**, and then follow the instructions.
   - To download fewer files, do the following:

     a. Select **Individual file download**.

     b. Under **SELECT AN INDIVIDUAL COMPONENT**, expand **TIBCO Spotfire Server Software**.

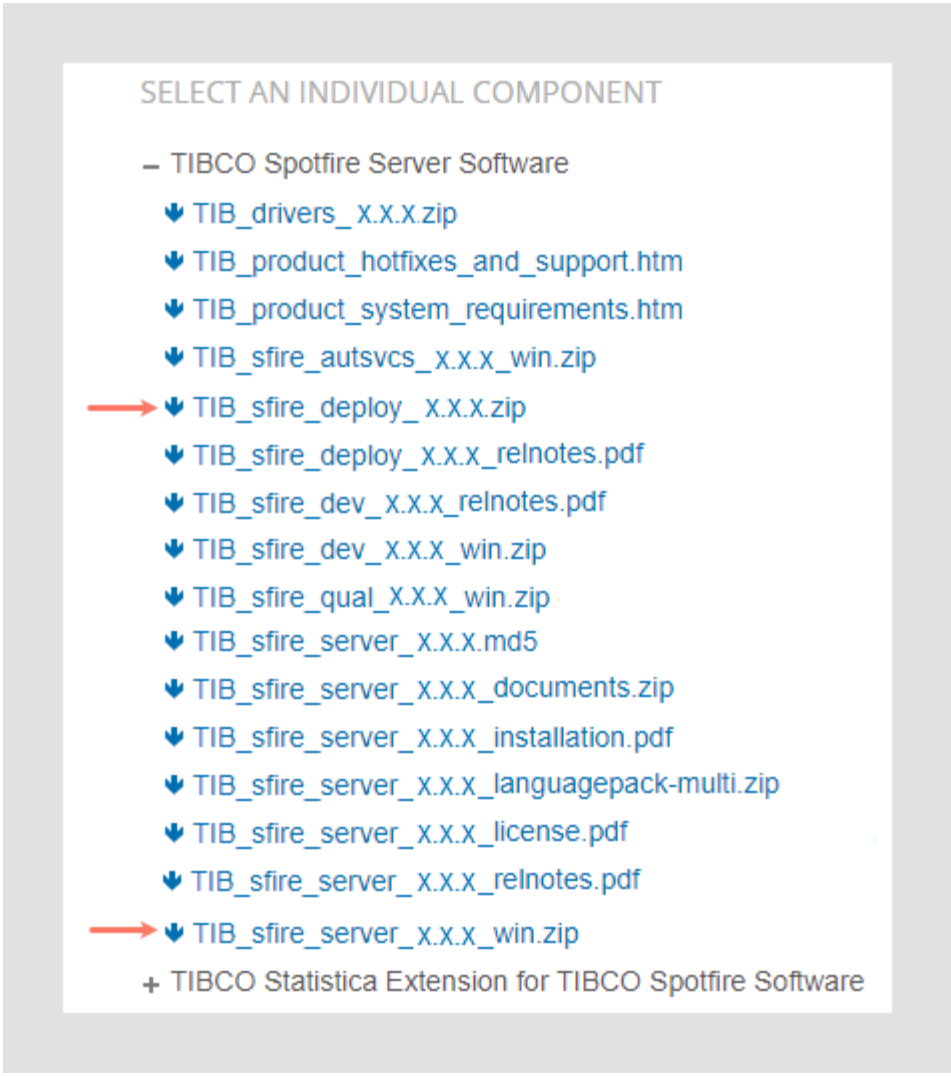
     c. Under **TIBCO Spotfire Server Software**, select `TIB_sfire_deploy_`*`version`*`.zip`.

     This file contains the client packages that your end-users will receive when connecting to the server.

     d. Select either `tib_sfire_server_`*`version`*`_win.zip` (Windows) or `tib_sfire_server_`*`version`*`_linux.tar` (Linux).

     This file contains the server installation or upgrade files.

     The following example shows the approximate location of the required software components for Windows. Note that you will need both the server file and the deploy file to upgrade your environment. The Linux options are similar.



     e. Select any other files that you want to download.

f. Unzip any zipped files that you downloaded to a folder on your desktop.

What to do next

Collect required information

Downloading hotfixes

As of Spotfire Server version 10.3.0, server hotfixes can be applied only on the specific service pack version that they were created for. Example: If you currently have version 10.3.1, you can only apply server hotfixes for the 10.3.1 version, such as 10.3.1 HF-001, 10.3.1 HF-002, and so on. If you want a hotfix of a different service pack level, such as 10.3.2 HF-001, you must first make sure to upgrade your Spotfire components to that service pack (10.3.2) before applying the hotfix.

Prerequisites

- You must have access to the required software on the TIBCO Support website. If you do not have access, contact your sales representative.

For general hotfix information and links to specific information about each hotfix, see Overview of hotfixes for TIBCO Spotfire in the TIBCO Community.

Procedure

1. Sign in to the TIBCO Support website.
2. Click **Downloads** > **Hotfixes**.
3. On the Available Hotfixes page, expand **AvailableDownloads**, **Spotfire**, and **Server**.
4. Select the .zip files containing the hotfixes for your Spotfire Server version (if you are upgrading, select the hotfixes for the new version), and click **Download**.

   The .md5 files verify the integrity of the files and do not need to be downloaded.

   The hotfixes are cumulative, so you need only the latest one.
5. Expand **Spotfire** again, and **Clients**, and download the latest hotfixes for your Spotfire clients.
6. Select the folders for the other components of your Spotfire environment, and download the latest hotfixes.
7. When the download is complete, unzip the contents of the folders to the computer running Spotfire Server.

Collecting required information

To set up the Spotfire database, and install and configure Spotfire Server, you must have certain information about the IT system at your site and how you want Spotfire Server to interact with the existing system.

Prerequisites

- A database server must be up and running before you can install Spotfire Server, preferably on a separate computer. The Spotfire Server installer will not install a database server. The Spotfire environment supports a couple of different database systems.

Procedure

1. Collect the following information about your **database server**:

   You may need to contact your database administrator.

Required information	Notes	Your information
Database server type	MSSQL,Oracle or PostgreSQL	
Database server hostname		
Administrator user name		
Administrator password		
Connection identifier	For Oracle only	

   For PostgreSQL, you should also take note of the path to the bin directory of the PostgreSQL command line tools (on the computer from which you will run the scripts).

2. Decide on the following information for the **Spotfire database**:

Required information	Notes	Your information
Spotfire database name	For MSSQL and PostgreSQL. The default is spotfire_server.	
Spotfire database user name	If the databases uses Integrated Windows authentication, note this user. If you use Integrated authentication, Spotfire Server must run as this Windows Domain user.	
Spotfire database password		

3. Decide on the following for **Spotfire Server**:

Required information	Notes	Your information
Spotfire Server front-end port	Used for communication with Spotfire clients. The default is 80. If another application on the same computer uses port 80, select a different port number.	
Back-end registration port	Used for key exchange to set up trusted communication between the Spotfire Server and nodes. The default is 9080.	

Required information	Notes	Your information
Back-end communication port (TLS)	Used for encrypted traffic between nodes. The default is 9443.	
Spotfire Server login method	Knowledge about your organization's IT infrastructure is required to set up any login method other than Spotfire database. Available login methods: • Username and password: Spotfire database, LDAP, Custom JAAS, Windows NT Domain • Single sign-on: NTLM, Kerberos, X.509 Client Certificate, web authentication	
Spotfire Server user directory	Knowledge about your organization's IT infrastructure is required to set up any user directory other than Spotfire database. Valid options are: Spotfire database, LDAP, and Windows NT Domain.	
Spotfire Server operating system		
Spotfire Servers hostnames		
Hostname of load balancer, if applicable		

What to do next

Spotfire database setup on page 31

Spotfire database setup

The database must be prepared for Spotfire before the server installer is run. The Spotfire environment supports a couple of different database systems.

Setting up the Spotfire database (Oracle)

If you are running Oracle Database, follow these steps to set up the Spotfire database before you run the Spotfire Server installer.

Prerequisites

- You have downloaded the Spotfire Server installation kit from the TIBCO eDelivery web site; for instructions, see Downloading installation software.

- The following settings must be configured on the Oracle Database server:
  - User name and password authentication.

    It is also possible to set up Spotfire Server to authenticate with an Oracle Database instance using Kerberos; for instructions, see Using Kerberos to log in to the Spotfire database. In this case, you must run the database preparation scripts manually; see Running database preparation scripts manually.
  - National Language Support (NLS) to match the language of the data you will bring into Spotfire.

    If the database server NLS cannot be set to match the language of your data, Oracle provides other methods of setting NLS to a specific database or user. For more information, consult your database administrator or see the Oracle database documentation.
- You must also have access to the Oracle Database. You may need assistance from your database administrator to copy the `install` directory to the database and to provide the database details for the script.

  The command-line database tools (for example, sqlplus) must be in the system path.

Procedure

1. Copy the `<installation files dir>/scripts/oracle_install` directory to a location where you can edit it.
2. Open the `oracle_install` directory, and then, in a text editor, open the `create_databases` script that corresponds to your platform:
   - Windows: `create_databases.bat`
   - Linux: `create_databases.sh`
   - Windows (Oracle Database running on Amazon RDS): `create_databases_rds.bat`
   - Linux (Oracle Database running on Amazon RDS): `create_databases_rds.sh`
3. In the section under "Set these variables to reflect the local environment", edit the `create_databases` script by providing the appropriate database server details.

   Definitions of the variables in create_databases

Variable	Description
ROOTFOLDER	Location where the tablespaces will be created. It must be a directory that is writable for the Oracle instance, usually *`oracle install dir`*`/oradata/`*`SID`* or *`oracle install dir`*`/oradata/`*`PDBNAME`*.  Do not add a slash or backslash after the <SID>.  This variable is not applicable for the Amazon RDS `create_databases` scripts.
CONNECTIDENTIFIER	Oracle TNS name/SID of the database/service name, for example ORCL or //localhost/pdborcl.example.com.
ADMINNAME	Name of a user with Oracle Database administrator privileges for the database identified in the CONNECTIDENTIFIER, for example "system".
ADMINPASSWORD	Password of the ADMINNAME user.

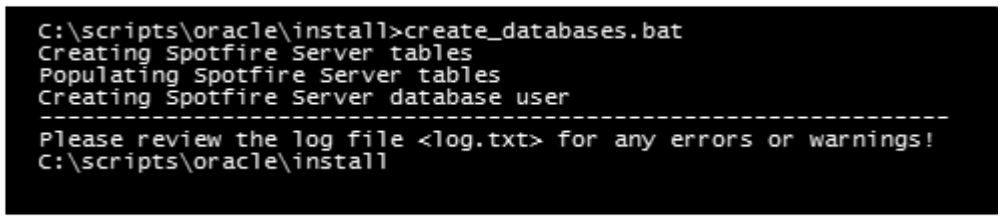
Variable	Description
SERVERDB_USER	Name of the user that will be created to set up the Spotfire database.
SERVERDB_PASSWORD	Password for SERVERDB_USER.
SERVER_DATA_TABLESPACE	Name of the tablespace that will be created. The default value works for most systems.
SERVER_TEMP_TABLESPACE	Name of the temporary tablespace that will be created. The default value works for most systems. Conflicting tablespaces can occur if you are creating the Spotfire tablespaces on a database server that is already hosting an Analytics Server or a previous version of Spotfire Server. Make sure that you do not select any names for the new tablespaces and users that conflict with the already hosted tablespaces and users.

Example

This is an example of what the file section might look like after modification:

```
rem Set these variables to reflect the local environment:
rem Where should the data be stored on the database server:
set ROOTFOLDER=C:\oracle\app\orcl
rem A connect identifier to the container database or the pluggable database
rem for a pluggable database a service name like //localhost/pdborcl.example.com
rem could be the SID for Oracle 11 or earlier, TNSNAME etc,
rem see the documentation for sqlplus
set CONNECTIDENTIFIER=//localhost/pdborcl.example.com
rem a username and password for an administrator in this (pluggable) database
set ADMINNAME=system
set ADMINPASSWORD=admin123
rem Username and password for the Spotfire instance this user will be created,
rem remember that the password is written here in cleartext,
rem you might want to delete this sensitive info once the script is run
set SERVERDB_USER=spotfire_db
set SERVERDB_PASSWORD=spotfire_db123
rem The spotfire tablespaces, alter if you want to run multiple instances in the same
 database
set SERVER_DATA_TABLESPACE=SPOTFIRE_DATA
set SERVER_TEMP_TABLESPACE=SPOTFIRE_TEMP
```

4. Save the file and close the text editor.
5. Open a command line and go to the directory where you placed the scripts.
6. Type `create_databases.bat` (Windows) or `create_databases.sh` (Linux) and press Enter.
   If the parameters are correct, text that is similar to the following text appears in the command-line interface:



The `log.txt` file is created in the same directory as the `create_databases` file. Examine this file to verify that no errors occurred, and retain the log for future reference.

Because the scripts contain sensitive information, it is good practice to remove them after your Spotfire environment has been installed.

What to do next

Install Spotfire Server

Setting up the Spotfire database (SQL Server)

If you are running Microsoft SQL Server, follow these steps to set up the Spotfire database before you run the Spotfire Server installer.

If you plan to configure Integrated Windows authentication (IWA) between Spotfire Server and the Spotfire database in SQL, see Setting up the Spotfire database (SQL Server with Integrated Windows authentication).

Prerequisites

- You have downloaded and unzipped the Spotfire Server installation kit from the TIBCO eDelivery web site; for instructions, see Downloading installation software.
- The following settings must be configured on the SQL Server:
  - TCP/IP communication listening on a port (the default is 1433).
  - Case-insensitive collation (at least for the Spotfire database).

    If your installation of SQL Server uses a case-sensitive collation by default, or your data uses a different collation than `Latin1_General_CI_AS`, you must edit the `create_server_db.sql` script before running the `create_databases.bat` script. See step 2 below.
  - The command line database tools (sqlcmd, etc.) must be in the system path.

Procedure

1. Copy the `<installation files dir>/scripts/mssql_install` directory to a location where you can edit it.
2. Optional: If your installation of SQL Server uses a case-sensitive collation by default, or if you need to define a different collation, follow these steps:
   a) On the SQL Server computer, open the `mssql_install` directory, and then open the `create_server_db.sql` script in a text editor.
   b) Locate the line `--create database $ (SERVERDB_NAME) collate Latin1_General_CI_AS;`
   c) Remove the leading dashes (--).
   d) If needed, replace `Latin1_General_CI_AS` with the name of the desired collation, but make sure it is case-insensitive (CI). See the SQL Server documentation for information about available collations.
   e) Comment out the next line by inserting leading dashes (--), so that the line looks like this: `--create database $(SERVERDB_NAME)`
   f) Save the file and close the text editor.
3. Open the `mssql_install` directory, and then, in a text editor, open the `create_databases` script that corresponds to your platform:
   - Windows: `create_databases.bat`
   - Linux: `create_databases.sh`
   - Windows (SQL Server running on Amazon RDS): `create_databases_rds.bat`
   - Linux (SQL Server running on Amazon RDS): `create_databases_rds.sh`

4. In the section under "Set these variables to reflect the local environment", edit the `create_databases` script by providing the appropriate database server details.

   Definitions of the variables in create_databases

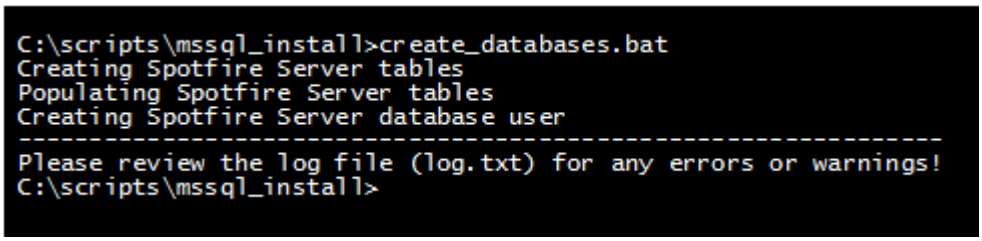
Variable	Description
CONNECTIDENTIFIER	Replace <*SERVER*> with the name of the server running the SQL Server instance, and replace <*MSSQL_INSTANCENAME*> with the name of the SQL Server instance. The default installation of SQL Server creates an unnamed instance of the SQL Server. If your SQL Server is a new installation, delete the "MSSQL_INSTANCENAME" part of the line and enter only the SERVER name. The connection will be made to the unnamed instance.
ADMINNAME	Name of a user with SQL database administrator privileges, usually "sa".
ADMINPASSWORD	Password of the *ADMINNAME* user.
SERVERDB_NAME	Name of the Spotfire database that will be created; spotfire_server is the default.
SERVERDB_USER	Name of the user that will be created to set up the Spotfire database.
SERVERDB_PASSWORD	Password for *SERVERDB_USER*.

   Example

   This is what the file section might look like after modification:

```
rem Set these variable to reflect the local environment:
set CONNECTIDENTIFIER=DBSERVER\MSSQL
set ADMINNAME=sa
set ADMINPASSWORD=admin123
set SERVERDB_NAME=spotfire_server
set SERVERDB_USER=spotfire_db
set SERVERDB_PASSWORD=spotfire_db123
```

5. Save the file and close the text editor.
6. Open a command line as an administrator and go to the directory where you placed the scripts.
7. Type `create_databases.bat` (Windows) or `create_databases.sh` (Linux) and press Enter.
   If the parameters are correct, text that is similar to the following text is displayed at the command line:



The `log.txt` file is created in the same directory as the `create_databases` file. Examine this file to verify that no errors occurred and retain the log for future reference.

Because the scripts contain sensitive information, it is good practice to remove them after your Spotfire environment has been installed.

What to do next

Install Spotfire Server

Setting up the Spotfire database (SQL Server with Integrated Windows authentication)

If you are running Microsoft SQL Server on a Windows computer in a Windows domain and plan to use Integrated Windows authentication between Spotfire Server and the Spotfire database in SQL, follow these steps to set up the database before you run the Spotfire Server installer.

Prerequisites

- You have downloaded and unzipped the Spotfire Server installation kit from the TIBCO eDelivery web site; for instructions, see Downloading installation software.
- The following settings must be configured on the SQL Server:
  - TCP/IP communication listening on a port (the default is 1433).
  - Case-insensitive collation (at least for the Spotfire database).

    If your installation of SQL Server uses a case-sensitive collation by default, or your data uses a different collation than `Latin1_General_CI_AS`, you must edit the `create_server_db.sql` script before running the `create_databases_ia.bat` script. See step 2 below.
  - The command line database tools (sqlcmd, etc.) must be in the system path.

With this type of configuration, the Spotfire database will use Windows accounts for authentication. The current user who is running the scripts to create the database must have administrative privileges on the database server, but the Spotfire process should run as a different user when connecting at runtime. Therefore, the scripts have been designed to access the database with a different Windows account when the server is running. This user is assigned to the variable WINDOWS_LOGIN_ACCOUNT. Note that the user who ran the scripts to create the database will get database owner permissions (dbo) to the database and will be able to administer the Spotfire database using integrated authentication.

If the user assigned to the WINDOWS_LOGIN_ACCOUNT variable already exists as a login on the database server, the `create_server_user_ia.sql` script must be edited. The following rows should then be commented out:

```
use master
GO
CREATE LOGIN [$(WINDOWS_LOGIN_ACCOUNT)] FROM WINDOWS WITH
DEFAULT_DATABASE=[$(SERVERDB_NAME)],DEFAULT_LANGUAGE=[us_english]
GO
ALTER LOGIN [$(WINDOWS_LOGIN_ACCOUNT)] ENABLE
GO
DENY VIEW ANY DATABASE
TO [$(WINDOWS_LOGIN_ACCOUNT)]
```

As mentioned above, the server process should connect as different user than the user that runs this script for security reasons. If you really want to use the same account then you must comment out the following lines from `create_server_user_ia.sql`:

```
CREATE USER [$(SERVERDB_USER)] FOR LOGIN [$(WINDOWS_LOGIN_ACCOUNT)]
GO
```

Procedure

1. Copy the `<installation files dir>/scripts/mssql_install` directory to a location where you can edit it.

2. If your installation of SQL Server uses a case-sensitive collation by default, or if you need to define a different collation, follow these steps:
   a) Open the `mssql_install` directory, and then open the `create_server_db.sql` script in a text editor.
   b) Locate the line `--create database $ (SERVERDB_NAME) collate Latin1_General_CI_AS;`
   c) Remove the leading dashes (--).
   d) If needed, replace`Latin1_General_CI_AS` with the name of the desired collation, but make sure it is case-insensitive (CI). See the SQL Server documentation for information about available collations.
   e) Comment out the next line by inserting leading dashes (--), so that the line looks like this: `--create database $(SERVERDB_NAME)`
   f) Save the file and close the text editor.
3. On the SQL Server computer, go to the `mssql_install` directory, and then open `create_databases_ia.bat` in a text editor.
4. In the section under "Set these variables to reflect the local environment", edit the `create_databases_ia.bat` script by providing the appropriate database server details. The definitions of the variables are listed at the top of the script.

   Definitions of the variables in create_databases_ia.bat

Variable	Description
CONNECTIDENTIFIER	Replace <*SERVER*> with the name of the server running the SQL Server instance, and replace <*MSSQL_INSTANCENAME*> with the name of the SQL Server instance.  The default installation of SQL Server creates an unnamed instance of the SQL Server. If your SQL Server is a new installation, delete the "MSSQL_INSTANCENAME" part of the line and enter only the SERVER name. The connection will be made to the unnamed instance.
WINDOWS_LOGIN_ACCOUNT	The Windows Login Account that should be created as a login on the database server. The server process must run as this user.
SERVERDB_NAME	Name of the Spotfire database that will be created; spotfire_server is the default.
SERVERDB_USER	Name of the user that will be created for the Spotfire database, associated with the WINDOWS_LOGIN_ACCOUNT.

   Example

   This is what the `create_databases_ia.bat` file section might look like after modification:

```
rem Set these variable to reflect the local environment:
set CONNECTIDENTIFIER=DBSERVER\MSSQL
set WINDOWS_LOGIN_ACCOUNT=example.com\win_user
set SERVERDB_NAME=spotfire_server
set SERVERDB_USER=spotfire_user
```

5. Save the file and close the text editor.
6. Open a command line as an administrator and go to the directory where you placed the scripts.

7. Type `create_databases_ia.bat` and press Enter.
   If the parameters are correct, text that is similar to the following text is displayed at the command prompt:

```
C:\scripts\mssql_install>create_databases_ia.bat
Creating Spotfire Server tables
Populating Spotfire Server tables
Creating Spotfire Server database user
------------------------------------------------------------
Please review the log file (log.txt) for any errors or warnings!
C:\scripts\mssql_install>
```

The `log.txt` file is created in the same directory as the `create_databases_ia.bat` file. Examine this file to verify that no errors occurred, and retain the log for future reference.

Because the scripts contain sensitive information, it is good practice to remove them after your Spotfire environment has been installed.

What to do next

Install Spotfire Server

Setting up the Spotfire database (PostgreSQL)

If you are running PostgreSQL, follow these steps to set up the Spotfire database before you run the Spotfire Server installer.

Prerequisites

- You have downloaded and unzipped the Spotfire Server installation kit from the TIBCO eDelivery web site; for instructions, see Downloading installation software.
- The following settings must be configured on the PostgreSQL server:
  - TCP/IP communication listening on a port (the default is 5432).
  - The command line database tool (psql) must be installed.

Procedure

1. Copy the `<installation files dir>/scripts/postgres_install` directory to a location where you can edit it.
2. Go to the `postgres_install` directory, and open the `create_databases` script that corresponds to your platform in a text editor:
   - Windows: `create_databases.bat`
   - Linux: `create_databases.sh`
3. Edit the `create_databases` script by providing the appropriate database server details.

   Definitions of the variables in create_databases

Variable	Description
DB_HOST	Replace <*DB_HOST*> with the name of the server running the PostgreSQL instance.
DBADMIN_NAME	Name of a user with PostgreSQL database administrator privileges, usually "postgres".

Variable	Description
DBADMIN_PASSWORD	Password of the *DBADMIN_NAME* user.
SERVERDB_NAME	Name of the Spotfire database that will be created; spotfire_server is the default.
SERVERDB_USER	Name of the user that will be created to set up the Spotfire database.
SERVERDB_PASSWORD	Password for *SERVERDB_USER*.
PSQL_PATH	The path to the bin directory of the PostgreSQL command line tools.

According to the PostgreSQL standards, it is recommended to use lower case characters for the SERVERDB_NAME and the SERVERDB_USER parameters.

Example

This is what the `create_databases.bat` file section might look like after modification:

```
rem Set this variable to the hostname of the PostgreSQL instance
set DB_HOST=dbsrv.example.com

rem Set these variables to the username and password of a database user
rem with permissions to create users and databases
set DBADMIN_NAME=postgres
set DBADMIN_PASSWORD=admin123

rem Set these variables to the name of the database to be created for the TIBCO Spotfire
rem Server, and the user to be created for TIBCO Spotfire Server to access the database.
rem Note that the password is entered here in plain text, you might want to delete
rem any sensitive information once the script has been run.
set SERVERDB_NAME=spotfire_server
set SERVERDB_USER=spotfire_db
set SERVERDB_PASSWORD=spotfire_db123

rem Set this variable to the bin directory of the PostgreSQL installation
rem where psql.exe can be found
set PSQL_PATH=C:\Program Files\PostgreSQL\12.1\bin
```

4. Save the file and close the text editor.
5. Open a command line as an administrator and go to the directory where you placed the scripts.
6. Type `create_databases.bat` and press Enter.
   If the parameters are correct, text that is similar to the following text is displayed at the command line:

```
C:\scripts\postgres_install>create_databases.bat
"Creating TIBCO Spotfire Server database and user"
"Creating TIBCO Spotfire Server tables"
"Populating TIBCO Spotfire Server tables"
-----------------------------------------------------------------
Please review the log file (log.txt) for any errors or warnings!
C:\scripts\postgres_install>
```

The `log.txt` file is created in the same directory as the `create_databases` file. Examine this file to verify that no errors occurred and retain the log for future reference. The log file will contain notices about which type is used for certain fields, which is the expected behavior.

Because the scripts contain sensitive information, it is good practice to remove them after your Spotfire environment has been installed.

What to do next

Install Spotfire Server

Running database preparation scripts manually

If you plan to set up Kerberos authentication between your database and Spotfire Server, you must run the database SQL preparation scripts manually.

Procedure

1. Read through the `create_databases` script to understand how the scripts work.
2. Run the following scripts:
   - `create_server_db.sql`
   - `populate_server_db.sql`
   - `create_server_env.sql`

For Oracle, the `create_databases` script passes the following variables to these scripts. When you run the database Oracle scripts manually, make sure to pass these variables along to the scripts:

- ROOTFOLDER
- CONNECTIDENTIFIER
- SERVER_DATA_TABLESPACE
- SERVER_TEMP_TABLESPACE

For SQL, the `create_databases` script passes the following variable to these scripts. When you run the database SQL scripts manually, make sure to pass this variable along to the scripts:

- SERVERDB_NAME

Installation

The Spotfire Server installer adds three major components to your system: A Java environment (JDK), a Tomcat application server, and a Spotfire Server web application.

Spotfire Server should run in an English (United States) language setting, as stated on the TIBCO Spotfire Server System Requirements page.

If you are upgrading, first read Upgrading Spotfire.

The JAVA_HOME of the Apache Tomcat is set to the path of the installed JDK.

Select the appropriate installation procedure for your system.

Installing the Spotfire Server files (interactively on Windows)

Running the Spotfire Server installer is the second step in a new Spotfire Server installation, after setting up the database.

This procedure describes an interactive installation, using the installation wizard. Alternatively, you can run a silent installation from the command line; for details, see Installing the Spotfire Server files (silently on Windows).

Prerequisites

The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.

For security and product performance reasons, it is recommended that you install Spotfire Server on a different computer than the database.

Procedure

1. In the server installation kit that you downloaded from the TIBCO eDelivery site, double-click `setup-win64.exe`.

   If you use Microsoft SQL Server with Windows Integrated Authentication, install Spotfire Server as the Domain User that you set up with the script `create_databases_ia.bat`. Also make sure that Spotfire Server always runs as this Domain User. Confirm with the logs that Spotfire Server starts.
2. In the installation wizard Welcome dialog, click **Next**.
3. In the License dialog, read the agreement, accept the terms, and then click **Next**.
4. In the Destination Folder dialog you can change the location if you want to, and then click **Next**.
5. In the Windows Service dialog, select the option you want and then click **Next**.
6. In the Spotfire Server Port dialog you can specify the front-end port, and then click **Next**.

   To check whether a port is in use, open a command prompt, type `netstat -na`, and press Enter.

   The ports selected during installation for front-end, back-end communication, and back-end registration ports must be open in the firewall. (The defaults are 80, 9443, and 9080.)
7. In the Backend Communication Ports dialog you can specify the back-end ports, and then click **Next**.
8. In the Ready to Install dialog, click **Install**.
   The Installing dialog tracks the progress of the installation.
9. When the installation is completed, select **Launch the configuration tool** to open the configuration tool, or **Launch the upgrade tool** if you are upgrading.

What to do next

Apply any available hotfixes for Spotfire Server: Applying hotfixes

Installing the Spotfire Server files (silently on Windows)

Instead of running the installation wizard, you can install the Spotfire Server files silently by running the installer from the command prompt.

Prerequisites

The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.

For security and product performance reasons, it is recommended that you install Spotfire Server on a different computer than the database.

To use the interactive installation wizard instead of the command prompt installation, see Installing the Spotfire Server files (interactively on Windows).

Procedure

1. Open a command prompt as an administrator.
2. Edit the default parameters in the following script. Make sure that none of the ports that you select are already in use.

```
“c:\Desktop\TIB_sfire_server_version_win\setup-win64.exe" INSTALLDIR="C:
\<MySpotfireServer>" SPOTFIRE_WINDOWS_SERVICE=Create SERVER_FRONTEND_PORT=8888
```

```
SERVER_BACKEND_REGISTRATION_PORT=7777 SERVER_BACKEND_COMMUNICATION_PORT=6666
NODEMANAGER_HOST_NAMES=localhost -silent -log "C:\Users\user\Log file2.log"
```

Silent installation parameters

Parameter	Description
INSTALLDIR	The installation directory.
SPOTFIRE_WINDOWS_SERVICE	The available options are `Create` and `DoNotCreate`.
SERVER_FRONTEND_PORT	Used for communication with Spotfire clients. The default is 80.
SERVER_BACKEND_REGISTRATION_PORT	Used for key exchange to set up trusted communication between the Spotfire Server and nodes. The default is 9080.
SERVER_BACKEND_COMMUNICATION_PORT	Used for encrypted traffic between nodes. The default is 9443.
NODEMANAGER_HOST_NAMES	A comma-separated list of IP addresses, hostnames, and FQDN names that can be used by back-end trust. These should be for the interfaces on the computer where the node manager is installed.  Valid hostnames can only contain alphabetic characters, numeric characters, hyphens and periods.  If you do not enter any values, the installer automatically provides values. After installation, confirm that these are correct in the following file: `<node manager installation dir>\nm\config\nodemanager.properties`.

3. Run the installation script.

What to do next

Apply any available hotfixes for Spotfire Server; see Applying hotfixes.

Installing the Spotfire Server files (RPM Linux)

If you have root access to the Linux computer on which you want to install Spotfire Server, you can use the RPM-based installer.

If you do not have root access, use the tarball installer instead.

Prerequisites

The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.

For security and product performance reasons, it is recommended that you install Spotfire Server on a different computer than the database.

Procedure

1. Open a terminal and go to the directory that contains the server installation file.
2. Enter the following command to install the server:

```
rpm -ivh tss-<version number>.x86_64.rpm
```

A successful execution of the command produces text similar to this:

```
Preparing...
Updating / Installing...
```

```
    l:tss-<version>-1-1
You must now execute /opt/tibco/tss/<version number>/configure to complete the
 configuration.
```

By default, the server is installed in the following directory: `/opt/tibco/tss/<version number>/`.

3. Replace the italicized parameters in the following post-installation script; the parameters are defined in the table below. Then run the script.

Alternatively, you can run the script without any parameters. In this case you will be prompted for the missing information.

```
<server installation dir>/<version number>/configure -s SERVER_FRONTEND_PORT -r
 SERVER_BACKEND_REGISTRATION_PORT -b SERVER_BACKEND_COMMUNICATION_PORT
```

Include only those parameters whose default values you are changing. The default values are listed in the table below.

Installation parameters

Parameter	Description
SERVER_FRONTEND_PORT	Used for communication with Spotfire clients. The default is 80.
SERVER_BACKEND_REGISTRATION_PORT	Used for key exchange to set up trusted communication between the Spotfire Server and nodes. The default is 9080.
SERVER_BACKEND_COMMUNICATION_PORT	Used for encrypted traffic between nodes. The default is 9443.

For more information about ports and how they are used, see Ports and firewall configuration.

Text similar to the following is shown on the command line:

```
Post install configuration of TIBCO Spotfire Server <version number> successful.
```

What to do next

Apply any available hotfixes for Spotfire Server: Applying hotfixes

Installing the Spotfire Server files (tarball Linux)

If you do not have root access to the Linux computer on which you want to install Spotfire Server, use the tarball installer rather than the RPM installer. Both the installation script and a post-installation script are run from the terminal.

Prerequisites

The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.

For security and product performance reasons, it is recommended that you install Spotfire Server on a different computer than the database.

When installing Spotfire Server on a Red Hat Linux computer with Security-Enhanced Linux enabled, it is recommended to use the RPM-based installer over the tar.gz file. If you choose to use the TAR file anyway, it is recommended to use `/opt/tibco/<subfolder>` as the installation directory.

Procedure

1. Open a terminal and go to the directory that contains the server installation file.

2. Unpack and run the TAR file by entering the following command:

```
tar xzf tss-<version number>.x86_64.tar.gz
```

The directory must contain the string "tss" for start and stop scripts to work.

The server is installed in the directory where you ran the command. A successful execution of the command does not result in any confirmation text.

3. Go to the new server directory; its name will be `tss-<version number>.x86_64`.
4. Replace the italicized parameters in the following post-installation script; the parameters are defined in the "Installation parameters" table, below. Then run the script.

Alternatively, you can run the command (`./configure`) without any parameters. In this case you will be prompted for the missing information.

```
./configure -s SERVER_FRONTEND_PORT -r SERVER_BACKEND_REGISTRATION_PORT -b
 SERVER_BACKEND_COMMUNICATION_PORT
```

Include only those parameters whose default values you are changing. The default values are listed in the table below.

Installation parameters

Parameter	Description
SERVER_FRONTEND_PORT	Used for communication with Spotfire clients. The default is 80.
SERVER_BACKEND_REGISTRATION_PORT	Used for key exchange to set up trusted communication between the Spotfire Server and nodes. The default is 9080.
SERVER_BACKEND_COMMUNICATION_PORT	Used for encrypted traffic between nodes. The default is 9443.

Text similar to the following is shown on the command line:

```
Post install configuration of TIBCO Spotfire Server <version number> successful.
```

5. Optional: If you have root access to the computer, configure the server to start when the computer starts by running this command:

```
./configure-boot
```

What to do next

Apply any available hotfixes for Spotfire Server: Applying hotfixes

Database drivers

Spotfire Server ships with the following database drivers.

- DataDirect drivers for Oracle and Microsoft SQL
- Microsoft SQL Server driver
- PostgreSQL driver
- Additional JDBC drivers for use with Information Services

Spotfire also supports the Oracle driver, which is available from the Oracle website.

Installing the Oracle database driver

If your implementation uses Oracle Database server, and you do not want to use the included DataDirect driver, you can use the Oracle driver available from the Oracle website.

Procedure

1. Download the database driver from the Oracle website.
2. Place the driver in the following directory: `<installation dir>/tomcat/custom-ext`.

Installing database drivers for Information Designer

To be able to access data from a JDBC-compliant data source with Information Services, you must install the appropriate JDBC driver on the computer that is running Spotfire Server.

If you have a clustered server deployment, you must install the driver on all computers that run Spotfire Server in the cluster.

Procedure

1. Download the database driver.
2. Place the driver in the following directory: `<installation dir>/tomcat/custom-ext`.

   This is the recommended directory for most JDBC drivers. There might be some JDBC drivers that you must install to a different directory. For information on some known exceptions, see JDBC Data Access Connectivity on the TIBCO Community
3. Restart Spotfire Server.

What to do next

To connect to an external data source, you must also add and/or enable a data source template that matches the database and the specific database driver.

You can add and enable data source templates in two different ways:

- Add a data source template with the configuration tool
- Add a data source template with the command add-ds-template

The database connection URL, used by the server to connect to the database, may differ for different database drivers; see Database drivers and database connection URLs.

Applying hotfixes to the server

Before you begin configuring Spotfire Server, you must install any available hotfix for this version of the server.

Prerequisites

- You have installed Spotfire Server.
- You have downloaded the latest hotfix for your version of Spotfire Server; for instructions, see Downloading hotfixes.

Procedure

- Follow the instructions in the `Installation_Instructions.htm` file that was included in the hotfix package that you downloaded.

  For more information, see Overview of hotfixes for TIBCO Spotfire in the TIBCO Community.

What to do next

Configure Spotfire Server; see Initial configuration.

Initial configuration

It is recommended that Spotfire administrators configure a successful basic installation of Spotfire Server before configuring more advanced implementations.

Multiple configurations can be stored in the Spotfire database, but only one can be active.

Configuration using the configuration tool

The Spotfire Server configuration tool provides a clear path to a basic installation, and offers the most frequently used configuration options.

The configuration tool must be run by a Spotfire administrator. If the Spotfire administrator does not have access to the computer running Spotfire Server, or if the server cannot display graphics, the configuration tool can be run from a local computer.

Opening the configuration tool

You can use the Spotfire Server configuration tool for the initial configuration of your Spotfire implementation, or for updating your configuration later on.

Procedure

- There are three ways to open the configuration tool:
  - Select the **Launch the Configuration Tool** check box on the last screen of the Spotfire Server installation wizard.
  - On the computer running Spotfire Server, from the Windows start menu, search for **Configure TIBCO Spotfire Server**.
  - Run the `uiconfig.bat` file (`uiconfig.sh` on Linux). These files are located in the `<installation dir>\tomcat\spotfire-bin` directory.

If you cannot run the configuration tool on the Spotfire Server computer, see Running the configuration tool on a local computer.

Running the configuration tool on a local computer

If running the configuration tool on the Spotfire Server computer is impossible or inconvenient, you can run the tool on a local computer.

Prerequisites

Java 11 runtime must be installed on the local computer.

Procedure

1. From the computer where Spotfire Server is installed, copy the `<installation dir>/tomcat/webapps/spotfire/tools/spotfireconfigtool.jar` file to the local computer.

   If Spotfire Server is up and running, you can also access the `spotfireconfigtool.jar` file on the **Server Tools** page.

2. On the local computer, unpack the `.jar` file by double-clicking the `spotfireconfigtool.jar` file.
   If your system does not recognize the file type, follow these steps instead:

   a. On the local computer, open a command line and go to the directory that contains the `spotfireconfigtool.jar` file.

   b. On the command line, enter the following command:

   ```
   java -jar spotfireconfigtool.jar
   ```

   A `spotfireconfigtool` directory is created in the same directory as the `.jar` file.

3. In the newly-created directory, double-click `uiconfig.bat` (Windows) or `uiconfig.sh` (Linux) to open the configuration tool.

Creating the bootstrap.xml file

The `bootstrap.xml` file contains basic information that the server needs to connect to the Spotfire database and retrieve its configuration. It also contains identity information for the server. If more than one server is connected in a cluster, then each server will have its own bootstrap file.

Prerequisites

Spotfire Server is installed.

For Integrated Windows authentication (IWA) between Spotfire Server and the Spotfire database, see Setting up the Spotfire Server bootstrap file for Integrated Windows authentication.

Procedure

1. If the configuration tool is not open, open it; for instructions see Opening the configuration tool. The configuration tool opens to the System Status page, which lists the necessary configuration steps.
2. Click **Create new bootstrap file**.
   The Bootstrap page is displayed.
3. Enter the following information in the fields:

Path	You may leave the default path as is.
Driver template	Select a template that is compatible with your database server.
Hostname	The Spotfire database host name (the address of the computer on which the database is installed).
Port	The Spotfire database port.
Identifier (SID/database/service)	The Server ID (for Oracle) or the database name (for MS SQL and PostgreSQL) of the Spotfire database that was created; spotfire_server is the default.
Username	The name of the database account used by Spotfire Server to connect to the Spotfire database. In the

	`create_databases.bat` file, this is the value for ADMINNAME.
Password	The password of the database account. Enter correct database login details, as specified earlier. In the `create_databases.bat` file, this is the value for ADMINPASSWORD
URL	The JDBC connection URL. This field is pre-populated from selections made but can be edited.
Driver class	This field is pre-populated from selections made, and cannot be edited. To be able to select Oracle, you must also download the JDBC driver. For details, see Database drivers and database connection URLs
Configuration tool password	Enter a configuration tool password of your choice. This will be used to protect the server configuration from unauthorized access.  The configuration tool password will be required when running the configuration tool.
Server alias	Enter any unique name for the Spotfire Server.
Encryption password (optional)	Enter an encryption password of your own choice. This will be used for encrypting other passwords stored in the Spotfire database. The passwords are encrypted with a static key if no encryption password is specified here.
Addresses	These values should match actual hostnames, fully qualified domain names (FQDN), and IP addresses (IPv4 or IPv6) at which the Spotfire Server can be reached by other Spotfire Servers and nodes. If any of these values do not describe the server, or are on a network that will not be used for backend communication, you should remove them. If you changed the hostname, domain, or IP address, add the new values.  Valid hostnames can only contain alphabetic characters, numeric characters, hyphen and period.  If you want to change these addresses after setting up your environment, use the set-addresses command.
Site	If you plan to use sites in your implementation you should assign the server to a site now. If you have not yet created the sites, see Creating sites. After creating the sites, click **Lookup** to select a site for this server. For more information, see Sites.

4. Click **Save Bootstrap**.
   The configuration tool checks that database drivers are installed and that the database is running. It also checks that the database accepts the given credentials. A message indicates whether the bootstrap file was successfully created. After it is created, the Configuration page of the configuration tool is displayed.

Setting up the Spotfire Server bootstrap file for Integrated Windows authentication

To configure Integrated Windows authentication (IWA) between Spotfire Server and the Spotfire database in SQL, follow these steps.

Prerequisites

You have followed the steps in Setting up the Spotfire database (SQL Server with Integrated Windows authentication).

Procedure

1. Change the login for the service to use the Windows account that has login rights to the Spotfire database.
2. In the bootstrap command, use the following database connection string, substituting actual values for <db_server>, <port>, and <instance>:

```
jdbc:sqlserver://<db_server>:<port>;DatabaseName=<instance>;integratedSecurity=true
```

Saving basic configuration data (authentication towards Spotfire database)

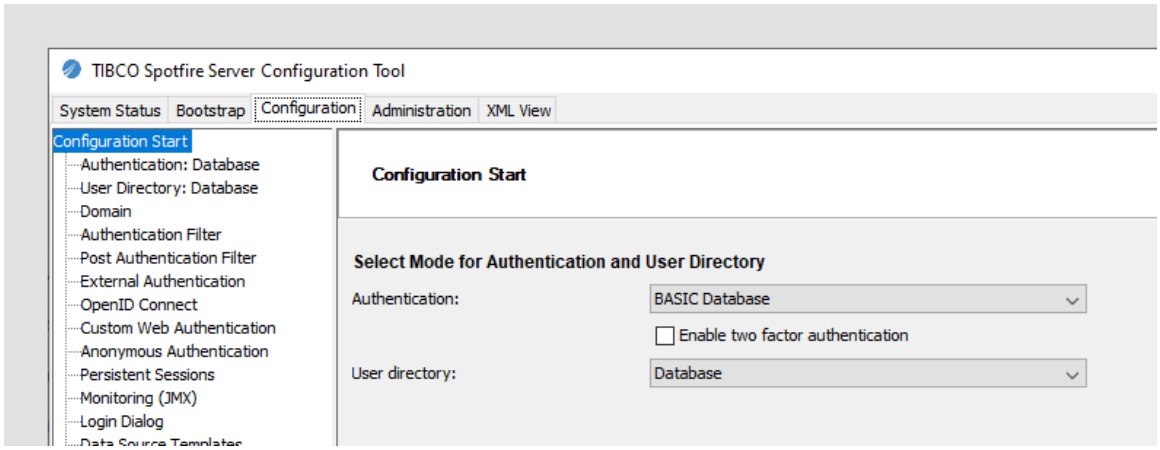
The Configuration page of the configuration tool shows the authentication type and the user directory for your installation. These instructions are for using the Spotfire database to authenticate users.

Prerequisites

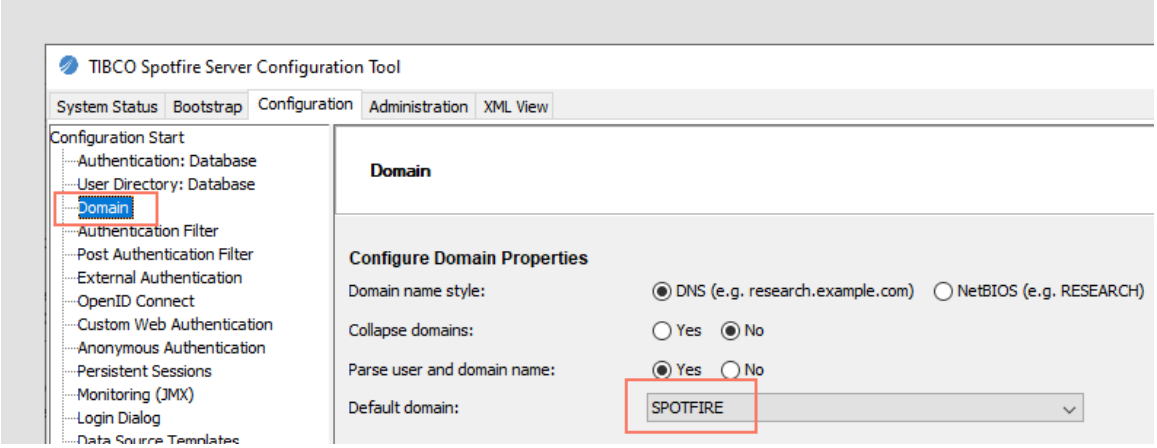
A `bootstrap.xml` file has been successfully saved (for instructions, see Creating the bootstrap.xml file).

Procedure

1. On the Configuration page of the configuration tool, verify that **BASIC Database** is selected for **Authentication** and that **Database** is selected for **User directory**.



2. In the left panel of the page click **Domain**, and then verify that **SPOTFIRE** is selected next to **Default domain**.



3. At the bottom of the page, click **Save configuration**.
   The Save configuration wizard is displayed. **Database (recommended)** is the pre-selected option, used to immediately apply the new configuration.
4. Click **Next**.
5. Enter a comment about the changes done to the configuration, and then click **Finish**.

Creating an administrator user

To continue the installation process, the administrator must create an administrator user who has access to all the functionality in the Spotfire implementation.

Prerequisites

Basic configuration data—the authentication mode and user directory for the system—have been saved on the **Configuration** tab of the configuration tool.

Procedure

1. On the Administration page of the configuration tool, under **Create new user**, enter a username and password, and click **Create**.
   The new user is displayed in the Users column.
2. Select the new user name and then click **Promote** to add that user to the Administrators group.

Deploying client packages to Spotfire Server

To install and use the Spotfire Analyst client and Spotfire web client, you must first deploy the `Spotfire.Dxp.sdn` distribution file to the server. This package is required for licenses to appear in the administration interface.

For more information about deployments, see Deployments and deployment areas.

Prerequisites

- A Spotfire Server administrator has been created. For instructions, see Creating an administrator user.
- You downloaded the `Spotfire.Dxp.sdn` file from the TIBCO eDelivery site. For details, see Downloading installation software.

Procedure

1. On the System Status page of the configuration tool, at the bottom of the list, click **Deploy client packages**.
2. In the Deploy Client Packages dialog, click **Browse**, and then locate and double-click the `Spotfire.Dxp.sdn` file.
3. Click **Deploy**.

What to do next

Node manager installation

Start Spotfire Server

Configuration using the command line

Executing commands on the command line provides greater flexibility and access to options that are not available in the configuration tool. Most administrators use the configuration tool.

The command line can be used in two ways: either by executing commands one-by-one, or by using a script containing several commands that are executed one after the other.

Executing commands on the command line

The command line offers more experienced administrators quick access to a wider variety of options than the configuration tool.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the location of the `config.bat` file (`config.sh` on Linux). The default location is `<server installation dir>/tomcat/spotfire-bin`.
   This is where you execute commands.

   You can also execute commands on a local computer rather than the server computer; for details, see Executing commands on a local computer.
2. Export the active server configuration (the `configuration.xml` file) by using the export-config command.
   Example:

   ```
   config export-config --tool-password=mypassword
   ```
3. On the command line, enter `config` (`config.sh` on Linux) followed by the command and any required parameters.
4. After you have finished running commands, upload the modified configuration back to the Spotfire database by using the import-config command. The configuration that you import becomes the active configuration for that server or cluster.
   Example:

   ```
   config import-config --tool-password=mypassword --comment=what was changed
   ```

5. Restart Spotfire Server; for instructions, see Start or stop Spotfire Server.

Because the `configuration.xml` file contains confidential information, you may want to restrict access to it.

Executing commands on a local computer

If it is more convenient, you can execute commands on a local computer rather than on the server computer.

Prerequisites

Follow the steps in Running the configuration tool on a local computer.

Procedure

1. On the local computer, on the System Status page of the configuration tool, create a new bootstrap file.
2. Each time that you run a command on the local computer, specify the location of the bootstrap file by using the `[-b value | --bootstrap-config=value]` option.

Example

To run the command `export-config` on a local computer where the `bootstrap.xml` file was placed on the desktop:

```
config export-config -b=C:\bootstrap.xml
```

Viewing help on configuration commands

You can view information about commands and their parameters from the command line.

Procedure

1. Open a command line and go to the folder that contains the **`config.bat`** file.

The default location is `<server installation dir>/tomcat/spotfire-bin`.

2. Type `config help <command name>` and press Enter.

Configuration and administration commands by function

These frequently-used commands are grouped by functional area for easy reviewing.

Command details are available in the Command-line reference. You can also view command details by running the **`help`** command on the command line (see Viewing help on configuration commands). The command parameters to use depend on your system setup and environment.

For instructions on using the commands, see Executing commands on the command line.

In general, commands work either towards the server `configuration.xml` file, or work directly on the database. For information about the server configuration files, see Bootstrap.xml file and Manual configuration.

Action log configuration commands

To configure user action logging on the Spotfire Server, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the user action database logger.	config-action-log-database-logger
Configure the action log web service.	config-action-log-web-service
Configure the user action logger.	config-action-logger

Administration commands

To perform one of these basic administration tasks, use the related command.

All administration commands connect directly to the Spotfire database and require that the server has been bootstrapped and that an initial configuration has been imported (by using the import-config command).

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Add a user or group as a member of a specified group.	add-member
Copy group membership from one principal to another.	copy-group-membership
Copy routing rules and schedules from one site to another.	copy-rules-to-site
Create a new user account.	create-user
Delete disabled users.	delete-disabled-users
Delete disconnected groups.	delete-disconnected-groups
Delete a specified OAuth2 client.	delete-oauth2-client
Delete a user account.	delete-user
Revoke full administrator privileges from a user.	demote-admin
Enable or disables a user in the Spotfire database.	enable-user
Export groups from the user directory.	export-groups
Export content from the library.	export-library-content
Export routing rules and schedules from the server.	export-rules
Export users from the user directory.	export-users
Finds scripts, data functions, and custom queries in files in the library.	find-analysis-scripts
Find URL references in analysis files (.dxp) in the library.	find-analysis-urls

Task	Command
Import groups to the user directory.	import-groups
Import content into the library.	import-library-content
Import routing rules and schedules to the server.	import-rules
Import scheduled updates from Web Player 7.0 and older.	import-scheduled updates
Import users to the user directory.	import-users
Invalidate all persistent sessions.	invalidate-persistent-sessions
List the server administrators.	list-admins
List the deployment areas.	list-deployment-areas
List all groups.	list-groups
List the currently known licenses and license functions.  You must deploy before getting licenses.	list-licenses
List registered OAuth2 clients.	list-oauth2-clients
List all online servers.	list-online-servers
List all users.	list-users
Manage the deployment areas.	manage-deployment-areas
Assign full administrator privileges to a user.	promote-admin
Register a new API client.	register-api-client
Register a new Automation Services Client Job Sender client.	register-job-sender-client
Remove a license from a group.	remove-license
Revokes consent that a specific user has given an OAuth2 client (or all such clients).	revoke-consent
Set a license and license functions for a group.	set-license
Set a new password for a given user.	set-user-password
Show the current deployment.	show-deployment
Show permissions for a specific directory in the library.	show-library-permissions
Show licenses set on the server.	show-licenses
Show the configuration of a specified OAuth2 client.	show-oauth2-client
Switch the domain names for all users and groups from one style (DNS or NetBIOS) to the other (for all configured domains).	switch-domain-name-style

Task	Command
Update the current deployment.	update-deployment

Authentication commands

To perform an authentication task, use the related command.

These commands are used to configure authentication. All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the anonymous authentication method.	config-anonymous-auth
Configure authentication and default domain.	config-auth
Configure the authentication filter.	config-auth-filter
Configure the Spotfire database authentication source for use with the BASIC authentication method.	config-basic-database-auth
Configure the LDAP authentication source for use with the basic authentication method.	config-basic-ldap-auth
Configure the Windows NT authentication source for use with the BASIC authentication method.	config-basic-windows-auth
Configure the CLIENT_CERT authentication method.	config-client-cert-auth
Configure custom web authentication.	config-custom-web-auth
Configure the external authentication method.	config-external-auth
Configure the authentication service used with the Kerberos authentication method.	config-kerberos-auth
Configure the authentication service used with the NTLM authentication method.	config-ntlm-auth
Configure authentication using OpenID Connect.	config-oidc
Configure the Persistent Sessions ("remember me") feature.	config-persistent-sessions
Configure the post-authentication filter.	config-post-auth-filter
Configure two-factor authentication.	config-two-factor-auth
Display the current authentication configuration.	list-auth-config
Display the NTLM authentication service configuration.	list-ntlm-auth
Display the current post-authentication filter configuration.	list-post-auth-filter

Task	Command
Show the LDAP authentication source for use with the basic authentication method.	show-basic-ldap-auth

Client configuration command

To configure clients connecting to the Spotfire Server, use this command.

This command works on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the client login dialog behavior.	config-login-dialog

Information Services commands

To perform an Information Services task, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Add a new data source template.	add-ds-template
Clear the default join database configuration.	clear-join-db
Configure the default join database.	create-join-db
Export the definition of a data source template.	export-ds-template
List the data source templates.	list-ds-template
Modify a data source template.	modify-ds-template
Remove a data source template.	remove-ds-template
Show the configured default join database.	show-join-database

JAAS commands

To perform a JAAS configuration task, use the related command.

The **`test-jaas-config`** command connects to the database in a read operation, but all other commands in this group work on the `configuration.xml` file. The `configuration.xml` file must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Import new JAAS application configurations into the server configuration.	import-jaas-config
List the JAAS application configurations.	list-jaas-config
Remove the specified JAAS application configurations from the server configuration.	remove-jaas-config
Test a JAAS application configuration.	test-jaas-config

LDAP commands

To manage LDAP configuration for both authentication and the user directory, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure group synchronization for an LDAP configuration.	config-ldap-group-sync
Create a new LDAP configuration to be used for authentication and/or the user directory LDAP provider.	create-ldap-config
Display LDAP configurations.	list-ldap-config
Remove LDAP configurations.	remove-ldap-config
Update LDAP configurations.	update-ldap-config

Library commands

To configure and administer the Spotfire library, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Check for inconsistencies between external storage and Spotfire database.	check-external-library
Configure the library import/export directory.	config-import-export-directory
Configure the external library data storage.	config-library-external-data-storage
Configure the file system storage of library item data.	config-library-external-file-storage
Configure the Amazon S3 storage of library item data.	config-library-external-s3-storage
Copy library permissions from one principal to another.	copy-library-permissions

Task	Command
Delete library content.	delete-library-content
Download the data of library items in Amazon S3 storage.	s3-download
Show the library import/export directory.	show-import-export-directory
Trusts scripts, data functions, or custom queries in a file in the library.	trust
Untrusts scripts, data functions, or custom queries in a file (or all files) in the library	untrust

Monitoring commands

To configure and administer JMX access to the monitoring component, use the related command.

Except for the **`config-jmx`** command, which works on the `configuration.xml` file, all monitoring commands connect directly to the database. The configuration must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the JMX RMI connector.	config-jmx
Create a new JMX user account.	create-jmx-user
Delete a JMX user.	delete-jmx-user
List all JMX users.	list-jmx-users

Server configuration commands

To perform basic server configuration tasks, use the related command.

Except for the **`create-default-config`** command, which creates a new `configuration.xml` file, all commands in this group connect directly to the database.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Create a new server configuration file containing the default configuration.	create-default-config
Export a server configuration from the server database to a file.	export-config
Import a server configuration from a file to the server database.	import-config
List all available server configurations.	list-configs
Set the current server configuration.	set-config
Show the configuration history.	show-config-history

Server database commands

To manage the server database connection pool, use the related command.

The **`bootstrap`** command creates a new `bootstrap.xml` file and optionally also attempts to connect to the database to test the file. The other commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Bootstrap the server by creating a new bootstrap configuration file.	bootstrap
Configure the encryption of sensitive information such as service account passwords.	config-encryption
Modify the common database connection configuration.	modify-db-config
Set the common database connection configuration.	set-db-config
Update an existing bootstrap configuration file.	update-bootstrap

Services commands

To configure services running on nodes, use the related command.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Delete a service configuration.	delete-service-config
Export a service configuration.	export-service-config
Import a service configuration.	import-service-config
List active (configured) service configurations.	list-active-service-configs
List available service configurations.	list-service-configs
Set the configuration for a service running in the Spotfire Server (typically the Spotfire Web Player front end).	set-server-service-config
Set the configuration for a service (running on a remote node).	set-service-config

Spotfire collective commands

To manage the Spotfire collective, use the related command.

Most commands in this group connect directly to the database and require that the server has been bootstrapped (by using the bootstrap command), and that a configuration has been imported using the import-config command. The **`config-cluster`** command works on the `configuration.xml file`, which must be imported using the import-config command for any changes to take effect. Some commands also require a running Spotfire Server to connect to.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure clustering.	config-cluster
Create a new site.	create-site
Delete a specified node.	delete-node
Delete a site.	delete-site
Download a troubleshooting bundle.	download-troubleshooting-bundle
List the addresses of a node.	list-addresses
List the certificates that establish the trust between components within the Spotfire collective.	list-certificates
List logging templates for a specified node.	list-logging
List the nodes in the collective.	list-nodes
List the service instances in the collective.	list-service-instances
List the installed services in the collective.	list-services
List the sites in the collective.	list-sites
Reset the trust within the Spotfire collective.	reset-trust
Set the addresses for a Spotfire Server node.	set-addresses
Set logging for a specified node.	set-logging
Set the site a node should belong to.	set-site
Trust a specified node.	trust-node
Revoke the trust of a specified node.	untrust-node
Updates a site	update-site

User directory commands

To configure the user directory, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the LDAP user directory.	config-ldap-userdir
Configure the user directory.	config-userdir
Configure the Windows user directory mode.	config-windows-userdir

Task	Command
List the configuration for the user directory LDAP mode.	list-ldap-userdir-config
List the current user directory configuration.	list-userdir-config
List the configuration for the user directory Windows NT mode.	list-windows-userdir-config

Miscellaneous configuration commands

To configure various aspects of the Spotfire Server, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Configure the attachment manager.	config-attachment-manager
Configure the CSRF protection.	config-csrf-protection
Configure external scheduled updates for the Spotfire Web Player.	config-external-scheduled-updates
Configure scheduled updates retries.	config-scheduled-updates-retries
Configure the public Web Service API.	config-web-service-api
Removes the value(s) of a specific configuration property.	remove-config-prop
Set the value of a specific configuration property.	set-config-prop
Configure the public address.	set-public-address

Manually creating a simple configuration

You can configure Spotfire Server by executing a series of commands on the command line.

These instructions are for using the Spotfire database to authenticate users.

Prerequisites

- The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.
- The Spotfire Server files have been installed; see Installation.

Procedure

1. Run the bootstrap command to create the connection configuration that Spotfire Server needs for connecting to the database. (For instructions on running commands on the command line, see Executing commands on the command line.)

   If you have already run the **`bootstrap`** command, there is no need to run it again unless you want to use different arguments.

   a) In the following command block, replace the argument values with the appropriate values:

   ```
   > config bootstrap --driver-class="<DRIVER CLASS>"
   --database-url="<DATABASE URL>" --username="<DATABASE USERNAME>"
   --password="<DATABASE PASSWORD>" --tool-password=
   "<CONFIG TOOL PASSWORD>"
   ```

   Argument definitions

`--driver-class`	The fully qualified class name of the JDBC driver
`--database-url`	The JDBC connection URL
`--username`	The name of the database account used by Spotfire Server to connect to the Spotfire database
`--password`	The password of the database account
`--tool-password`	Choose a command line password that will be used to protect the server configuration from unauthorized access and/or modification

   Example

   ```
   > config bootstrap --driver-class=
   "tibcosoftwareinc.jdbc.oracle.OracleDriver"
   --database-url="jdbc:tibcosoftwareinc:oracle://MyDBServer:1521;SID=spotfire"
   --username="dbuser" --password="dbpwd" --tool-password="configtoolpwd"
   ```

   A `bootstrap.xml` file is created in the `<installation directory>\tomcat\webapps\spotfire\WEB-INF` folder. For more information about this file, see The bootstrap.xml file.

2. Create a default configuration by using the create-default-config command.
   A `configuration.xml` file is created.

3. Import the configuration to the database by using the import-config command.

   a) In the following command block, replace the argument values with the appropriate values:

   ```
   > config import-config --tool-password="<CONFIG TOOL PASSWORD>" --
   comment="<DESCRIPTION>"
   ```

   Example

   ```
   > config import-config --tool-password="configtoolpwd" --comment="First config"
   ```

4. Create a first user by using the create-user command. This account can be used to log in to Spotfire Server.

   a) In the following command block, replace the argument values with the appropriate values:

   ```
   > config create-user --tool-password="<CONFIG TOOL PASSWORD>" --username=
   ```

```
"<SPOTFIRE ADMIN USERNAME>" --password="<SPOTFIRE ADMIN PASSWORD>"
```

Example

```
> config create-user --tool-password="configtoolpwd" --username="SpotfireAdmin" --
password="s3cr3t"
```

5. Add the first user to the Administrator group by using the promote-admin command.
   a) In the following command block, replace the argument values with the appropriate values:

```
> config promote-admin --tool-password="<CONFIG TOOL PASSWORD>" --username="<SPOTFIRE
 ADMIN USERNAME>"
```

   Example

```
> config promote-admin --tool-password="configtoolpwd" --username="SpotfireAdmin"
```

   When Spotfire Server is running, the first administrator can create other users and add them to the Administrator group.

What to do next

Start Spotfire Server

Deploy client packages to Spotfire Server

Scripting a configuration

For more experienced administrators, Spotfire Server includes two prepared configuration scripts that you can use to set up simple configurations. You can also create and run your own scripts.

- The `simple-config.txt` file sets up Spotfire database authentication and the user directory.
- The `simple-config-ldap.txt` file sets up LDAP authentication and the user directory.

These scripts are located in the `<installation dir>/tomcat/spotfire-bin` folder.

Example: The simple-config.txt file

The `simple-config.txt` file, shown below, is divided into three sections:

- The first two lines describe how the script is executed.
- The second section is a list of the variables that are used by the commands.
- The rest of the script contains the commands.

```
# Run this script from the command-line using the following
 command:
# config run simple-config.txt

# Before using this script you need to set the variables below:
set DB_DRIVER = "tibcosoftwareinc.jdbc.oracle.OracleDriver"
set DB_URL = "jdbc:tibcosoftwareinc:oracle://<server>:<port>;SID=
\ <SID>"
#set DB_DRIVER =
 "tibcosoftwareinc.jdbc.sqlserver.SQLServerDriver"
#set DB_URL = "jdbc:tibcosoftwareinc:sqlserver://
<server>:<port>;DatabaseName=<database name>"
set DB_USER = "<db username>"
set DB_PASSWORD = "<db password>"
set CONFIG_TOOL_PASSWORD = "<config tool password>"
set ADMIN_USER = "<admin username>"
set ADMIN_PASSWORD = "<admin password>"

echo Creating the database connection configuration
bootstrap --no-prompt --driver-class="${DB_DRIVER}" --database-
url=\ "${DB_URL}" \
  --username="${DB_USER}" --password="${DB_PASSWORD}" --tool-
password="${CONFIG_TOOL_PASSWORD}"
echo

echo Creating the default configuration
create-default-config
echo

echo Importing the configuration
import-config --tool-password="${CONFIG_TOOL_PASSWORD}" --
comment=\ "First config"
echo

echo Creating the '${ADMIN_USER}' user to become administrator
create-user --tool-password="${CONFIG_TOOL_PASSWORD}" --username=
\ "${ADMIN_USER}" --password="${ADMIN_PASSWORD}"
echo

echo Promoting the user '${ADMIN_USER}' to administrator
promote-admin --tool-password="${CONFIG_TOOL_PASSWORD}" --
username=\ "${ADMIN_USER}"
echo
```

Editing and running a basic configuration script

To use the `simple-config.txt` file to set up Spotfire database authentication and user directory, you must modify the script so that it works in your environment.

Prerequisites

- The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.
- The Spotfire Server files have been installed; see Installation.

Procedure

1. Open `<server installation dir>/tomcat/spotfire-bin/simple-config.txt` in a text editor and edit the variables:
   - Depending on the database you use, comment out DB_DRIVER and DB_URL variables ("#") that are not applicable for your system, and uncomment the variables for things you use (remove "#"). For example, if you use SQL Server, comment out Oracle variables and uncomment SQL Server variables.
   - For DB_URL, provide the specific values indicated by angle brackets.
   - For DB_USER and DB_PASSWORD, provide the Spotfire database user name and password from the `create_databases.bat` script (described in the sections listed under Spotfire database setup on page 31).
   - For the CONFIG_TOOL_PASSWORD, choose a command line password that will be used to protect the server configuration from unauthorized access and/or modification.
   - For the ADMIN_USER and ADMIN_PASSWORD, first create a user and add it to the Administrators group (see step 4 in Manually creating a simple configuration), and then provide the user name and password in the script.
2. Save the script. If you do not want to overwrite the existing script, use another name.
3. Open a command line and navigate to `<installation dir>/tomcat/spotfire-bin`.
4. Type `config run simple-config.txt` and press Enter.
   The script executes and creates a basic configuration for Spotfire Server.

The tool is conservative and does not overwrite the `bootstrap.xml` or `configuration.xml` files unless the **`--force`** flag is used.

It is recommended that you manually remove the `configuration.xml` file when you are done. Do not remove `bootstrap.xml` because it is required to start and run the server.

The `simple-config.txt` file contains sensitive information.

Script language

Spotfire provides a script language that you can use to create a script that runs multiple commands.

`#`	If a hash is the first character on a line, the line is a comment. Example: `# This is a comment that describes the next section.`
`set`	Defines a variable. The variable name and the value must be separated by an equal character (=). Example: `set PASSWORD = "abc123"`
`${Variable}`	Substitutes the dollar sign and curly braces with the variable value. If there is no matching variable, there is no substitution. Example: `--tool-password="${PASSWORD}"`
`\`	The logical line continues on the next line. Example: `bootstrap --no-prompt --driver-class="${DB_DRIVER}" \ --database-url="${DB_URL}"`
`echo`	Writes to console. Example: `echo This message will be posted echo`

	Empty rows are allowed

Paths and comments that include spaces must be enclosed in straight quotation marks ("). More advanced text editors may change straight quotation marks to smart quotation marks, resulting in errors when the commands are run.

Manual configuration

Certain configuration properties in the Spotfire system are rarely used and cannot be set using commands. To use these properties you manually edit the server configuration. You may also want to work directly in the configuration to configure features that require complex commands, such as enabling several authentication options.

There are two ways to manually edit the configuration:

- Export the configuration to a `configration.xml` file and open the file in an XML or text editor.

  Spotfire Server configurations are stored in the Spotfire database and can be exported to a `configuration.xml` file for editing or sharing. The configuration settings can also be exported to file for backup purposes, to be imported into another cluster to set up multiple clusters with similar settings, or to be sent to TIBCO Support for inspection. For instructions, see Manually editing the server configuration in an XML or text editor.
- Open the Spotfire configuration tool and edit the configuration on the XML View page.

  The active configuration file can also be edited in the Spotfire configuration tool, without having to export the file. For instructions, see Manually editing the server configuration in the configuration tool.

Manually editing the server configuration in an XML or text editor

Before editing the Spotfire Server configuration in an XML editor or a text editor, you must export the configuration to an XML file.

Alternatively you can edit the configuration in the configuration tool; see Manually editing the server configuration in the configuration tool.

For general information, see Manual configuration.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the location of the `config.bat` file (`config.sh` on Linux). The default location is `<installation dir>/tomcat/spotfire-bin`.
2. Export the active configuration to a `configuration.xml` file by using the export-config command. The `configuration.xml` file appears in your working directory.
3. Open `configuration.xml` in an XML editor or a text editor and make your changes.
4. When you have finished, save and close the file.
5. Upload the edited configuration file back to the Spotfire database by using the import-config command.
6. Restart the Spotfire Server; for instructions, see Start or stop Spotfire Server.

Result

The imported configuration becomes the active configuration for that server or cluster.

Manually editing the server configuration in the configuration tool

You can manually edit the active server configuration directly in the Spotfire configuration tool. This method of editing the configuration properties does not require you to export the configuration to an XML file.

Alternatively you can edit the configuration in an XML or text editor; see Manually editing the server configuration in an XML or text editor.

For general information, see Manual configuration.

Procedure

1. If the configuration tool is not open, open it; for instructions see Opening the configuration tool.
2. On the XML View page, click **Edit**.
3. Locate the sections that you want to change and make your edits. You can search using the box in the upper-right corner of the page.
4. When you've completed your edits, click **Update**.
5. On the Configuration page, click **Save configuration**.
6. Restart the Spotfire Server; for instructions, see Start or stop Spotfire Server.

Start or stop Spotfire Server

You must start Spotfire Server after completing initial configuration of the server, before deploying client packages. In addition, you must restart Spotfire Server any time that you change its configuration. The restart causes the server to retrieve a fresh copy of the `configuration.xml` file from the database.

Starting or stopping Spotfire Server (as a Windows service)

After configuring Spotfire Server, you must start it.

Prerequisites

You have successfully completed the initial configuration steps so that the System Status page of the configuration tool shows check marks before the following steps:

- Connect to Database
- Specify Configuration
- Configure Spotfire Server Settings
- Specify Server Administrator

Procedure

1. Log in to the Spotfire Server computer as an administrator.
2. Go to **Control Panel** > **Administrative Tools** > **Services** and then, in the Services dialog, locate and select the service called **TIBCO Spotfire Server**.
3. To the left of the services list, click **Start** in the phrase "Start the service".

To stop the service, click **Stop** to the left of the services list.

Result

The Status is changed to "Running".

What to do next

- Deploy the latest client package to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Starting or stopping Spotfire Server (Windows, no service)

If you did not install a Windows service you must start Spotfire Server manually.

Prerequisites

You have successfully completed the initial configuration steps so that the System Status page of the configuration tool contains four green check marks.

Procedure

1. Log in to the Spotfire Server computer as an administrator.
2. Open a command prompt and go to the following folder: `<installation dir>/tomcat/bin`.
3. Run the `startup.bat` file.

Result

Spotfire Server starts.

The server will stop running if you close the command prompt or log off from the computer.

What to do next

- Deploy the latest client package to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Starting or stopping Spotfire Server (Windows, service exists, Integrated Authentication for SQL Server)

If your database server uses Integrated Windows Authentication (IWA) for SQL Server, your Spotfire Server must run as a Windows Domain user that has permission to use the Spotfire database.

Prerequisites

You have successfully completed the initial configuration steps so that the System Status page of the configuration tool contains four green check marks.

Procedure

1. Go to **Control Panel** > **Administrative Tools** > **Services**.
2. Double-click the service called **TIBCO Spotfire Server**.
   The Properties dialog opens.
3. In the Properties dialog, click the **Log On** tab.
4. Click the **This account** radio button and enter the user credentials of the Domain User that was set up with the database preparation script `create_databases_ia.bat`.
5. Click **OK**.
6. Start or stop the service.

What to do next

- Deploy the latest client package to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Starting or stopping Spotfire Server (Windows, no service, Integrated Authentication for SQL Server)

If your database server uses Integrated Windows Authentication (IWA) for SQL Server, your Spotfire Server must run as a Windows Domain user that has permission to use the Spotfire database.

Prerequisites

You have successfully completed the initial configuration steps so that the System Status page of the configuration tool contains four green check marks.

Procedure

1. Log in to the Spotfire Server computer as the Domain User that was set up with the database preparation script `create_databases_ia.bat`.
2. Open a command prompt and go to the following folder: `<installation dir>/tomcat/bin`.
3. Run the `startup.bat` file.

Result

Spotfire Server starts.

The server will stop running if you close the command prompt or log off from the computer.

What to do next

- Deploy the latest client package to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Starting or stopping Spotfire Server (Linux)

On Red Hat and SUSE systems, the Spotfire Server service starts on system startup. Only a user with root user privileges can start and stop the server.

Prerequisites

You have successfully completed the initial configuration steps so that the System Status page of the configuration tool contains four green check marks.

Procedure

1. Log in as root or run with sudo -s.
2. Enter the command `/etc/init.d/tss-<version number> start`.

To stop the server, enter the command `/etc/init.d/tss-<version number> stop`.

What to do next

- Deploy the latest client package to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Clustered server deployments

Large companies often opt for clustered server deployments, where several Spotfire Servers share a database and work together to carry out the server tasks.

Clustered servers provide the following benefits:

- Failover protection if a server goes down.
- Scalability for the growing organization.
- Better performance in a system that handles a high volume of work.

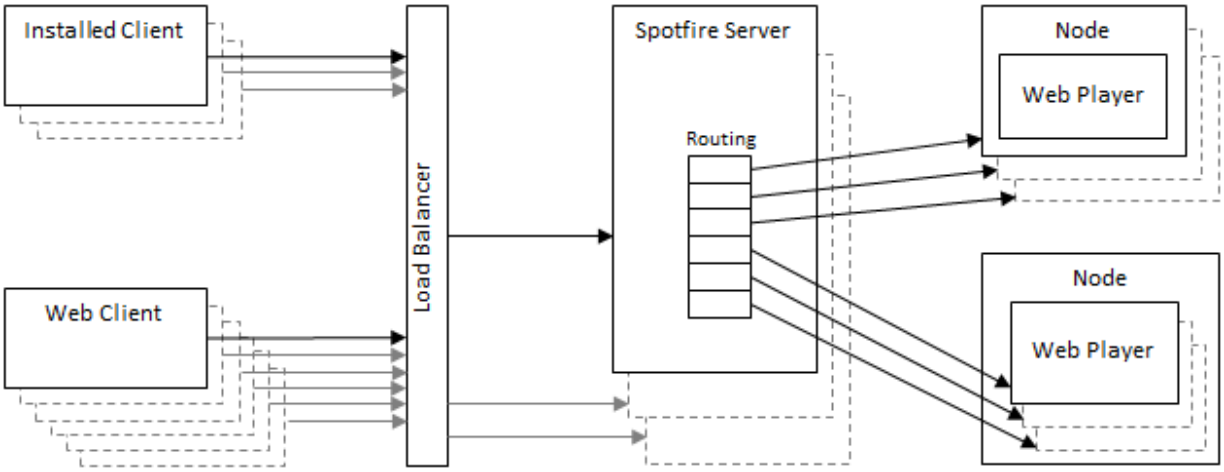
Clustering is enabled by default in Spotfire Server.

You must have at least one working Spotfire Server before you configure the cluster.

Configuration

Usually a load balancer is added to the clustered deployment, in front of the servers, to help distribute the workload, but this is not required. The load balancer you choose must support session affinity; this means that after a session has been established, the load balancer continues to route all requests from a particular client to a particular server. To be able to analyze streaming data in web clients, the load balancer must support WebSockets from the web client to the Spotfire Servers. Companies must supply their own load balancer.

The following diagram shows a typical clustered deployment with a load balancer.



A cluster may also contain multiple Spotfire Servers that can be accessed individually through their URLs, but share the same set of node managers.

Apache Ignite

Clustering is implemented using Apache Ignite. Apache Ignite clustering requires no manual configuration. It provides TLS version 1.2 for communication and a fast clustering solution. In addition, it looks for specific nodes by using their IP address, rather than discovering any node that communicates using multicasting.

Ports

By default, Apache Ignite uses these three ports:

- 5701 (this base value is configurable)
- 5702 (base value + 1)
- 5703 (base value +2)

You can change the default clustering ports when you configure the cluster, either in the configuration tool or by using the **`config-cluster`** command. For details, see Setting up a cluster of Spotfire Servers or config-cluster.

Multiple network interfaces

If there are multiple network interfaces, and at least one of these network interfaces is not reachable from all the nodes and servers in the cluster, you must configure each server and node manager to use a specific address/network interface.

To configure the correct address/network interface for servers to use, set the `addresses` property during server configuration; see Creating the bootstrap.xml file. Otherwise, use the **`set-addresses`** command to set the correct address/network interface for the servers; see set-addresses.

The correct address/network interface for node managers to use must be set when you install each node manager. For a Windows node manager, this is set on the Network Names page of an interactive installation, or the NODEMANAGER_HOST_NAMES in a silent installation. For a Linux node manager, set the NODEMANAGER_HOST_NAMES in the post-installation script.

Session clustering (for failover)

Session clustering enables users to continue their work without interruption even if a server goes down or must be restarted.

The sessions are stored in a shared in-memory store, so there must be at least one additional server online (within the same site) for the session to be kept. Note that there is no persistent storage available, so if all servers in a site are restarted at the same time, then the sessions will be lost.

The end-user will normally not notice that a session is handed over to another server, with the following exceptions:

- If the user had an ongoing execution of an information link on the server that went down.
- If the user was saving to or opening anything from the library on the server that went down.
- If an administrator user was deploying packages (using the Deployment & Packages app).

To disable the session clustering, change the `clustering.session-clustering.enabled` configuration property to false (the default is true), using the set-config-prop command:

```
config export-config --force
config set-config-prop -n clustering.session-clustering.enabled -v false
config import-config -c "Disabled session clustering"
```

Upgrading

Upgrading to a new version of Spotfire Server automatically switches any existing clusters to Apache Ignite clustering, if the old server environment was based on an earlier solution.

Setting up a cluster of Spotfire Servers

Some deployments that include clustered Spotfire Servers are very complex, and their installation and configuration are best left to a Spotfire consultant. However, if you plan to do it yourself, follow these guidelines.

Prerequisites

- The Spotfire database has been prepared; for instructions, see Spotfire database setup on page 31.

- If you plan on adding a load balancer in front of the servers, your load balancer supports session affinity; this means that after a session has been established, the load balancer continues to route all requests from a particular client to a particular server.
- To be able to analyze streaming data in web clients, the load balancer must support WebSockets from the web client to the Spotfire Servers.

For general information about Spotfire Server clusters, see Clustered server deployments.

Procedure

1. Install Spotfire Server on each computer; for instructions, see Installation.

   For reasons of security and performance, do not install a Spotfire Server on the same computer as the database. (This is true for non-clustered systems as well.)

   a) Ensure that all the clustered Spotfire Servers have the same:

      - Version number
      - Database
      - Database drivers
      - Encryption password. This is an optional setting on the Bootstrap page of the configuration tool.

2. Apply any available hotfix to each server. For instructions, see Applying hotfixes to the server.
3. Start at least one server to verify that it is working.
4. Optional: If you want to change the clustering port, follow these steps on only one of the servers to modify the shared Spotfire Server configuration.

   Make sure that none of the servers are running before you change the clustering configuration. For instructions, see Start or stop Spotfire Server.

   These instructions are for using the configuration tool. Alternatively you can use the config-cluster command on the command line. For more information, see Executing commands on the command line.

   a. If the configuration tool is not open, open it; for instructions see Opening the configuration tool.
   b. On the Configuration page, at the bottom of the left pane, click **Clustering**.
   c. Next to **Port**, enter the TCP/IP port that you want to use for clustering. This port is the same for all servers in the cluster. (The default is 5701.)

      Apache Ignite uses three ports. For information about how the ports are numbered, see the "Ports" section of Clustered server deployments. Make sure that on all the servers, these ports are not protected by a firewall.

   d. At the bottom of the page, click **Save configuration**.

5. Start all the servers in the cluster.

Configuring NTLM for a cluster of Spotfire Servers

To configure NTLM for clustered servers, first set the options common to all the servers and then set the server-specific options.

For general information about Spotfire Server clusters, see Clustered server deployments.

Procedure

1. Configure the options common to all servers in the cluster. This is performed according to the instructions in Configuring NTLM authentication for a single server, with the following modifications:
   - Specify the `DNS domain name` (recommended) or a `domain controller` (not recommended), and possibly also an `AD site name`.

     Do not specify the **`server`**, **`account name`**, or **`password`** options at this point.
2. Run the config-ntlm-auth command to add the account information for each Spotfire Server in the cluster:
   - Run the command once for each server in the cluster.
   - Enter the **`server`**, **`account name`**, and **`password`** options. The server option must reflect the server name as defined in the server's `bootstrap.xml` file.

Kerberos authentication for clustered servers with load balancer

In a clustered environment where Kerberos authentication is used to authenticate users, the load balancer forwards all Kerberos authentication information to the Spotfire Servers. No configuration on the load balancer is needed, but there are certain considerations to take into account when Kerberos authentication is set up.

These are the special considerations:

- Two Service Principal Names must be created for each Spotfire Server as well as for the load balancer.
- One keytab file must be created. This must use the fully qualified Service Principal Name of the load balancer.
- This keytab file must be copied to each Spotfire Server.
- When Kerberos authentication is set up, the fully qualified Service Principal Name of the load balancer must be provided.

For general information about Spotfire Server clusters, see Clustered server deployments.

X.509 client certificates for clustered servers with load balancer

When using X.509 client certificate authentication in a clustered environment, the clients see the load balancer as the server. The load balancer must therefore be provided and configured with a server certificate and its private key.

The load balancer also needs to be provided and configured with the CA certificate that was used to issue the server certificate.

For general information about Spotfire Server clusters, see Clustered server deployments.

Terminating TLS in a load balancer or reverse proxy

By providing some parameters in the HTTP Tomcat connector, present in `server.xml`, you can make the server behave as if it uses HTTPS (e.g, set secure cookies etc.), even when the server itself has not been configured with HTTPS.

If HTTPS is enabled on the Spotfire Server, then this results in a number of behavior changes, including that HTTP session cookies are marked as secure.

However, it is also possible to configure a load balancer or reverse-proxy with HTTPS, and use plain HTTP between the load balancer and the Spotfire Server. In these cases, the Spotfire Server will not automatically know that the connection is secure (from the client's point of view) and it will need some additional steps to set the secure attribute on cookies.

To indicate that the connection should be seen as secure, even though it uses HTTP, follow the instructions below:

These steps must be performed on all servers in a cluster.

Procedure

1. Open the following file in an XML editor or a text editor: `<server installation dir>/tomcat/conf/server.xml`
2. Locate the information about the HTTP connector (at the top of the file).
3. Add the following attributes to the list:

   `scheme="https"`

   `secure="true"`

   `proxyPort="443"` (or the port that your load balancer uses)

   `proxyName="example.com"` (the host name of your load balancer)

   The most important attribute here is `secure` – the other attributes are used only in some special cases (normally, the configured public address is used instead).
4. Save and close the file.
5. Restart the Spotfire Server.

Example:

```
<Connector port="80"
 maxHttpHeaderSize="65536"
 connectionTimeout="30000"
 enableLookups="false"
 URIEncoding="UTF-8"
 disableUploadTimeout="true"
 server="TIBCO Spotfire Server"
 compression="on"
 compressibleMimeType="text/html,text/xml,text/plain,text/css,application/
json,application/javascript,image/svg+xml,application/xml"
 acceptorThreadCount="2"
 keepAliveTimeout="30000"
 maxKeepAliveRequests="-1"
 maxThreads="2000"
 scheme="https"
 secure="true"
 proxyPort="443"
 proxyName="example.com"/>
```

Streaming and WebSockets for load balanced servers

When using a load balancer in front of a cluster of Spotfire Servers, depending on the load balancer's idle connection timeout, either the Spotfire Server configuration property `services.webplayer.websockets.client-heartbeat-interval-seconds` (default is 25 seconds) or the load balancer's timeout setting needs to be adjusted so that WebSocket connections are not closed prematurely. Some load balancers might require special handling of the WebSocket protocol upgrade mechanism. Spotfire web client WebSockets are established on `/spotfire/wp/ws`.

For general information about Spotfire Server clusters, see Clustered server deployments.

Procedure

1. Open a command-line interface and export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=services.webplayer.websockets.client-heartbeat-interval-seconds --value=XX
   ```

   where XX is an integer and represents the heartbeat interval in seconds, the value 0 disables heartbeats.

   For information about the command options, see set-config-prop.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Servers in the cluster.

Enabling health check URL for load balanced servers

When using a load balancer in front of a cluster of Spotfire Servers, a health check URL can be set up to show the status of the servers.

For general information about Spotfire Server clusters, see Clustered server deployments.

Procedure

1. Open a command-line interface and export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=status-controller.enabled --value=true
   ```

   For information about the command options, see set-config-prop.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Servers in the cluster.

Result

You can now use the URL `/spotfire/rest/status/getStatus` to check the status of the servers in your cluster.

- If the health check URL hasn't been enabled, the HTTP code 404 is returned.
- If the server is up and running, the HTTP code 200 is returned along with the text RUNNING.
- If the server is currently starting or stopping, the HTTP code 503 is returned along with the text STARTING or STOPPING.

Configuring shared import and export folders for clustered deployments

From the Library Administration tool in Spotfire Analyst, you can import and export library content. The import and export files are stored in a folder specified in the Spotfire Server configuration. In a clustered environment, where the client could be communicating with any of the servers, steps must be taken to ensure that the import and export files are always stored in the same folder.

For general information about Spotfire Server clusters, see Clustered server deployments.

Procedure

- Using Windows shared folder technology, set the location of the import and export folder to a folder that is shared with all the Spotfire Servers in the cluster.

User authentication

Spotfire supports a variety of user authentication protocols for verifying the identities of users logging in to the program.

To configure authentication, you select both an *authentication method* and a *user directory*.

Spotfire supports the two main types of authentication—user name and password, and single sign-on—as well as two-factor and external methods.

User name and password authentication methods

When users start a Spotfire Analyst client, they select which Spotfire Server to connect to. If that server is configured for a user name and password based authentication method, the users are also prompted for their user name and password.

The user name and password are then sent to Spotfire Server.

The login experience for the Spotfire Analyst client can be customized in several ways, including whether users have the option to save their login information, and whether the dialog contains an RSS feed. For details, see Login behavior configuration.

The credentials that users enter are not encrypted when they are transferred to Spotfire Server unless the server uses TLS. To help counter the risks associated with unencrypted data, enable TLS when configuring a user name and password authentication method.

For all the user name and password methods, an entry for each user is created in the Spotfire database.

- If you configure authentication towards an external user directory such as an LDAP directory, the user list or group hierarchies from the external directory are automatically copied to the Spotfire database.
- If you configure authentication towards the Spotfire database, the user and group information must be manually entered.
- It is possible to combine authentication towards an external user directory with users added manually to the Spotfire database.

Authentication towards the Spotfire database

This authentication method requires that the Spotfire user directory be configured for Spotfire database.

When the user directory is set to **Database**, the administrator usually enters the user names and passwords into the Spotfire database manually. The names and passwords can also be imported from a CSV file, or automatically created as new users log in to the server. The option to automatically create users is available through the *post-authentication filter*.

Authentication towards the Spotfire database is the default configuration for Spotfire Server, so no special configuration is required. It is easy and fast to set up and it is recommended for small implementations.

Authentication towards LDAP

This authentication method integrates with an existing LDAP directory and delegates the actual authentication responsibility to its configured LDAP servers.

The result is that only users with valid accounts in the LDAP directory can log in to Spotfire Server. This setup is recommended for larger implementations.

Spotfire Server supports the following LDAP servers:

- Microsoft Active Directory
- The Directory Server product family (Oracle Directory Server, Sun Java System Directory Server, Sun ONE Directory Server, iPlanet Directory Server, Netscape Directory Server)

Other types of LDAP servers may also work with Spotfire Server, but require more advanced configuration.

When Spotfire Server is authenticating towards a Microsoft Active Directory server, it automatically uses the Fast Bind Control (also known as Concurrent Bind Control) option to minimize the consumed resources on the LDAP server.

LDAP authentication can be combined with either the LDAP user directory or the Spotfire database user directory:

- When the user directory is set to **LDAP**, Spotfire Server can automatically import the user names from the LDAP directory. Passwords remain in the external directory, and Spotfire Server contacts this directory to validate users' passwords. You can set the frequency with which Spotfire Server checks the LDAP directory for updates.

  When the user directory mode is set to **LDAP**, Spotfire Server also imports the group names and group membership information. For information on groups, see Users & groups introduction and Group administration.

- When the user directory mode is set to **Database**, the administrator usually enters the valid user names and passwords into the Spotfire database manually. The names and passwords can also be imported from a CSV file, or be automatically created as new users log in to the server. The option to automatically create users as they log in is available through the post-authentication filter.

Configuring LDAP

When user authentication is configured towards an LDAP directory, Spotfire Server delegates authentication responsibility to the configured LDAP servers. Therefore only users with valid accounts in the LDAP directory can log in to Spotfire Server.

For information about supported LDAP servers and what you need to know about your organization's server, see Authentication towards LDAP.

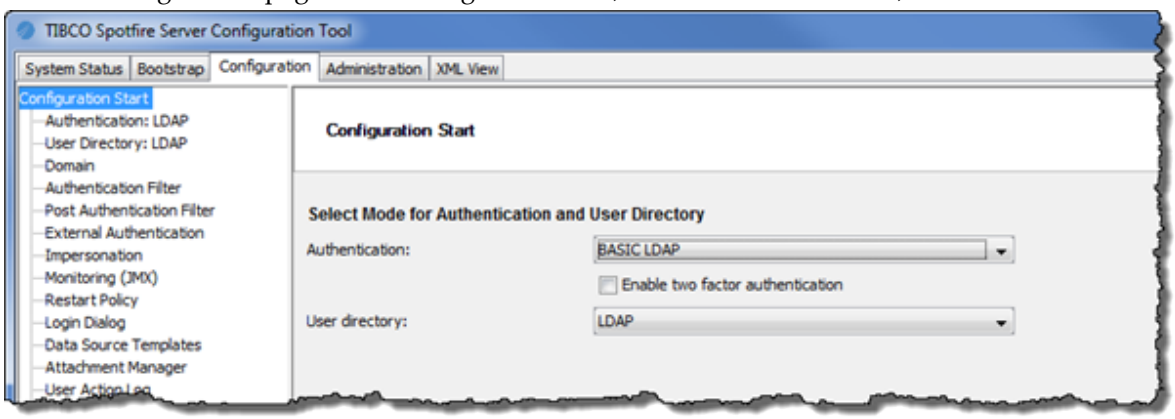
For information about other LDAP implementations, including Kerberos, NTLM, X.509 client certificates, and external authentication, see User authentication.

Prerequisites

- Your organization stores user information in an LDAP directory.
- A `bootstrap.xml` file has been successfully saved in the configuration tool; for instructions, see Creating the bootstrap.xml File.

Procedure

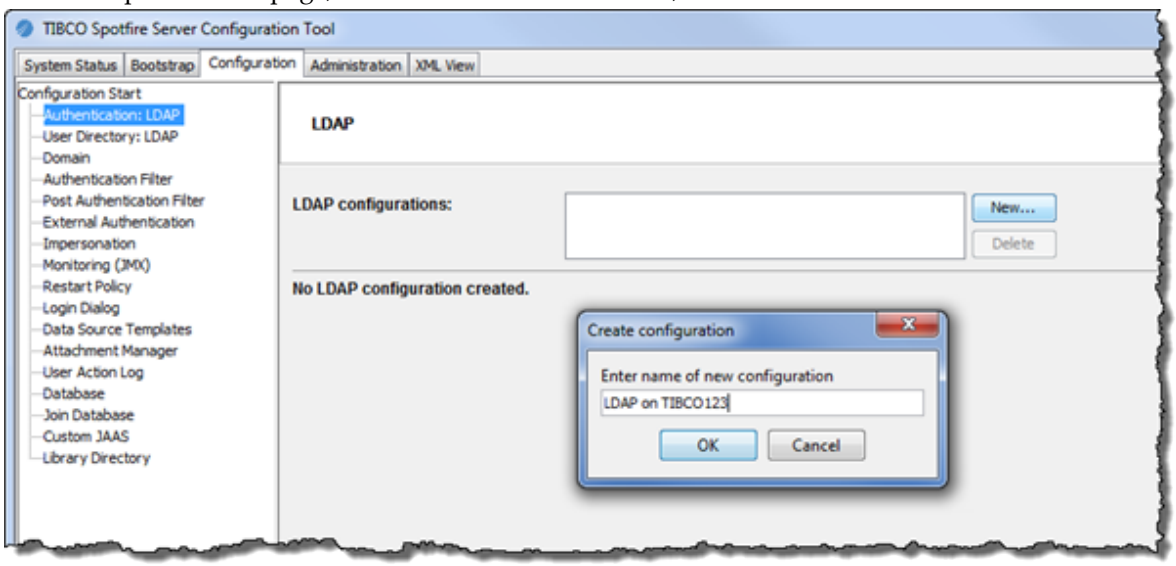
1. On the Configuration page of the configuration tool, next to **Authentication**, select **BASIC LDAP**.



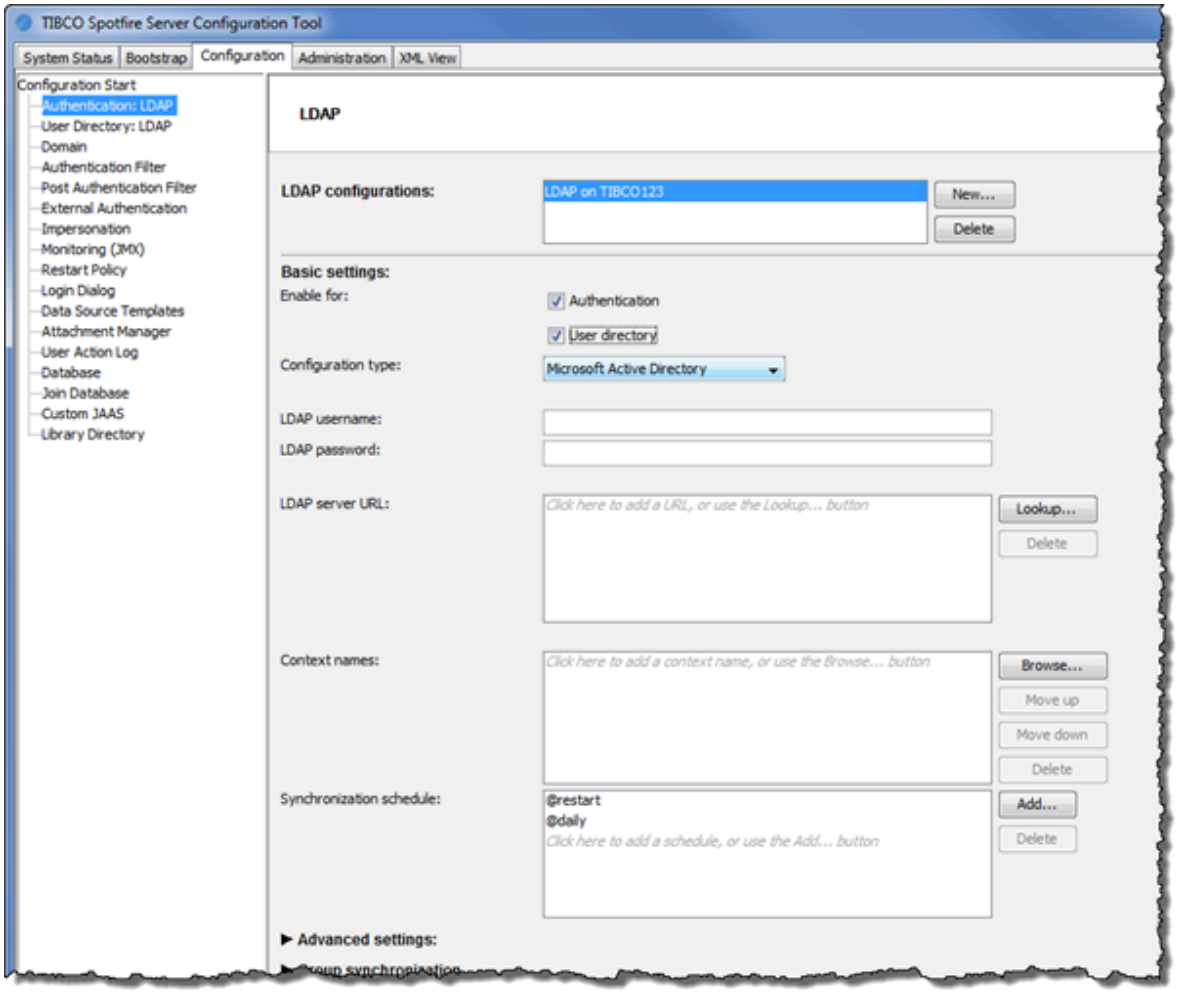
   The **User directory** field switches to **LDAP** along with the **Authentication** field. This is because in most cases it is recommended that LDAP authentication be paired with the user directory in LDAP mode.

   If your LDAP directory contains a very large number of users that are not divided into convenient sub-units (contexts), you may want to use the Spotfire database user directory instead. In this configuration, only users who log in to Spotfire Server are included in the user directory, so there are fewer users for Spotfire Server to track.

2. In the left panel of the page, click **Authentication: LDAP**, and then click **New**.



3. In the Create configuration dialog, enter a name for your LDAP configuration, for example "LDAP on TIBCO123", and then click **OK**.
   The LDAP configuration page is displayed.



4. Next to **Enable for**, select both the **Authentication** and **User directory** check boxes. This instructs Spotfire Server to create a user account in the Spotfire database for each user (within the configured scope) in the LDAP directory. When someone tries to log in to the Spotfire system, Spotfire Server accesses their account and then validates their password through the LDAP directory.
5. Next to **LDAP username** and **LDAP password**, enter the user name and password of an LDAP service account with read access to Active Directory.
6. Next to **LDAP server URL**, enter the URL in the form LDAP://server/:port, for example LDAP://computer1.TIBCO.com:389
7. Next to **Context names**, enter the contexts you want to synchronize.
8. Next to **Synchronization schedule** you can change the scheduled synchronization times between the LDAP directory and the Spotfire database. The default is to synchronize whenever Spotfire Server is restarted, in addition to daily. For additional synchronization options, click **Add**.
9. Click **Test connection** to verify your entries.
10. If you set the user directory to **Database** in step 1 above, click **Post Authentication Filter** in the left panel and then, next to **Default filter mode**, select **Auto-create**.
    When users log in to Spotfire Server they are added to the Spotfire user directory.
11. When you're finished, click **Save configuration**.

Configuring LDAPS

In an LDAP environment, where the Spotfire system communicates with an LDAP directory server, administrators often secure the LDAP protocol using TLS, if the LDAP directory supports this.

Prerequisites

- The LDAP directory server has been set up to communicate using TLS.

There are three different alternatives when it comes to working with LDAPS certificates.

Commercial certificates

Alternative 1:

If you are using commercial certificates, then Java most likely trusts them already and you do not need any further configuration.

Self-signed certificates

If you are using self-signed certificates with Spotfire Server, each certificate can have its own keystore file to handle trust for the SSL/TLS communication. The keystore files are either stored in the `tomcat/certs` directory (of each computer in the cluster) where they are automatically copied and kept during server upgrades (Alternative 2), or in the default Java trust store (Alternative 3).

All certificates used for LDAPS in the `tomcat/certs` directory must have the same password. The standard password for the Java `cacert` file can still be the default ("changeit") but if you choose to change the password, it should be the same one as used for the trust files in `tomcat/certs`. To use another password than "changeit", the additional Java startup parameter `javax.net.ssl.trustStorePassword` should be added, either to the start script or to the service. See "Virtual memory modification".

Alternative 2 (preferred when using self-signed certificates):

For each certificate that is to be trusted, go to the directory `<installation dir>/tomcat/certs` and add a .jks file. This must be done on every Spotfire Server in the cluster. Name the files with a descriptive name.

Alternative 3:

The default keystore provided by Java is located in `<installation dir>/jdk/lib/security/cacerts`. The default password for the included trust files is "changeit" (without quotation marks). This file can be modified with additional certificates.

This file will not be copied during upgrades.

To add certificates to the Java trust store, open a command-line interface, navigate to the `<installation dir>/jdk/lib/security` directory, and run the following keytool command: `../../bin/keytool -import -file ldapserver.crt -keystore cacerts -alias spotfire_ldaps`. Replace `ldapserver.crt` with the name of the exported certificate. When prompted, enter the password to the `cacerts` keystore.

SASL authentication for LDAP

Spotfire Server supports two SASL (Simple Authentication Socket Layer) mechanisms for authentication towards LDAP: DIGEST-MD5 and GSSAPI.

These mechanisms can provide secure authentication of Spotfire Server when it is connecting to LDAP servers by preventing clear text passwords from being transmitted over the network.

GSSAPI can provide secure authentication even over un-secure networks because it uses the Kerberos protocol for authentication.

These instructions apply for Active Directory LDAP configurations. Spotfire Server does not support GSSAPI for other LDAP configurations.

Configuring Spotfire Server for DIGEST-MD5 authentication of LDAP

These instructions apply for Active Directory LDAP configurations. Spotfire Server does not support GSSAPI for other LDAP configurations.

Procedure

- When configuring SASL authentication with DIGEST-MD5, follow these guidelines:
  - The distinguished name (DN) does not work for authentication; the userPrincipalName attribute must be used instead.
  - Set the **authentication attribute** option to **userPrincipalName**.
  - Set the **username attribute** option to **sAMAccountName**.
  - All accounts must use reversible encryption for their passwords. This is typically not the default setting for Active Directory.

Configuring Spotfire Server for GSSAPI authentication of LDAP

These instructions apply for Active Directory LDAP configurations. Spotfire Server does not support GSSAPI for other LDAP configurations.

Prerequisites

- Make sure that you have a fully working Active Directory LDAP configuration using clear-text password authentication (also known as simple authentication mechanism).
- Save this fully working Active Directory LDAP configuration to file.
- Make a note of the LDAP configuration's ID.
- Make sure that you have a fully working krb5.conf file. The content of the krb5.conf file must be the same as when setting up Spotfire Server for Kerberos authentication. See Configuring Kerberos for Java.

Make sure to stop the entire service/Java process before installing the file. If the krb5.conf file is modified after Spotfire Server has been started, you must restart the Spotfire Server process for the modifications to take effect.

Procedure

1. Stop Spotfire Server (see Start or stop Spotfire Server).
2. Copy the fully working krb5.conf file to the `<install dir>/tomcat/spotfire-config` directory on each Spotfire Server in the cluster.
3. Open the configuration tool and go to the LDAP Configuration panel.
4. Update the LDAP user name so that it is a proper Kerberos principal name. Usually it is sufficient to add the name of the account's Windows domain in upper-case letters. Sometimes it is also necessary

to include the Windows domain name. Using a name based on a distinguished name (DN) or including a NetBIOS domain name does not work when using GSSAPI.
Examples of correct names:

- ldapsvc@ RESEARCH.EXAMPLE.COM
- ldapsvc@research.example.com@ RESEARCH.EXAMPLE.COM

5. Select the specific LDAP configuration to be enabled for GSSAPI and then expand the **Advanced** settings.
6. In the **Advanced** dialog, make the following changes:
   a) Set the **security-authentication** configuration property to **GSSAPI**.
   b) Set the **authentication-attribute** to **sAMAccountName** or **userPrincipalName** (whichever works best for your configuration). The default value is empty.

      If the `krb5.conf` file contains more than one Kerberos realm, the authentication-attribute must be set to **userPrincipalName**.
   c) Add a custom property with the key kerberos.login.context.name and the value **SpotfireGSSAPI**.
7. Click **Save configuration**.
8. Restart Spotfire Server.

What to do next

Procedure steps related to LDAP configurations must be performed for each LDAP catalogue that you want to enable for GSSAPI. For multiple LDAP configurations, repeat these steps for each configuration.

Authentication towards Windows NT Domain (legacy)

With this authentication method, user authentication is delegated to Windows NT domain controllers.

Spotfire Server must be installed on a computer running Windows and there must be a working Windows NT 4 Server domain controller or a Windows Server 2000 or later domain controller running in mixed mode. This is a legacy solution that should only be used if LDAP cannot be used.

The Windows NT Domain authentication method can be combined with a user directory in either Windows NT Domain mode or in Spotfire database mode.

When combining this authentication method with a Spotfire database user directory mode, the post-authentication filter must be configured for auto-creating mode, so that the users will be automatically added to the user directory. When combining it with a Windows NT Domain User Directory, the default blocking post-authentication filter is already correctly configured.

Combination of LDAP and Spotfire database authentication

If you configure authentication towards an external user directory such as an LDAP directory, or a Windows NT Domain, you can combine this with adding users manually to the Spotfire database.

This feature allow users to access Spotfire even though they are not part of the external user directory. For example, there may be temporary users that you do not want to add to the LDAP directory, or you may want to ensure that administrators can access Spotfire if the connection to the LDAP directory is lost. These users will be added to the same domain as the users created in Spotfire.

This feature is enabled by default. For information on how to disable this feature, see Preventing administrators from adding local users when using LDAP.

If you switch from Spotfire database authentication to LDAP authentication, all users remaining in the Spotfire database will still have access to Spotfire.

Preventing administrators from adding local users when using LDAP

You can prevent administrator from adding local users to the Spotfire database when authenticating towards an external directory.

Procedure

1. Open a command line and export the active server configuration by using the export-config command; for additional information, see Executing commands on the command line.
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=user-directory.allow-database-user-creation --value=false
   ```

   For information on the command, see set-config-prop.

   To enable the feature again, run the same command but set the value to "true".
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server.

Authentication towards a custom JAAS module

All the user name and password authentication methods that are supported by Spotfire Server are implemented as Java Authentication and Authorization Service (JAAS) modules. Spotfire also supports third-party JAAS modules.

You may therefore use a custom JAAS module, provided that it does the following:

- Validates user name and password authentication.
- Uses JAAS' `NameCallback` and `PasswordCallback` objects for collecting the user names and passwords.

When using a custom JAAS module, you must place the jar file in the `<installation dir>/tomcat/custom-ext` directory on all Spotfire Servers.

For more information about JAAS, consult the JAAS Reference Guide.

Single sign-on authentication methods

Spotfire Server can be integrated with certain single sign-on systems that are used in enterprise environments.

Spotfire Server can use the NTLM or Kerberos single sign-on authentication methods, where the identity information stored within the user's current Windows session is reused to authenticate the user on the server. Thus, when using these authentication methods, users are never prompted for user name or password when they log in to Spotfire Server. The Kerberos and NTLM authentication methods are commonly referred to as Integrated Windows Authentication.

Spotfire Server can also authenticate users based on X.509 certificates. This requires the server to be configured for mutual TLS, meaning HTTPS with X.509 client certificates.

NTLM authentication

The NTLM authentication method reuses the identity information associated with the user's current Windows session. This identity information is gathered when the user initially logs in to Windows.

When both the client computer and the server computer belong to the same Windows domain or two separate Windows domains with established trust between them, this can provide a single sign-on experience.

If the client computer belongs to a separate Windows domain (without trust established to the server computer's domain), the current Windows session is not valid in the Windows domain of the server computer and the user will be prompted for user name and password. The user must then enter the user name and password of a valid account that belongs to the Windows domain of the server computer.

It is not possible to delegate NTLM authentication; Spotfire Server can not reuse the authentication credentials presented by the client, for example when authenticating against an Information Services data source that also uses NTLM. If you need such functionality, use Kerberos instead.

The NTLM authentication method can be combined with a user directory of either type:

- LDAP (recommended)
- Spotfire database, provided that the default post-authentication filter is configured in auto-creating mode

The following instructions assume that either combination of authentication and user directory is already fully working.

Setting up NTLM authentication involves two steps:

Creating a computer service account in your Windows domain

Configuring NTLM authentication

Creating a computer service account in your Windows domain

To set up NTLM authentication, you first create a computer service account by running a Visual Basic script that is distributed with Spotfire Server.

Alternatively, you can create the computer account manually; see Creating a computer service account manually.

Prerequisites

- The script must be run on a Windows computer, but does not have to be run on the same computer that the server is installed on.
- You must be logged in to your Windows domain as a member of the group Account Operators or Administrators to run the SetupWizard.vbs script.
- If Spotfire Server is installed on a Linux computer, copy the SetupWizard.vbs script to a Windows computer first.

Make sure to create a new computer account. A user account will not work. Reusing an existing computer account will not work either.

Procedure

1. Double-click the following file: `<installation dir>/tomcat/spotfire-bin/setupwizard.vbs`.
2. In the **Domain Controller Hostname** panel, enter the hostname of one of your domain controllers. Click **OK**.
3. In the **Account Name** panel, enter the short name of the computer account to be created. The short name must not exceed 15 characters. Click **OK**.
4. In the **Distinguished Name** panel, enter a distinguished name for the account to be created. We suggest that you use a distinguished name that is based on the short name entered in the previous

panel. You should edit this to match your Windows domain, with regards to parameters such as in which Organizational Units (OU) the account should be placed. Click **OK**.

5. In the **Account Password** panel, enter a password for the account to be created. Click **OK**.
   A dialog opens with text indicating if the tool was successful. Click **OK**.

   If the tool was unsuccessful, make sure that the logged in user has the required permissions to create accounts in the Windows Domain, and that the Domain Controller can be reached.

6. The file `SetupWizard.txt`, created by the tool in the folder where the tool is located, opens. If it does not, open it manually. The information in the file is required to run the NTLM authentication configuration commands.

Example of a SetupWizard.txt file

```
# Generated by the Jespa Setup Wizard from IOPLEX Software on 2011-04-07

jespa.bindstr = dc.example.research.com
jespa.dns.servers = 192.168.0.1
jespa.dns.site = Default-First-Site-Name
jespa.service.acctname = jespa-svc$@dc.example.research.com
jespa.service.password = Pa33w0rd
```

What to do next

Configure NTLM authentication using configuration commands

Creating a computer service account manually

If you are setting up NTLM authentication and you are unable to run the `SetupWizard.vbs` script, or you prefer to create the account manually, follow these steps.

Prerequisites

If Spotfire Server is installed on a Linux computer, copy the `SetComputerPassword.vbs` script to a Windows computer first.

Procedure

1. Create the computer account by using the Microsoft Management Console snap-in Domain Users and Computers. Refer to Microsoft documentation for details on how to use this tool.

   Make sure to create a new computer account. A user account will not work. Reusing an existing computer account will not work.

2. To set a password for this account, open a command line and run this script with the account name and password as arguments to the command: `<install-dir>/tomcat/spotfire-bin/SetComputerPassword.vbs`.
   Example:

   ```
   SetComputerPassword.vbs jespa-svc$@dc.example.research.com Pa33w0rd
   ```

What to do next

Configure NTLM authentication using configuration commands

Configuring NTLM authentication for a single server

These instructions are for configuring NTLM authentication by using the command line.

Prerequisites

You have created a computer service account; see Creating a computer service account in your Windows domain.

Procedure

1. Configure NTLM authentication by using the following commands: config-ntlm-auth and list-ntlm-auth.

   This is the information you must have to run the commands:

Server (optional)	The name of the server instance to which the specified configuration options belong. If no server name is specified, then all parameters will be shared, applying to all servers in the cluster. It is common to use server-specific values for the account name and password configuration options.
Account name (required)	Specifies the fully qualified name of the Active Directory computer account that is to be used by the NTLM authentication service. This account must be a proper computer account, created solely for the purpose of running the NTLM authentication service. It can neither be an ordinary user account, nor an account of an existing computer. Note that the local part of an Active Directory computer account name always ends with a dollar sign, and the local part of the account name (excluding the dollar sign) must not exceed 15 characters. Example: ntlm-svc$@research.example.com
Password (required)	Specifies the password for the computer account used by the NTLM authentication service.
DNS domain name (optional)	The DNS name of the Windows domain to which the Spotfire Server computer belongs. The specified domain name is automatically resolved into a domain controller hostname. As an alternative to specifying a DNS domain name, it is also possible to specify a domain controller hostname directly. The DNS domain name is recommended because you then automatically get the benefits of fail-over and load-balancing, provided that you have more than one domain controller. The DNS domain name and domain controller arguments are mutually exclusive. Example: research.example.com
Domain controller (optional)	The DNS hostname of an Active Directory domain controller. It is recommended that the DNS domain name option be used instead because that option gives the benefits of fail-over and load-balancing. The domain controller and DNS domain name arguments are mutually exclusive. Example: dc01.research.example.com
DNS servers (optional)	A comma-separated list of IP addresses of the DNS servers associated with the Windows domain. When no DNS servers are specified, the server will fall back to use the server computer's default DNS server configuration. Example: 192.168.1.1,192.168.1.2
AD site (optional)	Specifies the Active Directory site where the Spotfire system is located. Specifying an Active Directory site can potentially increase performance because the NTLM authentication service will then only communicate with the local Windows domain controllers. Example: VIENNA
DNS cache TTL (optional)	Specifies how long (in milliseconds) name server lookups should be cached. The default value is 5000 ms.

Connection ID header name (optional)	This parameter specifies the name of an HTTP header containing unique connection IDs in environments where the server is located behind a proxy or load-balancer that does not properly provide the server with the client's IP address. The specified HTTP header must contain unique connection IDs for each client connection and is thus typically based on the client's IP address together with the connection's port number on the client side.

2. Import the configuration using the config-auth command and restart the server to activate the NTLM single sign-on authentication method.

Kerberos authentication

Kerberos is a protocol that allows for secure authentication even over unsecure networks. It can be difficult to set up, but after it is fully working you have a very secure authentication system with the benefits of single sign-on.

It is usually a good idea to first create a working setup where the server uses username and password/LDAP authentication and a user directory in LDAP mode, and then proceed with switching from username and password/LDAP to Kerberos.

Setting up Kerberos authentication on Spotfire Server

If you intend to use the Kerberos authentication method on your system, the first thing you must do is to set up Spotfire Server to use Kerberos.

The following steps are required to configure Spotfire Server for the Kerberos authentication method. Steps 1-3 are performed as a Domain Administrator. Steps 4-7 are performed in Spotfire Server. See step 1 for a list of the prerequisites.

Creating a Kerberos service account

Creating a Kerberos service account is the first step in configuring Spotfire Server for the Kerberos authentication method.

Prerequisites

- Windows Domain Controllers running Windows Server 2008 or later.
- A computer with the Microsoft Active Directory Users and Computers MMC snap-in.
- A computer with the Microsoft Support Tools installed.
- A domain administrator account or a user account which is a member of the built-in Account Operators domain group, or any account with equivalent permissions.
- Windows Domain accounts for all Spotfire users.
- A fully-working user directory, with either of the following options:
  - LDAP (recommended)
  - Spotfire database, provided that the built-in post-authentication filter is auto-creating new users.

Procedure

1. Log in to the computer as a domain administrator or a user who is a member of the built-in Account Operators domain group.
2. Open the Active Directory Users and Computers MMC snap-in.

3. Create an ordinary user account with the following properties:

   - Use the same identifier in the **Full name** and **User logon name** (pre-Windows 2000) fields.

     Use only lowercase characters and make sure that there are no spaces in these fields.

   - Select the **Password never expires** check box.
   - Clear the **User must change password at next logon** check box.
   - If you want to use the crypto algorithm `aes128-sha1` or `aes256-sha1` the account option **This account supports Kerberos AES 128 bit encryption** or **This account supports Kerberos AES 256 bit encryption** must also be selected.

Registering Service Principal Names

Registering Service Principal Names (SPN) is the second step in configuring Spotfire Server for the Kerberos authentication method.

Procedure

1. Log in to the computer as a domain administrator or a user who is a member of the built-in Account Operators domain group.
2. From the Microsoft Support Tools package, use the **`setspn.exe`** command-line tool to register two SPNs for the Kerberos service account:

   - Execute the following two commands, replacing the variables as indicated in the table below the commands:

     ```
     > setspn -S HTTP/<fully qualified hostname>[:<port>] <service account name>
     > setspn -S HTTP/<hostname>[:<port>] <service account name>
     ```

   If the Spotfire Server is not listening on the default HTTP port 80 or the default HTTPS port 443, you should execute the `setspn` commands both with and without the port specified:

   ```
   > setspn -S HTTP/<fully qualified hostname>[:<port>] <service account name>
   > setspn -S HTTP/<hostname>[:<port>] <service account name>
   > setspn -S HTTP/<fully qualified hostname> <service account name>
   > setspn -S HTTP/<hostname> <service account name>
   ```

Variable	Description
fully qualified hostname	The fully qualified DNS hostname of the computer hosting Spotfire Server (in lowercase characters).
hostname	The short DNS hostname, without domain suffix, of the computer hosting Spotfire Server (in lowercase characters).
service account name	The user login name of the previously created Kerberos service account (in lowercase characters).

Variable	Description
port	The TCP port number on which Spotfire Server is listening. This is not required if using the default HTTP port 80 or the default HTTPS port 443.

You must use the name of a DNS A record for Spotfire Server. A CNAME record will not work.

Avoid explicitly specifying the port number if Spotfire Server is using the default HTTP port 80.

It is recommended that you not have multiple Kerberos-enabled HTTP services on one computer.

Registering Service Principal Names for the "spotsvc" Kerberos service account to be used by a Spotfire Server installed on the "spotfireserver.research.example.com" computer and listening on the default HTTP port 80 or the default HTTPS port 443:

```
> setspn -S HTTP/spotfireserver.research.example.com spotsvc
> setspn -S HTTP/spotfireserver spotsvc
```

This creates the following two SPNs for the "spotsvc" service account:

- HTTP/spotfireserver.research.example.com
- HTTP/spotfireserver

To list the resulting Service Principal Names for a Kerberos service account, execute the following command:

```
> setspn -L <service account name>
```

For example, for the "spotsvc" Kerberos service account, the previous command looks like this:

```
> setspn -L spotsvc
```

Creating a keytab file for the Kerberos service account

Creating the keytab file is the third step in configuring Spotfire Server for the Kerberos authentication method.

Procedure

1. Log in to the computer as a domain administrator or a user who is a member of the built-in Account Operators domain group.
2. Execute the following command, replacing the variables with the appropriate values:

```
> ktpass /princ HTTP/<fully qualified hostname>[:<port>]@<realm> /ptype krb5_nt_principal
 /crypto <crypto algorithm> /mapuser <service account name> /out spotfire.keytab -kvno 0
```

```
 /pass <service account password>
```

Make sure that the executed command does not have any newlines.

All values are case sensitive.

Older versions of the `ktpass.exe` tool will fail to create the keytab file when the tool is not run on an actual domain controller.

Variable	Description
fully qualified hostname	The fully qualified DNS hostname of the computer hosting Spotfire Server, which must exactly match the fully qualified hostname used when registering the SPNs (in lowercase characters).
port	The TCP port number on which Spotfire Server is listening (only specified if the port number was explicitly included in the registered Service Principal Names (SPN)). This is not required if using the default HTTP port 80 or the default HTTPS port 443.
realm	The name of the Kerberos realm, which is the DNS domain name written in uppercase characters.
crypto algorithm	Can be one of `aes128-sha1` or `aes256-sha1`. Make sure that the selected crypto algorithm is also specified in the `krb5.conf` file.
service account name	The user login name of the service account with the registered SPNs (written in lowercase characters).
service account password	The password for the service account.

If you change the password of the Kerberos service account, you must re-create the keytab file.

It is not critical to use the name "spotfire.keytab" for the keytab file, but the following instructions assume that this name is used.

Creating a keytab file for the "spotsvc" Kerberos service account in the "research.example.com" domain for Spotfire Server listening on the default HTTP port 80, or the default HTTPS port 443 on the "spotserver.research.example.com" computer:

```
> ktpass /princ HTTP/
spotfireserver.research.example.com@RESEARCH.EXAMPLE.COM
 /ptype krb5_nt_principal /crypto aes128-sha1 /mapuser spotsvc /out
 spotfire.keytab -kvno 0
 /pass spotsvcpassword
```

Creating a keytab file for the "spotsvc" Kerberos service account in the "research.example.com" domain for Spotfire Server listening on the HTTP port 8080 on the "spotserver.research.example.com" computer:

```
> ktpass /princ HTTP/
spotfireserver.research.example.com:8080@RESEARCH.EXAMPLE.COM
 /ptype krb5_nt_principal /crypto aes128-sha1 /mapuser spotsvc
 /out spotfire.keytab -kvno 0 /pass spotsvcpassword
```

Configuring Kerberos for Java

Configuring Kerberos for Java by editing the `krb5.conf` file is the fourth step in configuring Spotfire Server for the Kerberos authentication method.

Procedure

1. Open the following file in a text editor: `<server installation dir>\tomcat\spotfire-config\krb5.conf`.
2. Edit the following values to reflect your environment:

   The arguments are case sensitive.

   For more information, see Krb5.conf file.

   - MYDOMAIN: The name of the Kerberos realm, usually the same as the name of the Windows Domain, written in uppercase characters.
   - mydomain: The name of the Windows Domain, written in lowercase characters.
   - mydc: The name of the domain controller, written in lowercase characters.

   Example: Configuring Kerberos for Java in the "research.example.com" domain, with the two domain controllers "dc01.research.example.com" and "dc02.research.example.com":

   ```
   [libdefaults]
       default_realm = RESEARCH.EXAMPLE.COM
       default_keytab_name = spotfire.keytab
       default_tkt_enctypes = aes128-cts
       default_tgs_enctypes = aes128-cts
       forwardable = true

   [realms]
       RESEARCH.EXAMPLE.COM = {
           kdc = dc01.research.example.com
           kdc = dc02.research.example.com
           admin_server = dc01.research.example.com
           default_domain = research.example.com
       }

   [domain_realm]
       .research.example.com = RESEARCH.EXAMPLE.COM
       research.example.com = RESEARCH.EXAMPLE.COM

   [appdefaults]
       autologin = true
       forward = true
       forwardable = true
       encrypt = true
   ```

3. Optional: If you want to use the crypto algorithm aes256-sha1, perform the following tasks:

   a) Add `aes256-cts` as the first option in `default_tkt_enctypes` and `default_tgs_enctypes`.

Copying the Kerberos service account's keytab file to Spotfire Server

Copying the keytab file to Spotfire Server is the fifth step in configuring Spotfire Server for the Kerberos authentication method.

Procedure

1. Copy the `spotfire.keytab` file to the directory `<installation dir>\tomcat\spotfire-config` (Windows) or `<installation dir>/tomcat/spotfire-config` (Linux) in Spotfire Server.

   Because this file contains sensitive information, it must be handled with care. The file must not under any circumstances be readable by unauthorized users.

   To list the contents of the keytab file, use the **`klist`** command-line tool. It lists the principal name, crypto algorithm, and security credentials. The tool is included in the bundled JDK and is only available when installed on Windows:

   ```
   > <installation dir>\tomcat\spotfire-bin\klist.bat -k -t -e -K <keytab file>
   ```

   To test the keytab file, use the **`kinit`** command-line tool which is also included in the bundled JDK on Windows platforms:

   ```
   > <installation dir>\tomcat\spotfire-bin\kinit.bat -k -t <keytab file> HTTP/<fully
    qualified hostname>[:<port>]@<realm>
   ```

   If the keytab file is correctly set up, a ticket cache file is created in the logged-in user's home directory. It can typically be found in the path `C:\Users\<user>\krb5cc_<user>`.

2. As soon as you have verified that the ticket cache was created, you must delete the ticket cache file to prevent future problems.

Using Kerberos authentication with delegated credentials

Users can authenticate to different data sources using single sign-on login information. The server can delegate the user authentication to the data source, either through Information Services, or through a connector. This is possible only if you use the Kerberos single sign-on method.

If you are using a JDBC driver that supports passing the delegated user's Generic Security Standard (GSS) credentials through a connection property, then you can use constrained delegation with Information Services.

To enable constrained delegation for these drivers, add the following connection property to the corresponding Data Source Template.

```
<connection-property>
     <key>spotfire.kerberos.gsscredential.property</key>
     <value>connectionPropertyName</value>
   </connection-property>
```

Where *`connectionPropertyName`* is driver-specific. (Refer to your driver's documentation for more information.)

Prerequisites

For delegation to work, no client user account in the domain can have the setting **Account is sensitive and cannot be delegated**. By default, this setting is not enabled.

Procedure

1. Set up Kerberos authentication as described in Kerberos authentication. Make sure that users can log in with this method.

2. Grant the right to delegate client credentials to the Spotfire Server service account that is used for client authentication.

   Only the specified accounts can be delegated by the service account.

   - If possible, grant constrained delegation rights to the service account; see Enabling constrained delegation.
   - If you cannot use constrained delegation, grant unconstrained delegation rights. See the following topics for more information.
     - Enabling unconstrained delegation for an account on a domain controller in Windows 2000 mixed or native mode.
     - Enabling unconstrained delegation on a domain controller in Windows Server 2003 mode.

   The default delegation policy is "REQUIRE". This means that if Spotfire Server cannot delegate end user credentials, end users will not be able to open analyses in the web client. Prior to Spotfire version 7.7, the default delegation policy was "TRY", which would open analyses using impersonation if delegation failed.

Enabling constrained delegation

This is the second step in the process of setting up Kerberos authentication with delegated credentials for your Spotfire implementation. It allows the Spotfire Server to delegate user credentials to nodes.

Procedure

1. On the domain controller, go to **Administrative Tools**.
2. Select **Active Directory Users and Computers**.
3. Locate the Spotfire Server service account.
4. To open the account properties, right-click the account name and then click **Properties**.
5. On the **Delegation** tab, select **Trust this user for delegation to specified services only**.

   The **Delegation** tab is visible only for accounts to which SPNs are mapped.

6. Select **Use any authentication protocol**, and then click **Add**.
7. Click **Users or Computers** and select each user account or machine account that runs the node manager service on your nodes.

   If the node manager services are run by user accounts, you must first register SPNs for these. See Setting up Kerberos authentication on nodes.

8. Select the **http** service for each account, and then click **OK**.
9. Click **Apply**.

What to do next

Enabling constrained delegation on nodes

Enabling unconstrained delegation on a domain controller in Windows Server 2003 mode

This is the second step in the process of setting up Kerberos authentication with delegated credentials for your Spotfire implementation.

Procedure

1. On the domain controller, select **Start** > **Programs** > **Administrative Tools**.

2. Select **Active Directory Users and Computers**.
3. Locate the Spotfire Server service account.
4. To open the account properties, right-click the account name and then click **Properties**.
5. On the **Delegation** tab, select **Trust this user for delegation to any service (Kerberos only)**.

The **Delegation** tab is visible only for accounts to which SPNs are mapped.

6. Click **Apply**.

What to do next

Creating an Information Services data source template using Kerberos login

Enabling unconstrained delegation for an account on a domain controller in Windows 2000 mixed or native mode

This is the second step in the process of setting up Kerberos authentication with delegated credentials for your Spotfire implementation.

Procedure

1. On the domain controller, select **Start** > **Programs** > **Administrative Tools**.
2. Select **Active Directory Users and Computers**.
3. Locate the Spotfire Server service account.
4. To open the account properties, right-click the account name and then click **Properties**.
5. On the **Account** tab, in the **Account Options** list, select **Account is trusted for delegation**.
6. Click **Apply**.

What to do next

Creating an Information Services data source template using Kerberos login

Selecting Kerberos as the Spotfire login method

Selecting Kerberos as the Spotfire login method is the sixth step in configuring Spotfire Server for the Kerberos authentication method. You can use the configuration tool, or use the command line as detailed in this procedure.

Procedure

1. Execute the config-kerberos-auth command. The command takes the following two parameters:
   - Keytab file: The fully qualified path to the spotfire.keytab file. If the keytab file is named "spotfire.keytab" and has been copied to the recommended directory, the default path `${catalina.base}/spotfire-config/spotfire.keytab` is already correct. The shorthand `${catalina.base}` refers to the directory `<installation dir>\tomcat` (Windows) or `<installation dir>/tomcat` (Linux).
   - Service Principal Name: Specify the same Service Principal Name that was used when creating the keytab file. Example: HTTP/spotfireserver.research.example.com
2. Use the config-auth command to activate the Kerberos SSO authentication method.
3. Import the configuration and restart the server for the changes to take effect.

Disabling the username and password fields in the Spotfire Analyst login dialog

Because the Kerberos authentication method provides single sign-on capabilities, there is no need to prompt the end user for user name and password in the Spotfire Analyst login dialog.

This step is optional.

Procedure

1. Open a command line and export the active configuration (the `configuration.xml` file) by using the export-config command; for additional information, see Executing commands on the command line.
2. Execute the config-login-dialog command:

```
> config config-login-dialog --allow-user-provided-credentials=false
```

3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server service.

If you are using the configuration tool, on the Login dialog page, next to **Allow user provided credentials**, click **No**.

Kerberos authentication for clustered servers with load balancer

In a clustered environment where Kerberos authentication is used to authenticate users, the load balancer forwards all Kerberos authentication information to the Spotfire Servers. No configuration on the load balancer is needed, but there are certain considerations to take into account when Kerberos authentication is set up.

These are the special considerations:

- Two Service Principal Names must be created for each Spotfire Server as well as for the load balancer.
- One keytab file must be created. This must use the fully qualified Service Principal Name of the load balancer.
- This keytab file must be copied to each Spotfire Server.
- When Kerberos authentication is set up, the fully qualified Service Principal Name of the load balancer must be provided.

For general information about Spotfire Server clusters, see Clustered server deployments.

Setting up Kerberos authentication on nodes

After setting up Kerberos authentication on Spotfire Server, you must set it up for the nodes in your environment.

If you use Kerberos delegation, your Spotfire Server and Node Managers must be installed on different computers.

The account used to run the node manager service must be trusted for delegation, and you might need to register Service Principal Names (SPN) for that account. Also, all web client users must be given permission to modify the node manager services folder.

- If the node manager service is run using the local machine account, open the Active Directory Users and Computers MMC snap-in, select the machine account, and then select **Trust this computer for delegation to any service**.

- If the node manager service is run using a specified user account, open the Active Directory Users and Computers MMC snap-in, select the user account, and then select **Trust this user for delegation to any service**.

  If the node manager service is run using a specified user account, you must also register Service Principal Names (SPN) for that account.

```
> setspn  -S HTTP/<fully qualified node hostname>[:<port>] <node service account name>
> setspn  -S HTTP/<node hostname>[:<port>] <node service account name>
```

For information on how to register SPNs, see Registering Service Principal Names.

All web client user accounts must be given permission to modify the folder `nm\services`. This permission allows the delegated users to read, write, and delete temp files.

As of Spotfire version 7.7, the default delegation policy is "REQUIRE". This means that if Spotfire Server cannot delegate end user credentials, users will not be able to open analyses in the web client. Prior to this, the default delegation policy was to open analyses using impersonation if delegation failed. For details on this option, see config-kerberos-auth.

If Spotfire Connectors are used for the Web Player service, all delegated web client users must also have access to the applicable connector drivers.

Enabling constrained delegation on nodes

You must enable constrained delegation for your nodes. It allows the service on the node to delegate user credentials to the Spotfire Server and access external resources.

Prerequisites

You have enabled constrained delegation on Spotfire Server. See Enabling constrained delegation.

Procedure

1. On the domain controller, go to **Administrative Tools**.
2. Select **Active Directory Users and Computers**.
3. Locate the machine accounts or user accounts that runs the node manager services.

   Steps 4 through 11 must be performed for each account that runs a node manager service.
4. To open the account properties, right-click the account name and then click **Properties**.
5. On the **Delegation** tab, select **Trust this user for delegation to specified services only**.

   The **Delegation** tab is visible only for accounts to which SPNs are mapped. If the node manager services are run by user accounts, you must first register SPNs for these. See Setting up Kerberos authentication on nodes.
6. Select **Use any authentication protocol**, and then click **Add**.
7. Click **Users or Computers** and select any Spotfire Server service account.
8. Select the **http** service for each Spotfire Server service account, and then click **OK**.
9. Click **Users or Computers** and select any machine account or service account for a computer running the external resource you want to delegate to.
10. Select the applicable services for each account, and then click **OK**.

    For example the **MSSQLSvc** service for delegation to a Microsoft SQL Server or the **CIFS** service for delegation to a file share.
11. Click **Apply**.

Enable Kerberos authentication for end-users

If you use Kerberos authentication, it must be enabled in the browsers of all end-user computers.

This is applicable for all users accessing Spotfire Server, either from a browser, or Spotfire Analyst.

Enabling Kerberos for Internet Explorer and Spotfire Analyst

Follow these steps on every computer using Internet Explorer or Spotfire Analyst.

Procedure

1. Go to **Tools > Internet Options > Advanced** and select **Enable Integrated Windows Authentication (Requires Restart)**.
2. The Spotfire Server you are connecting to must be located in the **Intranet** security zone.

If the website is located in the **Internet** security zone, Internet Explorer will not even attempt Kerberos authentication. This is because in most **Internet** scenarios a connection with a domain controller can not be established. The simple rule is that any URL that contains periods, such as an IP address or Fully Qualified Domain Name (FQDN), is in the **Internet** zone. If you are connecting to an IP address or FQDN, you can use the settings in Internet Explorer or Group Policy to add this site to the **Intranet** security zone. For more information on how Internet Explorer evaluates the zone of a resource, see the Microsoft Knowledge Base article KB 258063.

If a client accesses a server belonging to another trusted domain, that server must be added to the **Local Intranet** zone, found under **Internet Options** > **Security** > **Local Intranet**. Without this setting, Internet Explorer, or Spotfire Analyst will not be able to authenticate using Kerberos.

For example, if the client `client.emea.example.com` accesses the server `server.na.example.com`, then `server.na.example.com` must be added to the **Local Intranet** zone.

Enabling delegated Kerberos for Google Chrome

Follow these instructions on every computer using Google Chrome.

You must create and set a registry key for Google Chrome.

1. The Spotfire Server you are connecting to must be located in the **Intranet** security zone.
2. In the Registry Editor, go to `[HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Google\Chrome]`.
3. Add the String Value `AuthNegotiateDelegateWhitelist`.
4. Modify `AuthNegotiateDelegateWhitelist` and add the URL to the Spotfire Server.

For more information, see the Chromium Projects developer page at https://dev.chromium.org/administrators/policy-list-3#AuthNegotiateDelegateWhitelist.

Enabling Kerberos for Mozilla Firefox

Follow these steps on every computer using Mozilla Firefox.

Procedure

1. In the Firefox browser address box, type `about:config`.

2. For the following parameters, set the values to the Spotfire Server URL for which you want to activate Negotiate.
   - `network.negotiate-auth.delegation-uris`
   - `network.negotiate-auth.trusted-uris`

Using Kerberos to log in to the Spotfire database

To increase security in your Spotfire implementation, you might want to set up Spotfire Server to authenticate with the Spotfire database on Oracle and Microsoft SQL Server using the Kerberos protocol.

This only affects how the database connections are authenticated and is not required for Spotfire Analyst clients or web clients to connect to Spotfire Server using the Kerberos authentication method.

Prerequisites

- Windows Domain Controllers running Windows Server 2008 R2 or later.
- A computer with the Microsoft Active Directory Users and Computers MMC snap-in.
- A computer with the Microsoft Support Tools installed.
- A domain administrator account or a user account which is a member of the built-in Account Operators domain group, or any account with equivalent permissions.
- The database server must already be installed and configured for both Kerberos authentication and user name/password authentication.
- Microsoft Active Directory is used as Kerberos environment.
- If the database is an Oracle database, then download Oracle's latest JDBC driver (`ojdbc*.jar`) from Oracle's web page.
- If the database is a Microsoft SQL Server database, use the bundled Microsoft JDBC driver (`sqljdbc*.jar`). Version 4.0 of the `sqljdc*.jar` driver introduced the new authenticationScheme=JavaKerberos directive, which is required.

Procedure

1. Create a Windows domain account for the Spotfire database.
2. Create the Spotfire database.
   - If you are using SQL Server database: Edit and run the `create_databases_ia.bat` script. This creates a SQL Server database account and connects it to the previously created Windows domain account. For instructions, see Setting up the Spotfire database (SQL Server with Integrated Windows authentication).
   - If you are using Oracle database: Edit and run the `create_databases.bat` script. This will create a normal Oracle database account that authenticates with user name and password; for instructions on creating the database account, see Setting up the Spotfire database (Oracle).
3. Oracle database only: Configure the Spotfire database account to the Windows domain account.
4. Install Spotfire Server.
5. Install a vendor database driver; see Database drivers.
6. Configure Kerberos for Java.
7. Optional: Create a keytab file for the Kerberos service account.
8. Create a JAAS application configuration for the Spotfire database connection pool.

9. Register the JAAS application configuration file with Java.
10. Connect to the Spotfire database by running the bootstrap command or by using the configuration tool; see Configuring the database connection for Spotfire Server using Kerberos (Oracle) or Configuring the database connection for Spotfire Server using Kerberos (SQL Server).

Creating a Windows domain account for the Spotfire database

Creating a Windows domain account for the database is the first step in setting up Kerberos authentication for database connections.

Prerequisites

See Using Kerberos to log in to the Spotfire database for the list of prerequisites.

Procedure

1. Log in to Windows with one of the following accounts:
   - A domain administrator
   - A user who is a member of the built-in Account Operators domain group
   - A user with equivalent privileges
2. Launch the Active Directory Users and Computers MMC snap-in and create a normal user account with the following properties:
   - Use the same identifier in the **Full name**, **User logon name**, and **User logon name (pre-Windows 2000)** fields.

     Make sure to use only lowercase characters, and leave no spaces in these fields.
   - Select the **Password never expires** check box.
   - Clear the **User must change password at next logon** check box.
   - Recommended: Select the **Account is sensitive and cannot be delegated** check box.

What to do next

- SQL Server database: Edit and run the `create_databases_ia.bat` script. This creates a SQL Server database account and connects it to the previously created Windows domain account. For instructions, see Setting up the Spotfire database (SQL Server with Integrated Windows authentication).
- If you are using Oracle database: Edit and run the `create_databases.bat` script. This will create a normal Oracle database account that authenticates with user name and password; for instructions on creating the database account, see Setting up the Spotfire database (Oracle).

Configuring the Spotfire database account to the Windows domain account

If you are using an Oracle database, this is the third step in setting up Kerberos to log in to the Spotfire database.

Procedure

1. Log in to the Oracle database instance with SYSDBA privileges to manage accounts.
   Connecting to a database with connection identifier ORCL as sysdba

   ```
   sqlplus sys@ORCL as sysdba
   ```

2. Alter the Spotfire database account so that it is identified externally by running the following command:

```
SQL> alter user <SERVERDB_USER> identified externally as '<SERVERDB_USER>@
REALM>';
```

Replace <SERVERDB_USER> and <REALM> with the Spotfire database account name and the Kerberos realm. Make sure to use uppercase letters when specifying the Kerberos realm.

```
SQL> alter user spotuser identified externally as
'spotuser@RESEARCH.EXAMPLE.COM';
```

3. Test the Kerberos-enabled Spotfire database account by opening a command prompt running as the created Windows domain account. It should now be possible to connect to the database using the following command, assuming the connection identifier is ORCL: `> sqlplus /@ORCL`

It is assumed that Kerberos authentication is already set up for the Oracle client.

Keytab file for the Kerberos service account

There are several methods for creating the keytab file for the Kerberos service account.

Creating a keytab file for the Kerberos service account (using the ktpass command from Microsoft Support Tools)

This method of creating a keytab file uses the **`ktpass`** command that is included with Microsoft Support Tools.

Procedure

1. On a computer with the Microsoft Support Tools installed (it is not necessary to be logged in as a privileged user), execute the following command, replacing the `<database account name>`, `<REALM>`, `<crypto algorithm>`, and `<database account password>` with the appropriate values. `<crypto algorithm>` can be one of `aes128-sha1` or `aes256-sha1`. Make sure that the selected crypto algorithm is also specified in the `krb5.conf` file.

All values are case sensitive.

```
> ktpass /princ <database account name>@<REALM> /ptype krb5_nt_principal /
crypto <crypto algorithm> /out spotfire-database.keytab  -kvno 0 /pass <database account
 password>
```

It is not critical to use the name "spotfire-database.keytab" for the keytab file, but the following instructions assume that this name is used.

Example of creating a keytab file for the Spotfire database account named "spotuser" in the research.example.com domain:

```
> ktpass /princ spotuser@RESEARCH.EXAMPLE.COM /ptype krb5_nt_principal / crypto
aes128-sha1 /out spotfire-database.keytab -kvno 0 /pass spotuserpassword
```

2. Copy the `spotfire-database.keytab` file to the directory `<installation dir>\tomcat\spotfire-config` (Windows) or `<installation dir>/tomcat/spotfire-config` (Linux) in Spotfire Server.

Because this file contains sensitive information, it must be handled with care. The file must not under any circumstances be readable by unauthorized users.

If you change the password of the Kerberos service account, you must re-create the keytab file.

Creating a keytab file for the Kerberos service account (using the ktab command from the bundled JDK)

This method of creating a keytab file uses the **`ktab`** command that is included with the bundled JDK.

Procedure

1. On the computer runningSpotfire Server, open a command line as an administrator and change the directory to `<server installation dir>/tomcat/spotfire-bin`.
2. Run the following command, replacing the *<database account name>* with the user login name of the Spotfire database account, written in lowercase letters:

   ```
   > ktab -k spotfire-database.keytab -a <database account name>
   ```

   All values are case sensitive.

   It is not critical to use the name "spotfire-database.keytab" for the keytab file, but the following instructions assume that this name is used.

   The tool prompts you for the password of the service account.
3. Enter the password that you used when creating the Spotfire database account.
4. Verify the created keytab by running the `klist` and `kinit` utilities:

   ```
   > klist  -k spotfire-database.keytab
   > kinit  -k  -t spotfire-database.keytab <database account name>@<realm>
   ```

   If you change the password of the Kerberos service account, you must re-create the keytab file.

   Creating and verifying a keytab file for the "serverdb_user" Spotfire database account in the research.example.com domain:

   ```
   > ktab  -k spotfire-database.keytab  -a serverdb_user
   > klist  -k spotfire-database.keytab
   > kinit  -k  -t spotfire-database.keytab serverdb_user@RESEARCH.EXAMPLE.COM
   ```
5. Copy the `spotfire-database.keytab` file to the Spotfire Server directory `<installation dir>\tomcat\spotfire-config` (Windows) or `<installation dir>/tomcat/spotfire-config` (Linux).

   Because this file contains sensitive information, it must be handled with care. The file must not under any circumstances be readable by unauthorized users.

   If you change the password of the Kerberos service account, you must re-create the keytab file.

Creating a keytab file for the Kerberos service account (using the ktutil command on Linux)

This method of creating a keytab file on Linux uses the **`ktutil`** command.

Prerequisites

- Kerberos is installed on the Linux host where Spotfire Server is installed.
- The tools **`ktutil`**, **`klist`**, and **`kinit`** are available on the Linux host.

Procedure

1. Start the ktutil tool by invoking it from the command line without any arguments. Execute the commands below, replacing <database account name> with the user login name of the Spotfire database account, written in lowercase letters:

```
> ktutil
ktutil:  add_entry -password -p <database account name> -k 0 -e aes128-sha1
Password for <database account name>:
ktutil:  write_kt spotfire-database.keytab
ktutil:  quit
```

All values are case sensitive.

It is not critical to use the name "spotfire-database.keytab" for the keytab file, but the following instructions assume that this name is used.

The tool prompts you for the password of the service account.

2. Enter the password that you used when creating the Spotfire database account.
3. Verify the created keytab by running the `klist` and `kinit` utilities:

```
> klist  -k spotfire-database.keytab
> kinit  -k  -t spotfire-database.keytab <database account name>@<realm>
```

If you change the password of the Kerberos service account, you must re-create the keytab file.

Creating and verifying a keytab file for the "serverdb_user" Spotfire database account in the research.example.com domain:

```
> ktutil
ktutil:  add_entry -password -p serverdb_user -k 0 -e aes128-sha1
Password for serverdb_user:
ktutil:  write_kt spotfire-database.keytab
ktutil:  quit
> klist -k spotfire-database.keytab
> kinit -k -t spotfire-database.keytab serverdb_user@RESEARCH.EXAMPLE.COM
```

4. Copy the `spotfire-database.keytab` file to the following Spotfire Server directory: `<installation dir>/tomcat/spotfire-config`.

Because this file contains sensitive information, it must be handled with care. The file must not under any circumstances be readable by unauthorized users.

If you change the password of the Kerberos service account, you must re-create the keytab file.

Creating a JAAS application configuration for the Spotfire database connection pool

Follow these instructions to create a JAAS application configuration for the Spotfire database connection pool.

Procedure

1. Acquire a Kerberos ticket in one of the following ways, and name the file "`spotfire-database.login`":
   - By using a keytab file; see Acquiring a Kerberos ticket using a keytab file.
   - By using a username and password; see Acquiring a Kerberos ticket using a username and password.
   - By using the identity of the account running the Spotfire Server process; see
2. In Spotfire Server, create the file `<install directory>\tomcat\spotfire-config\spotfire-database.login` (Windows) or `<install directory>/tomcat/spotfire-config/spotfire-database.login` (Linux) and populate it with the `spotfire-database.login` file.

Acquiring a Kerberos ticket by using a keytab file

This method of acquiring a Kerberos ticket uses a keytab file.

Procedure

- In the following code, replace `<service account name>` and `<realm>` with the name of the Spotfire database account and the Kerberos realm.

  Use lowercase letters for the account name and uppercase letters for the realm name.

```
DatabaseKerberos
{
 com.sun.security.auth.module.Krb5LoginModule
  required
  debug=true
  storeKey=true
  useKeyTab=true
  keyTab="${catalina.base}/spotfire-config/spotfire-database.keytab"
  principal="<service account name>@<realm>";
};
```

Acquiring a Kerberos ticket by using a username and password

This method of acquiring a Kerberos ticket uses a username and password.

Procedure

- Use the following code:

```
DatabaseKerberos
{
 com.sun.security.auth.module.Krb5LoginModule
  required
  debug=true
  storeKey=true
  useKeyTab=false
  doNotPrompt=false;
};
```

Acquiring a Kerberos ticket by using the identity of the account running the Spotfire Server process

To make it possible to log in to the Spotfire database as the user currently running the server, the connection pool must be able to acquire the initial Ticket-Granting-Ticket (TGT) from the native Ticket Cache of the Spotfire Server host.

Procedure

- Modify the following registry key so that the TGT session can be exported:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\
Parameters]"allowtgtsessionkey"=dword:00000001

DatabaseKerberos
{
 com.sun.security.auth.module.Krb5LoginModule
  required
  debug=true
  storeKey=true
  useTicketCache=true
  doNotPrompt=false;
};
```

Registering the JAAS application configuration file with Java

After you have created the `spotfire-database.login` file, it must be registered in Java.

Procedure

- Open the file `<install directory>/jdk/conf/security/java.security` in a text editor and add the following lines to the end of the file:

```
# Register Java Authentication & Authorization Services (JAAS)
configurations
login.config.url.1=file:${catalina.base}/spotfire-config/spotfire-database.login
```

Configuring the database connection for Spotfire Server using Kerberos (Oracle)

If you use an Oracle database, follow these instructions to configure the database connection for Spotfire Server.

Procedure

- To bootstrap Spotfire Server, execute the following bootstrap command, replacing `<database-url>` with the JDBC connection URL.

When using a username and a password to request the Kerberos ticket, make sure to also specify the `-username` and `-password` arguments.

```
> config bootstrap --test -driver-class=oracle.jdbc.OracleDriver --database-url=<database
url> --kerberos-login-context=DatabaseKerberos -Coracle.net.authentication_services=
(KERBEROS5)
```

```
> config bootstrap --test --driver-class=oracle.jdbc.OracleDriver --database-url=
jdbc:oracle:thin:@research.example.com:1521:orcl --kerberos-login-context=
DatabaseKerberos -Coracle.net.authentication_services=(KERBEROS5)
```

Authentication using X.509 client certificates

When Spotfire Server is set up with HTTPS and is configured to require client certificates, the information from the certificates can also be used for login purposes.

This method authenticates users by using an X.509 client certificate from the Spotfire client to Spotfire Server.

These are the general steps to configure Spotfire to use X.509 client certificates for authentication:

1. Configure Spotfire Server for HTTPS; see Configuring HTTPS.

2. Install client certificates on each client. For details, see the documentation provided by your operating system vendor.
3. If you have not already done so, import the Certification Authority (CA) certificate(s) to the keystore; see Installing CA certificates.
4. Configure Spotfire Server to require client certificates for HTTPS; see Configuring Spotfire Server to require X.509 client certificates for HTTPS.
5. Configure Spotfire Server to use X.509 client certificates to authenticate users; see Configuring Spotfire Server to use X.509 client certificates to authenticate users.

Installing CA certificates

To use X.509 client certificates for authentication, a keystore with CA certificate(s) must be placed in the installation directory.

Procedure

1. If you do not yet have a keystore, follow these steps:
   a) Create a keystore and import the CA certificate(s) by executing the following command:.

   ```
   ><installation dir>/jdk/bin/keytool -importcert -alias cacert -keystore <installation
    dir>/tomcat/certs/<keystore filename> -file <certificate filename>
   ```

   CA certificates can be in either PEM format or DER format.

   Example for Windows:

   > C:\tibco\tss*<version>*\jdk\bin\keytool -importcert -alias cacert -keystore C:\tibco\tss*<version>*\tomcat\certs*example*.jks -file cacert.cer

   where "example" in *example*.jks is the server hostname.

   b) Repeat the previous step for each additional CA certificate.
2. When you have a keystore containing the CA certificate(s), copy the keystore file to the `<installation dir>/tomcat/certs` directory.

The keystore containing the CA certificate(s) can be in either PKCS #12 or JKS format.

Configuring Spotfire Server to require client certificates for HTTPS

This procedure configures the server to require a valid user certificate for all connections.

This is done by editing the `server.xml` file.

Prerequisites

You have performed the first three steps in the topic Authentication using X.509 client certificates.

Procedure

1. Open the following configuration file in an XML editor or a text editor: `<server install dir>/tomcat/conf/server.xml`.
2. Locate the section containing the configuration for the HTTPS connector:

```
<Connector port="443"
           maxHttpHeaderSize="65536"
           connectionTimeout="30000"
           enableLookups="false"
           URIEncoding="UTF-8"
           disableUploadTimeout="true"
```

```
               server="TIBCO Spotfire Server"
               compression="on"
               compressibleMimeType="text/html,text/xml,text/plain,text/css,application/
json,application/javascript,image/svg+xml,application/xml"
               acceptorThreadCount="2"
               keepAliveTimeout="30000"
               maxKeepAliveRequests="-1"
               maxThreads="2000"
               SSLEnabled="true"
               scheme="https"
               secure="true">
      <SSLHostConfig certificateVerification="none"
                     truststoreFile="./certs/[server hostname].jks"
                     truststorePass="changeit"
                     truststoreType="jks"
                     sslProtocol="TLS"
                     protocols="TLSv1.2"
                     honorCipherOrder="true"
                     ciphers
   ...
        <Certificate certificateKeystoreFile="./certs/[server hostname].jks"
                     certificateKeystorePassword="changeit"
                     certificateKeystoreType="jks"
                     certificateKeyAlias="[server hostname]" />
      </SSLHostConfig>
    </Connector>
```

3. Update the **`truststoreFile`** parameter with the name of the keystore file containing the CA certificate(s).
4. Set the **`truststorePass`** parameter to the password for the keystore file containing the CA certificate(s).
5. Set the **`truststoreType`** parameter to `jks` for a Java keystore or `pkcs12` for a PKCS #12 keystore.
6. Set the **`certificateVerification`** parameter to `required`.

Configuring Spotfire Server to use X.509 client certificates to authenticate users

This procedure configures the server process for authenticating users with client certificates.

This configuration is done on the command line.

Prerequisites

You have performed the first four steps in the topic Authentication using X.509 client certificates.

Procedure

1. Use the command config-client-cert-auth to configure the client certificates authentication. For more information, see Executing commands on the command line.

2. Use the command config-auth to apply the X.509 client certificates single sign-on authentication method.

If you intend to use an LDAP user directory, an attribute in the certificate's Distinguished Name (DN) must match an LDAP account name. By default, the server will use the Common Name (CN) attribute as account name. Use the configuration tool or the config-client-cert-auth command to configure the server to use another attribute as account name.

Examples

- Using the entire DN as account name:

```
config config-client-cert-auth --name-attribute="DN"
```

This will use the entire DN as account name.

- Using the Subject Alternative Name of type rfc822Name as account name:

```
config config-client-cert-auth --name-attribute="subjectAltName:rfc822Name"
```

This will use a Subject Alternative Name as account name.

Web authentication

When using web authentication, a web browser will be displayed for all users, allowing them to log in to Spotfire using an external authentication provider, such as Google.

By default, the web authentication method supports authentication providers with OpenID Connect support, such as Google. The supported authentication providers can be expanded using the Custom Web Authenticator API. If you configure and enable several authentication providers, users will be allowed to select any of these providers. Users can select to remember the chosen provider, thereby enabling single sign-on, as long as they are logged in on that account.

Web authentication can be combined with username and password authentication.

Configuring OpenID Connect

These instructions are for configuring a default OpenID Connect web authentication provider using the configuration tool.

Prerequisites

- You have configured a public address URL. To do this, go to the Public Address page in the Spotfire Server configuration tool and enable the public address URL `http[s]://<spotfire server>[:<port>]/`.
- You have registered a client at the provider with a return endpoint URL, and received a client ID and a client secret from the provider.
  - The registered client must support the Authorization Code Grant.
  - The registered client must have permission to request the scopes that the server is configured to request. By default, these scopes are "openid", "profile", and "email", but the latter two can be removed and other scopes can be added.

For the default OpenID Connect web authentication providers, use the URL (starting with the configured public address URL):

```
http[s]://<spotfire server>[:<port>]/spotfire/auth/oidc/authenticate
```

When using web authentication, it is recommended to use HTTPS.

It is recommended to use the **Auto-create** option for the post-authentication filter.

Procedure

1. Open the Spotfire Server configuration tool. For information on launching the configuration tool, see Opening the configuration tool.
2. In the configuration tool, select the **Configuration** tab.
3. On the Configuration Start page, select the authentication method **Web authentication**.

   If, for example for backward compatibility with older Spotfire clients, you want to combine web authentication with username and password authentication, you should select the **BASIC** authentication method. This way, the launched web browser will have both a username and password alternative, and the alternative to use an external web authentication provider.
4. On the OpenID Connect page, select **Yes** to enable OpenID Connect authentication.
5. To add and configure a new provider, click **Add new provider**.
6. For each added provider, select **Yes** to enable the provider, and specify the **Provider name** (that will be displayed for users when selecting a provider).
7. For each provider, specify the **Discovery document URL**, the **Client ID** and the **Client secret**, as received when registering a client at the provider.
8. Save the configuration and restart the Spotfire Server.

Advanced OpenID Connect settings

More advanced settings can be configured for OpenID Connect, specifying what is displayed for end-users and what is communicated on the end-users between the provider and Spotfire Server.

For more information on these settings, refer to the documentation of the provider and to OpenID Connect, https://openid.net/specs/openid-connect-core-1_0.html.

Option	Description
Username claim	By default, the value of the **sub** claim is used. Another claim can be specified.
Display name claim	The name of the claim from which to retrieve the display name can be configured.
Email claim	The name of the claim from which to retrieve the email address can be configured.
Domain	Three different options are supported: • Option 1: use the value of a specific claim. This is the default option and the name of the default claim is **iss**. • Option 2: use a static domain name. • Option 3: parse the username claim. The value of the username claim should be on the format 'user@domain' or 'domain\user'. In this case, the 'user' part will be used as username and the 'domain' part will be used as domain.
Scopes	Add scopes to specify which access privileges are being requested. The requested scopes should preferably give access to the **name** and **email** claims.

Option	Description
Auth request prompt value	The value to give the **prompt** request parameter when making the authentication request. Controls how the provider prompts the end-user. May be one of **none**, **login**, **consent** and **select_account**. This is optional. By default the parameter will be omitted from the request.
Custom parameters	Specify any custom parameters that will be included in the request to the authorization endpoint.
Background color	You can specify a background color, as a hexadecimal value, for the added provider on the login page.

Configuring custom web authentication

These instructions are for configuring custom web authentication using the configuration tool.

Prerequisites

- You have implemented the CustomWebAuthenticator API.
- If applicable, you have registered a client at the provider, using a return endpoint URL, and have received a client ID and a client secret from the provider. Use the URL:

```
http[s]://<spotfire server>[:<port>]/spotfire/auth/custom/authenticate
```

When using web authentication, it is recommended to use HTTPS.

It is recommended to use the **Auto-create** option for the post-authentication filter.

Procedure

1. Open the Spotfire Server configuration tool. For information on how to launch the configuration tool, see Configuration using the configuration tool.
2. In the configuration tool, select the **Configuration** tab.
3. On the Configuration Start page, select the authentication method **Web authentication**.

   If, for example for backward compatibility with older Spotfire clients, you want to combine web authentication with username and password authentication, select the **BASIC** authentication method. This way, the launched web browser will have both a username and password alternative, and the alternative to use an external web authentication provider.
4. On the Custom Web Authentication page, select **Yes** to enable custom web authentication.
5. Specify the **Authenticator class** - the class implementing the CustomWebAuthenticator API interface.
6. Add any **Initialization parameters** relevant to your custom web authentication implementation.
7. Save the configuration and restart the Spotfire Server.

Configuring the database connection for Spotfire Server using Kerberos (SQL Server)

If you use an SQL Server database, follow these instructions to configure the database connection for Spotfire Server.

Procedure

- To bootstrap Spotfire Server, execute the following `bootstrap` command, replacing `<database url>` with the JDBC connection URL. This URL must include `;integratedSecurity=true;authenticationScheme=JavaKerberos` options.

```
> config bootstrap --test --driver-class=com.microsoft.sqlserver.jdbc.SQLServerDriver
--database-url=<database url> --kerberos-login-context=DatabaseKerberos
```

```
> config bootstrap --test --driver-class=com.microsoft.sqlserver.jdbc.SQLServerDriver
--database-url=jdbc:sqlserver://db.research.example.com:1433;DatabaseName=
spotfire_server;integratedSecurity=true;authenticationScheme=JavaKerberos
--kerberos-login-context=DatabaseKerberos
```

Configuring anonymous authentication

Anonymous authentication allows anyone to access public information that is available for viewing on the Spotfire web client without prompting them for a user name or password.

Procedure

1. Open a command line and export the active server configuration by using the export-config command; for additional information, see Executing commands on the command line.
2. On the command line, enter the following command:

   ```
   config config-anonymous-auth --enabled=true
   ```

   For details on the command, see config-anonymous-auth.
3. Enter the following command to enable the guest account:

   ```
   config enable-user --username=ANONYMOUS\guest
   ```

   For details on the command, see enable-user.
4. Import the configuration back to the Spotfire database by using the import-config command.
5. Restart the Spotfire Server.

Two-factor authentication

Spotfire Server supports one form of two-factor authentication. It is possible to combine the chosen primary authentication method with X.509 client certificates.

Typically, the primary authentication method in the two-factor authentication is Basic, but it is also possible to use the other authentication methods.

When two-factor authentication is enabled, the server requires the name of the authenticated user to match the user name in the provided X.509 certificate. For instructions, see Configuring two-factor authentication.

Configuring two-factor authentication

You can configure authentication through X.509 client certificates in addition to your primary authentication method.

Procedure

1. Configure the server to use the chosen primary authentication method.

2. In the configuration tool, on the Configuration page, in the Configuration Start panel, select **Enable two-factor authentication**.
   A second Authentication panel is added.
3. In the second Authentication panel, configure the server to use client certificates.

Configuring two-factor authentication using the command line

You can set up two-factor authentication by using the command line or the configuration tool.

Procedure

1. Use the command line to set up the primary authentication method and the client certificates.
2. On the command line, enter the following command:

```
config config-two-factor-auth --enabled=true
```

External authentication

Spotfire clients may access Spotfire Server through an external authentication mechanism, usually a proxy or a load balancer.

When using an external authentication mechanism, Spotfire Server gets the external user name from an HTTP header or a cookie. Getting the external user name from an HTTP header or a cookie could potentially be a security risk and it is strongly recommended that you restrict the permissions to use this feature. It is also recommended to use the external authentication method only when using a load balancer or proxy.

When configuring external authentication, you can add several constraints:

- You can configure Spotfire Server to allow external authentication only when using a secure (TLS) connection.
- You can specify allowed hostnames and/or IP addresses of the client computers that are permitted to log in using external authentication. You can list allowed IP addresses and/or write regular expressions; if you specify both, Spotfire Server first checks in the list and then the regular expression.

In some cases, the proxy or load balancer has already forced the client to authenticate itself. Some proxies and load balancers are capable of forwarding the name of the authenticated user to Spotfire Server. By enabling external authentication on Spotfire Server, the server can extract the identity of the client so that the client does not have to authenticate twice. Any proxy or load balancer that can propagate the user name so that it is available in the HTTP request to the server as a request attribute, is compatible

Typical scenarios are:

- When both the Spotfire Server cluster and its load balancer are configured for NTLM authentication.
- When the load balancer is configured for X.509 client certificate authentication and propagates the user names extracted from the certificates.
- When the load balancer requires the user to authenticate with username and password in a web form (for example SiteMinder). In this case, you must configure the load balancer to intercept and authenticate requests to, and only to, the path `/spotfire/sf_security_check_external_auth`.

External authentication may be used as a supplementary authentication method that can be used together with the main authentication method, but it can also be used as the main and only authentication method.

- If clients are to always go through a load balancer to reach Spotfire Server, configure external as the main authentication method in the **Authentication** panel. In this case it is not possible to access a Spotfire Server directly. You must also specify a declared authentication method in the **External Authentication** panel.
- Even if a load balancer is used in front of a set of Spotfire Servers, accessing the server directly may be desired. If this is the case, configure another authentication mechanism (any mechanism is allowed) as the main authentication method, and configure external as a supplementary authentication method.

See Configuring external authentication on page 112 for more information.

Configuring external authentication

You can configure external authentication by using the configuration tool or the command line.

Procedure

- Use the configuration tool or the config-external-auth command to set up and enable the external authentication method.

  Use the following information to set options:

Enable External Authentication (required)	Specifies whether the external authentication method should be enabled.
Declared authentication method	Select the authentication method used by the load balancer.
Source	**Attribute:** Enter the name of the HTTP request attribute that contains the name of the authenticated user. **Header:** Enter the name of the HTTP request header that contains the name of the authenticated user. **Cookie:** Enter the name of the HTTP request cookie that contains the name of the authenticated user. **Custom Authenticator:** Enter the name of the class that implements the com.spotfire.server.security.CustomAuthenticator interface. **Authentication Filter:** Retrieves the user name from the getUserPrincipal() method of javax.servlet.http.HttpServletRequest.  The Authentication Filter API has been deprecated. Use the CustomAuthenticator API, the CustomWebAuthenticator API, or a custom login page instead.
Require TLS	Select `yes` for external authentication to be available for TLS connections only.
Allowed host (hostname or IP address)	A list of hostnames and/or IP addresses of the client computers that are allowed to perform external authentication. If no allowed hosts are specified, all client computers are permitted to perform external authentication.
Allowed IP:s (regular expression)	Add a regular expression that matches the IP addresses of remote hosts that are permitted to perform external authentication. The regular expression shall be written in the syntax supported by java.util.regex.Pattern.

Name filter expression (optional)	A regular expression that can be used to filter the user name that is extracted from the specified request attribute. The value of the regular expression's first capturing group will be used as the new user name.  One use of this feature is to remove the domain names in cases where Spotfire Server is configured to collapse the domains into one single domain within the server. For example, if the attribute contains "domainname\username", you can use the regular expression ".*\\(.*)" to remove "domainname\".
Lower case conversion (optional)	Specifies whether to convert the propagated user name to lowercase. The default is not to convert to lowercase.

External directories and domains

You can configure Spotfire Server to integrate with external directories such as LDAP directories or Windows domains.

Spotfire Server keeps track of which domain every user belongs to. Users who are created by an administrator directly within Spotfire Server belong to the SPOTFIRE domain. When the user directory is configured for **Database**, this is the domain being used.

External users keep their domain name from the external directory, and the domain name appears as part of their user name throughout the Spotfire interface.

The supported external directories can have domain names in two forms:

- DNS domain names, for example "research.example.com". A complete user name looks like this: someone@research.example.com.
- NetBIOS domain names, for example "RESEARCH". A complete user name looks like this: RESEARCH\someone.

When configuring Spotfire Server, the desired domain name style must be set before the server is started for the first time. The domain name style to use is dependent on the combination of authentication method and user directory of your Spotfire implementation.

Be careful when selecting a domain name style for your system; it will affect what information Spotfire Server stores within the Spotfire database. The domain name style can be changed using the switch-domain-name-style command if the user directory is in LDAP mode and is synchronizing with an Active Directory Server. For other user directory modes, there are no tools to alter that information if the domain name style later needs to be changed.

Below is a matrix showing which domain name style to use for different combinations of authentication method and user directory. Combinations that are not supported are marked " — ".

Spotfire Server will warn and even refuse to start if you try to set up an authentication method and a user directory with incompatible domain name styles. If you for some reason need to go ahead with an officially incompatible configuration, you will need to set the **allow incompatible domain name styles** configuration property to make the server start at all. One way to handle this could be a custom post-authentication filter that creates a bridge between the two originally incompatible domain name styles. (The **allow incompatible domain name styles** option can be set using the config-userdir command. For information about custom post-authentication filters, see Post-authentication filter.)

Collapse Domains Configuration Property Enabled

	User directory type			
Authentication method	**Database**	**LPAD/AD**	**LDAP/other**	**Windows NT**
Basic database	NetBIOS(DNS)	—	—	—
Basic/LDAP/AD	NetBIOS(DNS)	NetBIOS(DNS)	NetBIOS(DNS)	—
Basic/LDAP/other	NetBIOS(DNS)	NetBIOS(DNS)	NetBIOS(DNS)	—
Basic/Windows NT	—	—	—	NetBIOS(DNS)
NTLM	NetBIOS(DNS)	NetBIOS(DNS)	NetBIOS(DNS)	—
Kerberos	NetBIOS(DNS)	NetBIOS(DNS)	NetBIOS(DNS)	—
X.509 Client Certs.	NetBIOS(DNS)	NetBIOS(DNS)	NetBIOS(DNS)	—

— Unsupported combination of authentication method and user directory.

Collapse Domains Configuration Property Not Enabled

	User directory type			
Authentication method	**Database**	**LPAD/AD**	**LDAP/other**	**Windows NT**
Basic database	NetBIOS, DNS	—	—	—
Basic/LDAP/AD	NetBIOS, DNS	NetBIOS, DNS	#	—
Basic/LDAP/other	NetBIOS, DNS	#	DNS	—
Basic/Windows NT	—	—	—	NetBIOS, DNS
NTLM	NetBIOS, DNS	NetBIOS, DNS	#	—
Kerberos	NetBIOS, DNS	NetBIOS, DNS	DNS	—
X.509 Client Certs.	NetBIOS, DNS	NetBIOS, DNS	DNS	—

NetBIOS is the recommended domain name style, but DNS will also work.

— Unsupported combination of authentication method and user directory.

For this combination of authentication method and user directory, enable the collapse domains option.

A consequence of the new domain tracking is that users may have to provide the domain names as part of their user names when logging in to Spotfire Server. For the Basic/LDAP and Basic/Windows NT authentication methods, the setting of the wildcard domain configuration property decides how the server maps a user to a domain during authentication. When the wildcard domain configuration property is enabled (this is the default), Spotfire Server checks whether the user name contains a domain name, and if it does, that domain name is used. If not, the server attempts to authenticate the user with the provided user name and password in every domain it knows about, until the combination

of domain name, user name, and password results in a successful authentication, or until there are no more domain names to try. If the wildcard domain configuration property is turned off, the domain name must be specified by the user unless it belongs to the configured default domain. This can be configured in the configuration tool.

If the wildcard domain configuration property is enabled and two identically named users in different domains have the same password, there is a risk that the wrong account will be selected when one of these users logs in. Thus, if security has a higher priority than user convenience, make sure to turn off the wildcard domain configuration property. There is also the risk that multiple authentication attempts will lock out the "correct" user.

Spotfire Server provides a configuration property that reverts to the behavior from previous releases. The configuration property is called `collapse-domains` and enabling this means that the external domain of a user is essentially ignored, and that different users with the same user name, but in different domains, will share an account on Spotfire Server. When the collapse domains configuration property is enabled, all external users and groups will be associated with the SPOTFIRE domain, regardless of which domain they belong to in the external directory.

If you want to keep running Spotfire Server without ever caring about domain names, enable both the `collapse-domains` and `wildcard-domain` configuration properties. Doing so will ensure that all users belong to the internal SPOTFIRE domain, and no users will have to enter a domain name when logging in. (The `collapse-domains` configuration property can be set in the configuration tool or by using the config-userdir command).

All users will belong to one domain when the `collapse-domains` configuration property is enabled. If there are multiple users with the same account name in different external domains, they will now effectively share the same account within Spotfire Server. If security has a higher priority than user convenience, make sure not to enable the collapse domain configuration property.

It is not recommended to change the `collapse-domains` configuration property after once having synchronized Spotfire Server with an external directory. This creates double accounts with different domain names for every synchronized user and group in the user directory. The new accounts do not inherit the permissions of the old accounts.

LDAP synchronizations

You can schedule when Spotfire Server synchronizes its user directory with LDAP directories. Both users and groups are synchronized in the background, and user and group look-ups query the Spotfire database rather than the LDAP directory.

There are two algorithms that can be used when configuring the recurrence of synchronization tasks: one is based on cron schedules and the other on sleep time between synchronizations.

Sleep time is only used when no cron schedule exists for the LDAP configuration. The sleeping period is configurable and by default it is set to 60 minutes.

New configurations have two default cron schedules: "restart" and "daily". "Restart" runs synchronization at each restart of Spotfire Server; "daily" runs synchronization once a day (at midnight server time). Upgraded configurations may not have these default cron schedules.

Each LDAP configuration has its own schedules. It is possible to use cron schedules for one LDAP configuration and sleep time for another.

User synchronization

By default, the user directory only synchronizes users (not groups) from the LDAP directories.

After an LDAP user has been synchronized and imported to the user directory, the user account becomes a permanent part of the user directory. If the LDAP user is later removed from the LDAP

directory, the corresponding user account in the user directory is disabled. Disabled accounts remain visible in the Spotfire system but the user cannot log in.

To prevent user accounts from being disabled by failed synchronization attempts, for example caused by network errors, the `safe-synchronization` option can be enabled. When this option is enabled, no user accounts are disabled solely because they could not be found during synchronization. By default, this option is not enabled because of the potential security issues.

It is usually not possible to log in as a removed LDAP user anyway because the LDAP directory blocks the authentication attempt if it is also responsible for authenticating users.

User accounts may also be explicitly disabled in the LDAP directories. In this case the user accounts are disabled in the user directory, regardless of the safe synchronization setting.

Group synchronization

Group synchronization mirrors in the user directory the group hierarchies that are in the LDAP directory.

When you set the `group-sync-enabled` option (in the config-ldap-group-sync command), the user directory synchronizes groups from the LDAP directory. Synchronizing groups relieves the administrator of the responsibility of managing group memberships. Assigning licenses and privileges to Spotfire groups is still accomplished in the Administrator Manager in Spotfire Analyst.

Synchronized LDAP groups cannot be manually modified in the user directory. Synchronized groups can be placed into manually created groups in the user directory, and thereby be granted permissions. If an LDAP group has been synchronized and it is removed from the list of groups to synchronize, it keeps the members from the last synchronization, but becomes an ordinary group that can be modified in Spotfire.

The user directory does not support cyclic group memberships, where the ancestor of a group is also a descendant of the same group. If the user directory detects a group membership cycle, it will be broken up arbitrarily.

When configuring the groups to be synchronized, specify either the group account names or the distinguished names. The account names and the distinguished names may contain an asterisk (*) as a wildcard character. This wildcard behaves just like the asterisk wildcard in standard LDAP search filters.

It is also possible to specify the distinguished name of an LDAP container containing one or more groups. All those groups will then be synchronized. It is possible to mix all variants.

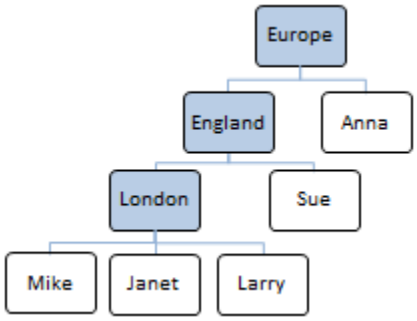
If the `Group synchronization enabled` configuration property is set and no groups or group context names are configured, the user directory synchronizes all groups that it can find in the configured context names.

The synchronized groups can also be used to filter the set of users that are synchronized with the user directory. By enabling the `filter-users-by-groups` option, only users that are members of at least one of the synchronized groups are synchronized with the user directory.

Group-based and role-based synchronization

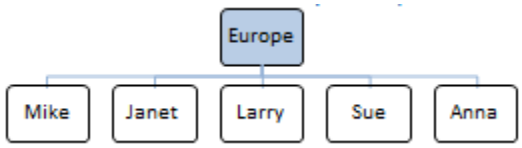
For Active Directory servers, Spotfire Server can synchronize groups. For the Directory Server product family, Spotfire Server can synchronize either groups or roles.

Here are examples of the default behavior of group-based and role-based group synchronization. The examples are based on the following figure:

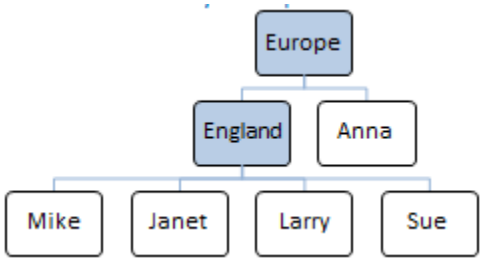


Group-based synchronization:

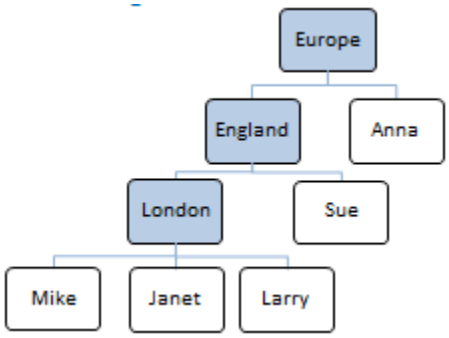
- If you only specify the group "Europe" to be synchronized in your LDAP configuration, the user directory synchronizes according to the figure below. The groups England and London will not be visible because they are automatically replaced with their members:



- If you specify the groups "Europe" and "England" to be synchronized in your LDAP configuration, the user directory will synchronize according to the figure below. The group London will not be visible, but will automatically be replaced with its members:

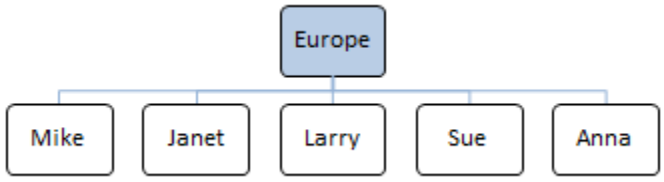


- If you specify the groups "Europe", "England", and "London" explicitly to be synchronized in your LDAP configuration, the user directory will synchronize according to the figure below:

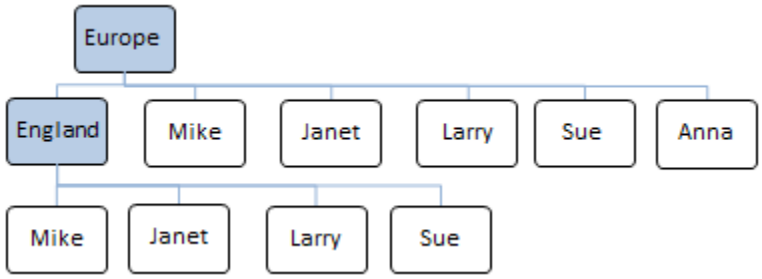


Role-based synchronization:

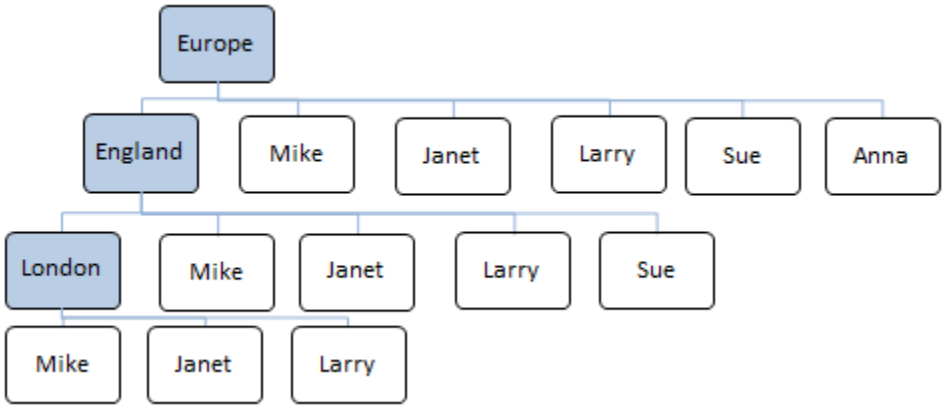
- If you only specify the role "Europe" to be synchronized in your LDAP configuration, the user directory will synchronize according to the figure below. The roles England and London will not be visible, but will automatically be replaced with their members:



- If you specify the roles "Europe" and "England" to be synchronized in your LDAP configuration, the user directory will synchronize according to the figure below. The role London will not be visible. Due to the nature of roles in the Directory Server product family, every role will automatically include all direct members as well as all members of sub roles:



- If you specify the roles "Europe", "England" and "London" explicitly to be synchronized in your LDAP configuration, the user directory synchronizes according to the figure below. Due to the nature of roles in the Directory Server product family, every role automatically includes all direct members as well as all members of sub-roles:



There are two algorithms to choose from when configuring group synchronization: the `memberOf` and the `member` algorithms.

- The `memberOf` algorithm relies on a calculated attribute in the LDAP directory and may induce more load on the LDAP servers. Not all LDAP directories support the `memberOf` algorithm.
- The `member` algorithm performs significantly more LDAP queries, but with much smaller result sets than the `memberOf` algorithm. See the recommendations below for group synchronization on different LDAP servers.

Recommendations

For Microsoft Active Directory server:

- Configure group-based synchronization with the `memberOf` algorithm.

For Sun Java System Directory Server (version 6 and later), do one of the following:

- Configure group-based synchronization with the `memberOf` algorithm.
- Configure role-based synchronization with the `memberOf` algorithm.

For Sun ONE Directory Server (version 5 and earlier), do one of the following:

- Configure role-based synchronization with the `memberOf` algorithm.
- Configure group-based synchronization with the `member` algorithm.

The following combinations do *not* work on Sun ONE Directory Servers:

- Configuring group-based synchronization with the `memberOf` algorithm.
- Configuring role-based synchronization with the `member` algorithm.

LDAP authentication and user directory settings

The following information is required to set up LDAP authentication and user directory mode, including LDAP group synchronization. Contact the LDAP directory administrator if you do not have the required information.

The following table provides an overview of LDAP settings and their applicability. Detailed descriptions of the settings are provided below the table.

- A: Applicable to LDAP as authentication mechanism
- UD: Applicable to LDAP User Directory mode
- GS: Applicable to LDAP User Directory mode with group synchronization
- M: Mandatory
- **: Required by configurations with LDAP server type **Custom**. These options have template values for the non-predefined LDAP server types. The template values can be overridden when necessary.

A			**Authentication Attribute** Specifies the name of the LDAP attribute containing a user identity that can be used for authenticating with the LDAP server.
A	UD	M	**LDAP Server Type** Specifies the type of LDAP server: ActiveDirectory, SunOne, SunJavaSystem, or Custom.
A	UD	M	**LDAP Server URLs** A white-space separated list of LDAP server URLs.
A	UD	M	**Context Names** A list of distinguished names (DNs) of the containers holding the user accounts to be visible within Spotfire Server.
A	UD		**Username** The name of the LDAP service account to be used when searching for users and groups in the LDAP directory.
A	UD		**Password** The password for the LDAP service account.
A	UD		**Security Authentication** Specifies the security level to use when binding to the LDAP server. The default value is simple.

A	UD	**	**User Search Filter** Specifies an LDAP search expression filter to be used when searching for users.
A	UD		**Referral Mode** Specifies how LDAP referrals should be handled.
A	UD	**	**Username Attribute** Specifies the name of the LDAP attribute containing the user account names.
A	UD		**Custom LDAP Properties** Multiple key-value pairs specifying additional JNDI environment properties to be used when connecting to the LDAP server.
	UD		**Request Control** Specifies the type of LDAP controls to be used when executing search queries to the LDAP server: Probe, PagedResultsControl, VirtualListViewControl or none.
	UD		**Page Size** Specifies the page size to be used with the paged results control or the virtual list view control when performing search queries to the LDAP server. The page size value defaults to 1000 for both the paged results control and the virtual list view control.
	UD		**Import Limit** Specifies a threshold that limits the number of users that can be imported from an LDAP server to Spotfire Server in one query.
	UD		**Synchronization Schedules** Specifies a list of schedules for when the synchronization task should be performed.
	GS		**Group Synchronization Enabled** Specifies whether or not group synchronization should be enabled for this LDAP configuration.
	GS		**Group Names** Specifies a list of distinguished names (DNs) of either individual groups to be synchronized or a context name where all groups are to be synchronized. If the group synchronization enabled option is set and the list of group names is empty, then all groups that can be found in the LDAP directory will be synchronized.
	GS	**	**Group Search Filter** Specifies an LDAP search expression filter to be used when searching for groups.
	GS	**	**Group Name Attribute** Specifies the name of the LDAP attribute containing the group account names
	GS	**	**Supports memberOf** Specifies whether or not the LDAP servers support a memberOf-like attribute on the user accounts that contain the names of the groups or roles that the users are members of. In general, this is true for all Microsoft Active Directory servers and all types of Sun directory servers.
	GS	**	**Member Attribute** For all LDAP servers with support for a memberOf-like attribute, this option specifies the name of the LDAP attribute on the user account that contains the names of the groups or roles that the user is a member of.

GS	**	**Ignore Member Groups** Specifies whether or not the group synchronization mechanism should recursively traverse the synchronized groups' non-synchronized subgroups and include their members in the search result.

Authentication Attribute

Specifies the name of the LDAP attribute containing a user identity that can be used for authenticating with the LDAP server. This attribute fills no purpose in most common LDAP configurations, but can be useful in more advanced setups where the distinguished name (DN) does not work for authentication or where users should be able to log in using a username that does not map directly to an actual LDAP account. A typical case for using this option is when setting up SASL; see SASL authentication for LDAP.

LDAP Server Type

Specifies the type of LDAP server. There are four valid types: ActiveDirectory, SunOne, SunJavaSystem, and Custom.

When specifying one of the predefined server types, we will assume that default values will be applied for the most fundamental configuration options. It is possible to override the default values. When specifying a Custom LDAP server type, there is no configuration template and all fundamental configuration options must be specified explicitly. The table above shows which configuration options are required for a Custom LDAP server type.

LDAP Server URLs

A whitespace-separated list of LDAP server URLs. An LDAP server URL has the format `<protocol>://<server>[:<port>]`

- <protocol>: Either LDAP or LDAPS
- <server>: The fully qualified DNS name of the LDAP server
- <port>: An optional number indicating the TCP port the LDAP service is listening on. When using the LDAP protocol, the port number defaults to 389. When using the LDAPS protocol, the port number defaults to 636. Active Directory LDAP servers also provide a Global Catalog containing forest-wide information, instead of domain-wide information only. The Global Catalog LDAP service by default listens on port number 3268 (LDAP) or 3269 (LDAPS).

Spotfire Server does not expect any search base, scope, filter, or other additional parameters after the port number in the LDAP server URLs. Such properties are specified using other configuration options for this command.

Examples of LDAP server URLs:

LDAP://myserver.example.com

LDAPS://myserver.example.com

LDAP://myserver.example.com:389

LDAPS://myserver.example.com:636

LDAP://myserver.example.com:3268

LDAPS://myserver.example.com:3269

Context Names

A list of distinguished names (DNs) of the containers holding the LDAP accounts to be visible within Spotfire Server. When specifying more than one DN, the DNs must be separated by pipe characters (|). If the specified containers contain a large number of users, but only a few should be visible in Spotfire Server, a custom user search filter can be specified to include only the filtered users; see "User Search Filter", below.

Username

The name of the LDAP service account to be used when searching for users and groups in the LDAP directory. This service account does not need to have any write permissions, but it needs to have read permissions for all configured context names (LDAP containers). For most LDAP servers, the account name is the account's distinguished name (DN). For Active Directory, the account name can also be specified in the forms `ntdomain\name` or `name@dnsdomain`.

Examples:

CN=spotsvc,OU=services,DC=research,DC=example,dc=COM

RESEARCH\spotsvc (Active Directory only)

spotsvc@research.example.com (Active Directory only)

Password

The password for the LDAP service account.

Security Authentication

Specifies the security level to use when binding to the LDAP server. The default value is `simple`. Only use this parameter in special cases, and use it with care in production environments.

- To enable anonymous binding, it should be set to **none**.
- To enable plain user name/password authentication, it should be set to **simple**.
- To enable SASL authentication, it should be set to the name of the SASL mechanism to be used. Spotfire Server supports the two SASL mechanisms DIGEST-MD5 and GSSAPI. You can set multiple -C flags to set the additional JNDI environment properties that the SASL authentication mechanism typically requires

A typical case for using this option is when setting up SASL; see SASL authentication for LDAP.

User Search Filter

This parameter specifies an LDAP search expression filter to be used when searching for users.

If only a subset of all the users in the specified LDAP containers should be allowed access to Spotfire Server, a restrictive user search filter can be specified. For instance, the search expression can be configured so that it puts restrictions on which groups the users belong to, or which roles they have.

- For Active Directory servers, the parameter value defaults to objectClass=user
- For Active Directory servers, access can be restricted to only those users belonging to a certain group by using a search expression with the pattern &(objectClass=user)(memberOf=<groupDN>) where <groupDN> is to be replaced by the real DN of the group to which the users must belong. If the users are divided among multiple groups, use the pattern &(objectClass=user)(|(memberOf=<firstDN>)(memberOf=<secondDN>)). Add extra (memberOf=<groupDN>) sub-expressions as needed.

  Example: &(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)
- For any version of the Sun Directory Servers, it defaults to objectClass=person.
- For a Sun Java System Directory Server version 6 and later, the same effect can be achieved by using a search expression with the pattern &(objectClass= person)(isMemberOf=<groupDN>). If the users are divided among multiple groups, use the pattern &(objectClass=person)(|(isMemberOf=<firstDN>)(isMemberOf=<secondDN>)). Add extra (isMemberOf=<groupDN>) sub-expressions as needed.

  Example: &(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)
- For the Directory Server product family, access can be restricted to only those users having certain specific roles. The search expression for role filtering must match the pattern &(objectClass=person)(nsRole=<roleDN>). If multiple roles are of interest, use the pattern &(objectClass=person)(|

(nsRole=<firstDN>))(nsRole=<secondDN>)). Add extra (nsRole=<roleDN>)) sub-expressions as needed.

Example: &(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)

The syntax of LDAP search expression filters is specified by RFC 4515. Consult this specification for information about more advanced filters.

Referral Mode

This argument specifies how LDAP referrals should be handled. Valid arguments are follow (automatically follow any referrals), ignore (ignore referrals) and throw (fail with an error). The default and recommended value is follow.

Username Attribute

Specifies the name of the LDAP attribute containing the user account names. For Active Directory servers the value defaults to sAMAccountName. For the Directory Server product family with a default configuration, it defaults to uid.

Custom LDAP Properties

Multiple key-value pairs specifying additional JNDI environment properties to be used when connecting to the LDAP server. For instance, specifying the key java.naming.security.authentication and the value simple have the same result as setting the Security Authentication option to "simple".

Request Control

This option determines the type of LDAP controls to be used when executing search queries to the LDAP server. Valid controls are Probe, PagedResultsControl, VirtualListViewControl, and none.

The default behavior is to probe the LDAP server for the best supported request control. The paged results control is always preferred, since it provides the most efficient way of retrieving the result of the query. The virtual list view control can also be used to retrieve a large number of users, if the paged results control is not supported. The virtual list view control will automatically be used together with a sort control. Both the paged results control and the virtual list view control support a configurable page size, as specified by the page size option.

Page Size

This argument specifies the page size to be used with the paged results control or the virtual list view control when performing search queries to the LDAP server. The page size value defaults to 1000 for both the paged results control and the virtual list view control.

Import Limit

This argument specifies a threshold that limits the number of users that can be imported from an LDAP server to Spotfire Server in one query. This can be used to prevent accidental flooding of Spotfire Server's User Directory when integrating with an LDAP server with tens or even hundreds of thousands of users. By setting an import limit, the administrator can be sure that an unexpected high number of users won't affect the server's performance. By default, there is no import limit. To explicitly request unlimited import, set the parameter value to -1. All positive numbers are treated as an import limit. Leave this parameter untouched. in most cases.

Group Synchronization Enabled

Specifies whether or not group synchronization should be enabled for this LDAP configuration.

Group Names

Specifies the groups to be synchronized. Groups can be specified with either their account names or their distinguished names (DNs). The account names and the distinguished names may contain an asterisk (*) as a wildcard character. This wildcard behaves just like the asterisk wildcard in standard LDAP search filters. Wildcards work for both account names and distinguished names.

It is also possible to specify the distinguished name of an LDAP container containing multiple groups and thereby synchronizing all those groups. Wildcards can also be used for specifying group containers.

It is possible to mix all variants above. Consider the following when specifying a group to be synchronized:

- Specify either the group's account name or its distinguished name (DN). The account name must match the value of the configured group name attribute.
- It is possible to use an asterisk (*) as a wildcard character s in the account names when specifying group names. If a configured group name contains wildcard characters and matches multiple groups in the directory, all those groups will be synchronized.
- It is also possible to specify the distinguished name of an LDAP container containing one or more groups. All those groups will then be synchronized.
- It is possible to mix all variants.

If the enable group synchronization configuration property is set and the list of group names is empty, then all groups that can be found in the configured context names in the LDAP directory will be synchronized.

Synchronization Schedules

Specifies a list of schedules for when the group synchronization task should be performed. The schedules are specified in the cron format, where each schedule consists of either five fields or one shorthand label.

The five fields are, from left to right, with their valid ranges:

- minute (0-59)
- hour (0-23)
- day of month (1-31)
- month (1-12)
- day of week (0-7, where both 0 and 7 indicate Sunday)

A field may also be configured with the wildcard character (*), indicating that any moment in time matches this field. A group synchronization is triggered when all fields match the current time. If both day of month and day of week have non-wildcard values, then only one of them has to match.

There are also the following shorthand labels that can be used instead of the full cron expressions:

@yearly or @annually: run once a year (equivalent to 0 0 1 1 *)

@monthly: run once a month (equivalent to 0 0 1 * *)

@weekly: run once a week (equivalent to 0 0 * * 0)

@daily or @midnight: run once a day (equivalent to 0 0 * * *)

@hourly: run once an hour (equivalent to 0 * * * *)

@minutely: run once a minute (equivalent to * * * * *)

@reboot or @restart: run every time Spotfire Server is started

Refer to the Wikipedia overview article on the cron scheduler.

Group Search Filter

This parameter specifies an LDAP search expression filter to be used when searching for groups.

- For Active Directory servers, the parameter value defaults to objectClass=group

- For Oracle Directory Servers and Sun Java System Directory Servers, it defaults to objectClass=groupOfUniqueNames
- For Sun ONE Directory Servers, it defaults to &(|(objectclass= nsManagedRoleDefinition) (objectClass=nsNestedRoleDefinition))(objectclass= ldapSubEntry)

Group Name Attribute

Specifies the name of the LDAP attribute containing the group account names:

- For Active Directory servers the value defaults to sAMAccountName
- For any version of the Sun directory servers with a default configuration, it defaults to cn

Supports memberOf

Specifies whether or not the LDAP servers support a memberOf-like attribute on the user accounts that contain the names of the groups or roles that the users are members of. In general, this is true for all Microsoft Active Directory servers and the Directory Server product family.

For some LDAP servers with configurations of type **Custom**, there is no memberOf-like attribute. This is declared by setting the supports memberOf configuration property to "false".

Member Attribute

This parameter value can be set to: memberOf, nsRole, or isMemberOf.

For LDAP configurations with the supports memberOf option set to false, the member attribute option specifies the name of the LDAP attribute on the group accounts that contains the distinguished names (DNs) of its members. In general, this includes LDAP servers with configurations of type Custom and any Sun ONE Directory Servers (version 5 and earlier) when used with group-based synchronization.

For LDAP configurations with the supports memberOf option set to "true", the member attribute option specifies the name of the LDAP attribute on the user accounts that contain the names of the groups or roles that the users are members of. In general, this includes all Microsoft Active Directory servers and all types of Sun Directory Servers version 6 and later. For Sun ONE Directory Servers (version 5 and older), this also applies for roles.

- For Microsoft Active Directory servers, the member attribute value defaults to memberOf.
- For Sun ONE Directory Servers, the member attribute option defaults to nsRole.
- For Sun Java System Directory Server version 6.0 or later, the member attribute option defaults to isMemberOf. To use the roles with the Sun Java System Directory Server or later, it is recommended to use the SunONE configuration template instead.

All configurations with the `memberOf` option set to "false" will use a far less efficient group synchronization algorithm that will generate more traffic to the LDAP servers, because Spotfire Server will first have to search for the distinguished names (DNs) of the group members within the groups, and then perform repeated lookups to translate the member DN to the correct account name.

Ignore Member Groups

This argument determines whether or not the group synchronization mechanism should recursively traverse the synchronized groups' non-synchronized subgroups and include their members in the search result.

For Microsoft Active Directory servers, the parameter value defaults to "false" so that all inherited group memberships are correctly reflected. For any version of the Sun Directory Servers, it defaults to "true" because the role and groups mechanisms in those servers automatically include those members.

Post-authentication filter

After a user's identity is validated, Spotfire Server performs an additional check using the *post-authentication filter*.

This filter has two built-in modes:

- **Block**. When the post-authentication filter is set to **Block**, it blocks all users who are not already present in the Spotfire Server user directory. This is the default mode, and the appropriate mode to use with an LDAP user directory.
- **Auto-create**. When the post-authentication filter is set to **Auto-create**, it automatically creates new accounts for any user who logs in to the server for the first time. This mode is valid only when the user directory mode is set to **Database**.

The blocking mode is the default mode. When it is used with a user directory in LDAP/Active Directory mode, it automatically transforms to the domain name of the authenticated user to match the configured domain name style.

The auto-creating mode is typically applied when using an LDAP directory or X.509 certificates for authentication together with the User Directory set up in database mode. The Post-authentication filter will create users with their external domain names, even though the user directory is in database mode, unless the collapse domains configuration property is enabled. This makes it possible to later switch to LDAP or Windows NT mode. If the collapse domains configuration property is enabled, the users will be created within the internal SPOTFIRE domain and it will not be possible to later switch to LDAP or Windows NT mode.

It is also possible to use the Spotfire Server API to create a custom post-authentication filter to perform additional validation. This filter must be installed in the `<server installation dir>/tomcat/custom-ext` directory on all servers. It is enabled using the config-post-auth-filter command. If a custom filter is used, it will be combined with the built-in filter, meaning that the filters will work together.

HTTPS

By default, Spotfire uses the HTTP protocol for communication between clients and Spotfire Server. To achieve a higher level of security, use the HTTPS protocol instead, ensuring encryption between clients and server.

HTTPS also includes a mechanism for clients to authenticate the server. To have the server authenticate the clients as well, you can enable X.509 client certificate authentication.

To enable encrypted communication using HTTPS, see Configuring HTTPS.

To enable X.509 client certificate authentication, start with Configuring HTTP and then proceed to Authentication using X.509 client certificates.

To configure the server for use with an HTTPS load balancer, see Terminating TLS in a load balancer or reverse proxy on page 73.

Configuring HTTPS

HTTPS ensures that the communication between clients and Spotfire Servers is encrypted.

Prerequisites

Obtain a server certificate and private key, stored in a Java keystore (JKS) or PKCS #12 keystore (P12/PFX).

Procedure

1. Stop Spotfire Server.
2. Copy the keystore file to the <*server installation dir*>/tomcat/certs directory. We suggest using the server's hostname as keystore filename.
3. Open the configuration file <*server installation dir*>/tomcat/conf/server.xml in an XML editor or a text editor and locate the section containing the configuration template for an HTTPS connector:

```
<!--
      <Connector port="443"
                maxHttpHeaderSize="65536"
                connectionTimeout="30000"
                enableLookups="false"
                URIEncoding="UTF-8"
                disableUploadTimeout="true"
                server="TIBCO Spotfire Server"
                compression="on"
                compressibleMimeType="text/html,text/xml,text/plain,text/css,application/
json,application/javascript,image/svg+xml,application/xml"
                acceptorThreadCount="2"
                keepAliveTimeout="30000"
                maxKeepAliveRequests="-1"
                maxThreads="2000"
                SSLEnabled="true"
                scheme="https"
                secure="true">
      <SSLHostConfig certificateVerification="none"
                     truststoreFile="./certs/[server hostname].jks"
                     truststorePass="changeit"
                     truststoreType="jks"
                     sslProtocol="TLS"
                     protocols="TLSv1.2"
                     honorCipherOrder="true"
                     ciphers
   ...
          <Certificate certificateKeystoreFile="./certs/[server hostname].jks"
                       certificateKeystorePassword="changeit"
                       certificateKeystoreType="jks"
                       certificateKeyAlias="[server hostname]" />
      </SSLHostConfig>
    </Connector>
-->
```

   (In your installation, [server hostname] is replaced with the actual hostname of your server.)
4. Remove the lines with the comment markers <!-- and --> .
5. Update the **certificateKeystoreFile** parameter with the name of the keystore file containing the server certificate and private key.
6. Set the **certificateKeystorePassword** parameter to the password for the keystore file containing the server certificate and private key.
7. Set the **certificateKeystoreType** parameter to jks for a Java keystore or pkcs12 for a PKCS #12 keystore.
8. If the keystore contains more certificates than the server certificate, the **certificateKeyAlias** parameter must be set to the alias for the server certificate and private key.
9. Unless you will enable X.509 client certificate authentication, remove the **truststoreFile**, **truststorePass**, and **truststoreType** parameters.
10. To disable unencrypted HTTP traffic, follow these steps:
    a. Locate the section containing the default HTTP connector:

```
<Connector port="[HTTP port]"
maxHttpHeaderSize="65536"
connectionTimeout="30000"
enableLookups="false"
```

```
URIEncoding="UTF-8"
disableUploadTimeout="true"
server="TIBCO Spotfire Server"
compression="on"
compressibleMimeType="text/html,text/xml,text/plain,text/css,application/
json,application/javascript,image/svg+xml,application/xml"
acceptorThreadCount="2"
keepAliveTimeout="30000"
maxKeepAliveRequests="-1"
maxThreads="2000" />
```

(In your installation, [HTTP port] is replaced with the HTTP port of your server.)

b. Add comment markers <!-- and --> around the HTTP connector configuration:

```
<!--
<Connector port="[HTTP port]"
maxHttpHeaderSize="65536"
connectionTimeout="30000"
enableLookups="false"
URIEncoding="UTF-8"
disableUploadTimeout="true"
server="TIBCO Spotfire Server"
compression="on"
compressibleMimeType="text/html,text/xml,text/plain,text/css,application/
json,application/javascript,image/svg+xml,application/xml"
acceptorThreadCount="2"
keepAliveTimeout="30000"
maxKeepAliveRequests="-1"
maxThreads="2000" />
-->
```

11. Start Spotfire Server.

Node manager installation

To be able to run services, you must first install and trust one or several node managers, depending on the expected workload. Node managers should not be installed on computers that are running Spotfire Server.

Currently the Windows-based node manager is capable of running services with four different capabilities: Spotfire Web Player, Spotfire Automation Services, the TERR service, and Spotfire Service for Python.

The Linux-based node manager can run only the TERR service and Spotfire Service for Python.

The installation of the Windows node manager creates a Windows service that runs as the LocalSystem account.

If you change the node manager service account, make sure that the account is a local administrator and that it has read and write access to the node manager installation directory and subdirectories.

There are two principal ways to install a Windows node manager:

- In an interactive installation, you run the `nm-setup.exe` file and then use the administrative tools in Spotfire Server to trust the node and install services and service instances. This is the most common method. For details, see Installing a node manager interactively.
- In a silent installation, you run the installer from the command line. For details, see Installing a node manager silently.

To install a Linux node manager, you have the option of an RPM installation or a tarball installation. For details, see Installing a node manager (RPM Linux) or Installing a node manager (tarball Linux).

For administrators of large implementations who want to be able to quickly scale their Spotfire system as necessary, an automated method of installing and configuring services and service instances is available. For details, see Automatically installing services and instances.

For more information, see Nodes and services introduction.

Installing a node manager (interactively on Windows)

To make Spotfire Web Player, Spotfire Automation Services, the TERR service, and Spotfire Service for Python available to end users, you first must install a node manager. A Spotfire implementation can contain several nodes, but each one must be installed on a different computer.

Prerequisites

- Spotfire Server is installed and running.
- In the firewall of the computer on which you are installing the node manager, open the ports that will be used for the node manager and the services. (See step 5 below for information on how these ports are used.)

This procedure describes an interactive installation, using the installation wizard. Alternatively, you can run a silent installation from the command line; for details, see Installing a node manager (silently on Windows).

Procedure

1. Double-click `nm-setup.exe`.

   You might be prompted to install Microsoft .NET Framework at this point.

2. On the installation wizard Welcome page, click **Next**.
3. On the License page, read the agreement, select **I accept**, and then click **Next**.
4. On the Destination Folder page you can change the location if you want to, and then click **Next**.
5. On the Node Manager Ports page, enter numbers (or leave the defaults) for the following ports:

   - **Node Manager registration port**—The port that is used to set up secure internal communication channels.

     In a production environment, it is not advisable to run the node manager and the Spotfire Server on the same computer. However, if you are installing the node manager on the same computer as the server, this port must be different than the Spotfire Server backend communication port.

   - **Node Manager communication port (TLS)**—The port that is used for secure (TLS) communication within the implementation.

     If you are installing the node manager on the same computer as Spotfire Server, this port must be different than the Spotfire Server backend communication port. The default for the Spotfire Server port is 9443.

   The selected ports must be available and not blocked by a firewall.

   To check whether a port is in use, on a command line enter `netstat -na`.

6. Click **Next**.
   The Spotfire Server page opens.

7. On the Spotfire Server page, enter the following information, and then click **Next**.

   These values must match the values you used when installing the Spotfire Server files.

   - **Server name**—The hostname of Spotfire Server.

     Valid hostnames can only contain alphabetic characters, numeric characters, hyphens, and periods.

   - **Server backend registration port**—The registration port that you specified during Spotfire Server installation.
   - **Server backend communication port (TLS)**—The backend communication port that you specified during Spotfire Server installation.
8. On the Network Names page, select the computer names that can be used by backend trust. In general you can leave all the listed names as they are.
9. On the Ready to Install page, click **Install**.
10. Click **Finish** when done.

What to do next

After the installation wizard finishes running, you must start the new node manager manually; see Starting or stopping a node manager (as a Windows service).

Installing a node manager (silently on Windows)

To make Spotfire Web Player, Spotfire Automation Services, the TERR service, and Spotfire Service for Python available to end users, you first must install a node manager. A Spotfire implementation can contain several nodes, but each one must be installed on a different computer.

Prerequisites

- Spotfire Server is installed and running.
- In the firewall of the computer on which you are installing the node manager, open the ports that will be used for the node manager and the service instances. The default ports are listed in the "Silent installation parameters" table, below.

To use the interactive installation wizard instead of the command-line installation, see Installing a node manager interactively.

Procedure

1. Open a command line as an administrator.
2. Replace the italicized entries in the following script:

```
nm-setup.exe INSTALLDIR="<node manager installation dir>"
 NODEMANAGER_REGISTRATION_PORT=83 NODEMANAGER_COMMUNICATION_PORT=84
 SERVER_NAME=SpotfireServerName SERVER_BACKEND_REGISTRATION_PORT=81
```

```
SERVER_BACKEND_COMMUNICATION_PORT=82 NODEMANAGER_HOST_NAMES=NodeManagerHostNames
NODEMANAGER_HOST=NodeManagerHost -silent -log "C:\Users\user\Log file.log"
```

Include only those parameters whose default values you are changing. The default values are listed in the table below.

Silent installation parameters

Parameter	Description
INSTALLDIR	The node manager installation directory.
NODEMANAGER_REGISTRATION_PORT	Node manager registration port (Default: 9080) nodemanager.properties: nodemanager.cleartext.port • Port used for initial setup of internal secure communication channels. • Needs only be accessible from Spotfire Server(s). In a production environment, it is not advisable to run the node manager and the Spotfire Server on the same computer. However, if you are installing the node manager on the same computer as the server, this port must be different than the Spotfire Server backend communication port.
NODEMANAGER_COMMUNICATION_PORT	Node manager communication port (TLS) (Default: 9443) nodemanager.properties: nodemanager.port • Port used for secure (TLS) internal communication within the environment. • Needs only be accessible from Spotfire Server(s). If you are installing the node manager on the same computer as Spotfire Server, this port must be different than the Spotfire Server backend communication port.
SERVER_NAME	nodemanager.properties: nodemanager.supervisor • Must match the host name of the Spotfire Server. Valid hostnames can only contain alphabetic characters, numeric characters, hyphens, and periods.
SERVER_BACKEND_REGISTRATION_PORT	Server backend registration port (Default: 9080) nodemanager.properties: nodemanager.supervisor.cleartext.port • Must match the registration port specified in the Spotfire Server installation.
SERVER_BACKEND_COMMUNICATION_PORT	Server backend communication port (TLS): (Default: 9443) nodemanager.properties: nodemanager.supervisor.port • Must match the backend communication port specified in the Spotfire Server installation.

Parameter	Description
NODEMANAGER_HOST_NAMES	A comma-separated list of IP addresses, hostnames, and FQDN names that can be used by backend trust. These should be for the interface(s) on the computer where the node manager is installed.  Valid hostnames can only contain alphabetic characters, numeric characters, hyphens and periods.  If you do not enter any values, the installer automatically provides values. After installation, confirm that these are correct in the following file: `<node manager installation dir>\nm\config\nodemanager.properties`.
NODEMANAGER_HOST	The computer where the node manager is installed.

3. Run the installation script.

What to do next

After installation, you must start the new node manager manually; see Starting or stopping a node manager (as a Windows service).

Installing a node manager (RPM Linux)

To make the TERR service or Spotfire Service for Python available to end users, you have the option of installing a node manager on a Linux computer instead of a Windows computer. A Spotfire implementation can contain several nodes, but each one must be installed on a different computer.

If you have root access to the Linux computer on which you want to install the node manager, you can use the RPM-based installer. If you do not have root access, use the tarball installer instead; for details, see Installing a node manager (tarball Linux).

Prerequisites

- Spotfire Server is installed and running.
- In the firewall of the computer on which you are installing the node manager, open the ports that will be used for the node manager and the services. The default ports are listed in the "Installation parameters" table, below.

Procedure

1. Open a terminal and go to the directory that contains the node manager installation file.
2. Enter the following command to install the node manager:

```
rpm -ivh tss-nm-<version number>.x86_64.rpm
```

A successful execution of the command produces text similar to this:

```
Preparing...
Updating / Installing...
    l:tss-nm-<version>-1-1
You must now execute /opt/tibco/tsnm/<version number>/configure to complete the
 configuration.
```

By default, the node manager is installed in the following directory: `/opt/tibco/tsnm/<version number>`.

3. Replace the italicized parameters in the following post-installation script; the parameters are defined in the table below. Then run the script.

Alternatively, you can run the script without any parameters. In this case you will be prompted for the missing information.

```
<node manager installation dir>/<version number>/configure -
m NODEMANAGER_REGISTRATION_PORT -c NODEMANAGER_COMMUNICATION_PORT -s SERVER_NAME
 -r SERVER_BACKEND_REGISTRATION_PORT -b SERVER_BACKEND_COMMUNICATION_PORT -
n NODEMANAGER_HOST_NAMES
```

Include only those parameters whose default values you are changing, in addition to the server name. The default values are listed in the table below.

Installation parameters

Parameter	Description
NODEMANAGER_REGISTRATION_PORT	Node manager registration port (Default: 9080) • Port used for initial setup of internal secure communication channels. • Needs only be accessible from Spotfire Server(s).  Running the node manager and the Spotfire Server on the same computer is not a supported configuration. However, if you decide to install the node manager on the same computer as the server, this port must be different than the Spotfire Server backend registration port.
NODEMANAGER_COMMUNICATION_PORT	Node manager communication port (TLS) (Default: 9443) • Port used for secure (TLS) internal communication within the environment. • Needs only be accessible from Spotfire Server(s).  Running the node manager and the Spotfire Server on the same computer is not a supported configuration. However, if you decide to install the node manager on the same computer as the server, this port must be different than the Spotfire Server backend communication port.
SERVER_NAME	• Must match the host name of the Spotfire Server.  Use the FQDN (Fully Qualified Domain Name).
SERVER_BACKEND_REGISTRATION_PORT	Server backend registration port (Default: 9080) • Must match the registration port specified in the Spotfire Server installation.
SERVER_BACKEND_COMMUNICATION_PORT	Server backend communication port (TLS): (Default: 9443) • Must match the backend communication port specified in the Spotfire Server installation.

Parameter	Description
NODEMANAGER_HOST_NAMES	A comma-separated list of IP addresses, hostnames, and FQDN names that can be used by backend trust. These should be for the interface(s) on the computer where the node manager is installed. Valid hostnames may only contain alphabetic characters, numeric characters, hyphens and periods. If you do not enter any values, the installer automatically provides values. After installation, confirm that these are correct in the following file: `<node manager installation dir>/<version number>/nm/config/nodemanager.properties`.

For more information about ports and how they are used, see Ports and firewall configuration.

Text similar to the following appears on the command line:

```
Post install configuration of TIBCO Spotfire Server Node Manager <version number>
 successful.
```

What to do next

After installation, start the new node manager; see Starting or stopping a node manager (Linux).

Installing a node manager (tarball Linux)

To make the TERR service or Spotfire Service for Python available to end users, you have the option of installing a node manager on a Linux computer instead of a Windows computer. A Spotfire implementation can contain several nodes, but each one must be installed on a different computer.

If you have root access to the Linux computer on which you want to install the node manager, you can use the RPM-based installer; for details, see Installing a node manager (RPM Linux). If you do not have root access, use the tarball installer instead.

When installing a node manager on a Red Hat Linux computer with Security-Enhanced Linux enabled, it is recommended to use the RPM-based installer over the tar.gz file. If you choose to use the TAR file anyway, it is recommended to use `/opt/tibco/<subfolder>` as the installation directory.

Prerequisites

- Spotfire Server is installed and running.
- In the firewall of the computer on which you are installing the node manager, open the ports that will be used for the node manager and the services. The default ports are listed in the "Installation parameters" table, below.

Procedure

1. Open a terminal and go to the directory that contains the node manager installation file.
2. Unpack and run the TAR file by entering the following command:

   ```
   tar xzf tss-nm-<version number>.x86_64.tar.gz
   ```

   The node manager is installed in the directory where you ran the command. A successful execution of the command does not result in any confirmation text.
3. Go to the new node manager directory; its name will be `tss-nm-<version number>.x86_64`.

4. Replace the italicized parameters in the following post-installation script; the parameters are defined in the table below. Then run the script in the directory where the TAR file was unpacked.

Alternatively, you can run the command (`./configure`) without any parameters. In this case you will be prompted for the missing information.

```
./configure -m NODEMANAGER_REGISTRATION_PORT -c NODEMANAGER_COMMUNICATION_PORT -
s SERVER_NAME -r SERVER_BACKEND_REGISTRATION_PORT -b SERVER_BACKEND_COMMUNICATION_PORT -
n NODEMANAGER_HOST_NAMES
```

Include only those parameters whose default values you are changing, in addition to the server name. The default values are listed in the table below.

Installation parameters

Parameter	Description
`NODEMANAGER_REGISTRATION_PORT`	Node manager registration port (Default: 9080) • Port used for initial setup of internal secure communication channels. • Needs only be accessible from Spotfire Server(s).  Running the node manager and the Spotfire Server on the same computer is not a supported configuration. However, if you decide to install the node manager on the same computer as the server, this port must be different than the Spotfire Server backend registration port.
`NODEMANAGER_COMMUNICATION_PORT`	Node manager communication port (TLS) (Default: 9443) • Port used for secure (TLS) internal communication within the environment. • Needs only be accessible from Spotfire Server(s).  Running the node manager and the Spotfire Server on the same computer is not a supported configuration. However, if you decide to install the node manager on the same computer as the server, this port must be different than the Spotfire Server backend communication port.
`SERVER_NAME`	• Must match the host name of the Spotfire Server.  Use the FQDN (Fully Qualified Domain Name).
`SERVER_BACKEND_REGISTRATION_PORT`	Server backend registration port (Default: 9080) • Must match the registration port specified in the Spotfire Server installation.
`SERVER_BACKEND_COMMUNICATION_PORT`	Server backend communication port (TLS): (Default: 9443) • Must match the backend communication port specified in the Spotfire Server installation.

Parameter	Description
NODEMANAGER_HOST_NAMES	A comma-separated list of IP addresses, hostnames, and FQDN names that can be used by backend trust. These should be for the interface(s) on the computer where the node manager is installed.  Valid hostnames may only contain alphabetic characters, numeric characters, hyphens and periods.  If you do not enter any values, the installer automatically provides values. After installation, confirm that these are correct in the following file: `<node manager installation dir>/<version number>/nm/config/nodemanager.properties`.

For more information about ports and how they are used, see Ports and firewall configuration.

The following message appears on the command line:

```
Post install configuration of TIBCO Spotfire Server Node Manager <version number>
 successful.
```

5. In the directory that contains the `configure-boot` file, enter the following command:

```
./configure-boot
```

The following message appears on the command line:

```
TIBCO Spotfire Server Node Manager <version number> has been successfully configured to
 start on system boot.
```

It will also be possible to start and stop the node manager from the command line.

What to do next

After installation, start the new node manager; see Starting or stopping a node manager (Linux) or Manually starting or stopping a node manager (tarball Linux). The second procedure does not require root user privileges.

Starting or stopping a node manager (as a Windows service)

Start or stop the node manager Windows service from the Control Panel on the node manager computer.

Procedure

1. Log in as an administrator to the computer on which the node manager is installed.
2. Go to **Control Panel** > **Administrative Tools** > **Services** and then, in the Services dialog, locate and select the service called **TIBCO Spotfire Node Manager**.
3. To the left of the services list, click **Start** in the phrase "Start the service" to start the node manager Windows service.

 To stop the service, click **Stop** to the left of the services list.

Result

"Running" appears in the Status column.

What to do next

After starting a node manager for the first time, you must indicate to the server that you "trust" it; see Trusting a node.

Starting or stopping a node manager (Linux)

If you have root user privileges, you can start and stop the node manager on a Linux computer by using this procedure.

If you used the tarball installation method and you do not have root user privileges, see Manually starting or stopping a node manager (tarball Linux).

Prerequisites

You have installed the node manager; for instructions, see Installing a node manager (RPM Linux) or Installing a node manager (tarball Linux).

Procedure

1. Log in as root or run with sudo `-s`.
2. Enter the following command: `/etc/init.d/tss-nm-<version number> start`

To stop the node manager, enter the following command: `/etc/init.d/tss-nm-<version number> stop`

What to do next

After starting a node manager for the first time, you must indicate to the server that you "trust" it; see Trusting a node.

Manually starting or stopping a node manager (tarball Linux)

After completing a tarball Linux installation of a node manager, you can manually start the node manager. This procedure makes it possible to start, stop, and run the node manager as a user other than root.

Prerequisites

You have installed the node manager using the tarball method; see Installing a node manager (tarball Linux).

Procedure

1. Log in to the Linux computer where the node manager is installed.
2. Enter the following command: `<node manager installation dir>/boot/inittss-nm-rh start`

To stop the node manager, enter the following command: `<node manager installation dir>/boot/inittss-nm-rh stop`

What to do next

After starting a node manager for the first time, you must indicate to the server that you "trust" it; see Trusting a node.

Trusting a node

After installing the node manager, you must indicate in Spotfire Server that you trust the node.

Prerequisites

- You have followed the appropriate procedure in the Node manager installation section.
- Both Spotfire Server and the newly-installed node manager are running.

Procedure

1. Log in to Spotfire Server. (For instructions on accessing the server, see Opening Spotfire Server.)
2. Click **Nodes & Services**, and then click the **Untrusted nodes** tab.
3. Under **Untrusted nodes**, select the check box next to the new node manager and then click **Trust nodes**.
4. In the "Trust node" dialog, click **Trust**.

Result

After a pause, the new node appears on the **Your network** page when you select the **Nodes** view.

What to do next

Service installation on a node on page 142

Automatically trusting new nodes

To speed up the process of adding nodes to your Spotfire implementation, you can configure the system so that all new nodes are automatically trusted by Spotfire Server, or you can limit the automatic trust to specific nodes. In combination with the automatic process for installing services and instances, administrators of large Spotfire implementations in private sub-nets can quickly scale up their system as needed.

Prerequisites

- Spotfire Server is installed and running.
- In the firewall of the computer(s) on which you are installing the node manager, open the ports that will be used for the node manager and the services.

Procedure

1. Open a command line and export the active server configuration (the `configuration.xml` file) by using the export-config command; for additional information, see Executing commands on the command line.
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=security.trust.auto-trust.enabled --value=true
   ```

   This sets up automatic trust for all new nodes in the Spotfire implementation.

3. Optional: If you want to limit automatic trust to certain nodes, do one of the following:
   - To allow one specific node to be automatically trusted, enter one of the following commands:
     - ```
       config set-config-prop --name=security.trust.auto-trust.allowed-hosts-
       config.allowed-hosts.allowed-host --value=example.com
       ```
       where *example.com* is the hostname of the node that will be automatically trusted.
     - ```
       config set-config-prop --name=security.trust.auto-trust.allowed-hosts-
       config.allowed-ip-regexps.allowed-ip-regexp --value=203\.0\.113\.1
       ```
       where *203\.0\.113\.1* is a regular expression for the IP address of the node that will be trusted.
   - To allow several specific nodes to be automatically trusted, do the following:
     a. Open the `configuration.xml` file in an XML editor or a text editor and locate the `<auto-trust>` section.
     b. Enter an edited version of the following code under `<enabled>true</enabled>`:
     ```
     <allowed-hosts-config>
                 <allowed-hosts>
                   <allowed-host>host1.example.com</allowed-host>
                   <allowed-host>host2.example.com</allowed-host>
                 </allowed-hosts>
                 <allowed-ip-regexps>
                   <allowed-ip>203\.0\.113\.1</allowed-ip>
                   <allowed-ip>203\.0\.113\.2</allowed-ip>
                 </allowed-ip-regexps>
         </allowed-hosts-config>
     ```
     where *hostn.example.com* is the hostname of a node that will be trusted, and *203\.0\.113\.n* is a regular expression for the IP address of a node that will be trusted. These lines can be repeated as often as necessary.

     You can also specify a range of regular expressions. The following example allows any IP address between 203.0.113.0 and 203.0.113.255:
     ```
     203\.0\.113\.\d{1,3}
     ```
     c. Save and close the configuration file.
4. Import the configuration file back to the Spotfire database by using the import-config command.
5. Restart the Spotfire Server service.

Result

When a new node that is enabled for auto-trust comes online and requests authorization from Spotfire Server, the server trusts the node automatically.

Automatically installing services and instances

To quickly and automatically add services and instances to your Spotfire implementation whenever you add and trust a new node, you can prepare a node template file that is triggered when a new node manager comes online and is trusted. This method is most appropriate for large and growing Spotfire implementations.

If you are configuring an automated deployment in a private subnet, you may also want to automatically trust nodes; for details, see Automatically trusting new nodes.

Prerequisites

- Spotfire Server is up and running.
- In the firewall of the computer on which you are installing the node manager, open the ports that will be used for the service instances.
- You have deployed client packages to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Procedure

1. Install and start the node manager(s) but do not trust them; for instructions, see Node manager installation.
2. Open a text editor, or a more advanced tool capable of detecting syntax problems in JSON text, and create a JSON file that contains the following code:

```
{
   "services" : [ {
           "capability" : "WEB_PLAYER",
           "deploymentArea" : "Production",
           "configuration" : "Web Player Configuration",
           "customPrefix" : "Prefix",
           "resourcePool" : "Pool A",
           "instances" : 2,
           "port" : 9501
           }
    ],
    "strict" : "false"
}
```

3. Edit the default parameters as necessary:

Parameter	Description
capability	The service to install. Current options for Windows node managers are WEB_PLAYER, AUTOMATION_SERVICES, TERR, and PYTHON. For Linux node managers, the TERR service and Spotfire Service for Python are available.
deploymentArea	Name of an existing deployment area.
configuration	Name of an existing configuration (default or otherwise) that is available in the deployment area for the service being deployed. For information on creating new service configurations, see Manually editing the service configuration files.
customName	Name of the new service. If present, this setting overrides any customPrefix setting. This parameter is optional.
customPrefix	Text to add before the name of the service. For example, if the customPrefix value is "Finance Dept.", the new Spotfire Web Player name will be "Finance Dept. Web Player". This parameter is optional.
resourcePool	For Spotfire Web Players, the name of a resource pool that the new instances will join. If the named resource pool does not exist, Spotfire Server creates it. This parameter is optional.
instances	Number of service instances to create. If no number is specified, only the service is created. This parameter is optional.
port	Communications port that the instances should use. This parameter is optional.

Parameter	Description
strict	Changing this parameter to "true" means that the installation will fail if any of the following parameters are not specified or are incorrect: • capability • deployment area • configuration

The text between the square brackets can be repeated as often as necessary in the file to create the required services and instances.

4. Name the file `default.conf` and place it in the following directory: `<node manager install directory>/nm/config/`
5. Trust the node manager; for instructions, see Trusting a node.

Result

The services specified in the `default.conf` file are installed and the service instances start running.

After the file is processed, the file's name changes to `default.bak`.

What to do next

For information on the remaining setup tasks, see Post-installation steps.

Login behavior configuration

You can configure various aspects of the Spotfire login dialog.

These are the behaviors that are configurable:

- If the login dialog should be displayed.
- If users should be allowed to work offline or if they always must log in.
- If users can select "Save my login information" in the login dialog and store the login information for future automatic login.
- If users should be forced to log in after working offline for a certain number of days.
- If you want an RSS feed to be shown in the login dialog.
- If users should be able to enter their own credentials in the login dialog.

To configure the login dialog, use the command config-login-dialog.

To change the look and feel of the login dialog and other Spotfire windows, see the TIBCO Spotfire Cobranding help.

For cobranding to work on a Linux system, cabextract must be installed.

Enabling an RSS feed in the Spotfire login dialog

Spotfire Server can be configured to display messages to end users in the login dialog, such as news of upcoming scheduled maintenance. One option is to specify a path to an `rss.xml` file that is located on a Spotfire Server; in this case the XML file is updated manually. The other option is to specify the URL to an external RSS feed.

Procedure

1. If you are using an `rss.xml` file that you will update manually, copy the file to the following directory: `<server install dir>\tomcat\webapps\spotfire`.
2. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
3. On the command line, use the config-login-dialog command to set up the feed.

   Make sure that the specified RSS feed complies with the standard RSS 2.0 specification, and that the source is available to the end users' clients.

   To enable all users see the news in the login dialog, set the display behavior setting (`-s value`) to "always". The login dialog will be shown to all users, even if they opt to save their login credentials for automatic login.

   Example using a relative URL on the Spotfire Server:

   ```
   config config-login-dialog -s always -R "/spotfire/rss.xml"
   ```

4. Import the configuration file back to the Spotfire database by using the import-config command.
5. Restart the Spotfire Server service.

Service installation on a node

After installing and trusting a node manager, you configure and install services and service instances on the node.

For each service you install on the node, select a capability, and the number of instances for that service, Spotfire Web Player, Spotfire Automation Services, the TERR service or Spotfire Service for Python.

The TERR service and Spotfire Service for Python can have only one service instance each.

For more information about installation of the various services, see its documentation:

- Installing Spotfire Web Player instances
- Installing Spotfire Automation Services instances
- TIBCO® Enterprise Runtime for R - Server Edition
- TIBCO® Spotfire Service for Python

The services are automatically set up with a default configuration. You can edit the default configuration files manually to create your own service configurations. For more information on how to manually configure the services, see Manually editing the service configuration files.

Preconfiguring Spotfire Web Player services (optional)

You can prepare one or several Spotfire Web Player configurations to apply to new services as you create them. This gives you access to an extended set of Spotfire Web Player options, and simplifies the task of setting up a group of services with identical properties.

Prerequisites

The Spotfire client distribution file (.sdn file) has been deployed to the server; for instructions see Deploying client packages to Spotfire Server.

Procedure

- Follow the steps in Manually editing the service configuration files.

Result

When you install a new Spotfire Web Player, you can select the customized configuration.

Installing Spotfire Web Player instances

After installing and authorizing a node manager, you install the Spotfire Web Player service and indicate the number of Spotfire Web Player instances that you want to make available. The Spotfire Web Player instances can then be accessed on any computer in the network.

Prerequisites

- You have installed and authorized a node manager; for instructions, see Installing a node manager interactively and Trusting a node.
- Spotfire Server and the node manager are up and running.
- You have deployed client packages to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.

Do not install the Spotfire Web Player on a node that contains other services, such as the TERR service or Spotfire Service for Python.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. Under **Select a view**, select **Nodes**, and then select the node to which you want to add the Spotfire Web Player service. There should be a green circle with a check mark next to the selected node.
3. In the lower-right pane, click **Install new service**.
4. Make your selections in the "Install new service" dialog:
   a) Under **Deployment area**, select the area you are using.

      Administrators generally create a Test deployment area to use as a staging server.

   b) Under **Capability**, select **Web Player**.
   c) Under **Configuration**, select the service configuration that you want to apply to the service.

      Spotfire Server contains a default service configuration that you can replace later. If you want to prepare a configuration file ahead of time, see Preconfiguring Spotfire Web Player services.

   d) Under **Number of instances**, enter the number of instances of the service that you want to make available. For more information, see Multiple service instances on one node.
   e) Under **Port**, you can change the default of 9501 if you want to.
   f) Enter a name for this service.
5. Click **Install and start**.
   To view the progress of the installation, click the **Activity** tab.

What to do next

- If applicable, install Spotfire Automation Services; for instructions, see Installing Spotfire Automation Services instances.

- For information on the remaining setup tasks, see Post-installation steps.

Multiple Web Player instances on one node

Adding more than one Spotfire Web Player instance could be beneficial, particularly on large computers with NUMA architecture.

For failover reasons, it is recommended to have more than one instance in your environment. However, the instances can be on the same node.

There are two main reasons for adding more service instances on the same node:

- If there are unstable analyses that are suspected to result in issues for the process, these analyses can be routed to one dedicated service instance using file routing rules. This isolates the analyses from other instances.
- A very large .NET heap may lead to long running garbage collections, blocking normal execution. By distributing analyses that lead to a large .NET memory footprint over more than one service instance, the .NET heap becomes smaller, which leads to quicker garbage collections.

There are two reasons to avoid using too many service instances:

- Each service instance requires some overhead, mostly in terms of memory usage but also some CPU usage.
- There is no data or document sharing between service instances.

You may want to experiment with fewer or more service instances, especially on large computers.

Preconfiguring Spotfire Automation Services (optional)

You can prepare one or several Spotfire Automation Services configurations to apply to new services as you create them. This gives you access to an extended set of Spotfire Automation Services options, and simplifies the task of setting up a group of services with identical properties.

Prerequisites

The Spotfire client distribution file (.sdn file) has been deployed to the server; for instructions see Deploying client packages to Spotfire Server.

Procedure

- Follow the steps in Manually editing the service configuration files.

Result

When you install a new Spotfire Automation Services, you can select the customized configuration.

Installing Spotfire Automation Services instances

After installing and authorizing a node manager, you can install Spotfire Automation Services and indicate the number of instances of this service that you want to make available. Spotfire Automation Services can then be accessed on any computer in the network.

All users that execute Automation Services jobs on the server, using the Job Builder or the Client Job Sender, must be members of the group Automation Services Users.

Prerequisites

- You have installed and authorized a node manager; for instructions, see Installing a node manager and Trusting a node.

- Spotfire Server and the node manager are up and running.
- You have deployed client packages to Spotfire Server; for instructions, see Deploying client packages to Spotfire Server.
- You have assigned *licenses* required by the Automation Services jobs to the automationservices@SPOTFIRESYSTEM user, which is the account used to execute the jobs on the service instance. You have also given the user the appropriate read and write access to the library for the data required by the Automation Services tasks, and the job files.

For a description of the licenses, see the License feature reference.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. In the **Nodes** view, select the node to which you want to add the Spotfire Automation Services service. There should be a green circle with a check mark next to the selected node manager. The words **Installed services** followed by the name of the node manager are displayed in the lower-right pane of the window.
3. Click **Install new service**.
4. Make your selections in the "Install new service" dialog:
   a) Under **Deployment area**, select the area you are using.

Administrators generally create a Test deployment area to use as a staging server.

   b) Under **Capability** select **Automation Services**.
   c) Under **Configuration**, select the service configuration that you want to apply to the service.

Spotfire Server contains a default service configuration that you can replace later. If you want to prepare a configuration file ahead of time, see Preconfiguring Spotfire Automation Services.

   d) Under **Number of instances**, enter the number of instances of the service that you want to make available.
   e) Under **Port**, you can change the default of 9501 if you want to.
   f) Enter a name for this service.
5. Click **Install and start**.
   To view the progress of the installation, click the **Activity** tab.

What to do next

For information on the remaining setup tasks, see Post-installation steps.

Automation Services Job Builder and Client Job Sender

Spotfire Automation Services includes the Job Builder tool for creating multi-step jobs, and the Client Job Sender tool for running jobs that are created in the Job Builder.

The Job Builder requires no installation. It is accessed from Spotfire Analyst.

The easiest way to schedule and monitor Automation Services jobs is from the Automation Services area of the Spotfire administration interface. For details, see Automation Services job scheduling.

To run jobs using the Client Job Sender, the tool must be installed and then configured to communicate with the Spotfire Server. The job execution schedule is set by using Windows Task Scheduler. For more information, see the TIBCO Spotfire Automation Services User's Manual.

Spotfire Analyst installation

You can install Spotfire Analyst either interactively, by running the `setup-shared-computer-<version>.exe` file, or silently, either on the command line or by using a software distribution system.

Installing Spotfire Analyst silently (by using the command line)

You can silently install Spotfire Analyst (the Spotfire Windows client) from the command line or by using a software distribution system.

Procedure

1. Open a command line as an administrator.
2. Replace the parameters in the following script:

```
setup-shared-computer-<version>.exe -silent -log <logfile> SERVERURL=http[s]://
<hostname>:[<port>]/ INSTALLDIR="<location>"
```

   where `-log` creates a verbose installations log. The directory and file name for the log are specified after the "-log" text.

   Silent installation parameters

Parameter	Description
SERVERURL	The server to which Spotfire Analyst will connect the first time that the client is started.
INSTALLDIR	The installation directory for Spotfire Analyst. If not specified, the program will be installed in the standard %programfiles% folder.

3. Run the installation script.

Example

```
setup-shared-computer-10.3.0.exe -silent -log C:\AnalystInstallLog.txt
 SERVERURL=https:\\spotfireserver.example.com:8080 INSTALLDIR="C:\Program
 Files (x86)"
```

Installing Spotfire Analyst silently (by using a software distribution system)

You can silently install Spotfire Analyst (the Spotfire Windows client) by using a `.msi` file, also known as a Windows Installer Package.

Procedure

1. Open a command line as an administrator.
2. Extract the `.msi` file from the .exe file by running the `setup-shared-computer-<version>.exe` file and stopping at the first screen (the Welcome page).
   This unpacks the `.msi` file in the following location: `C:\ProgramData\Package Cache \{<GUID>}v<version>\ts-setup.msi`.
3. Copy the `ts-setup.msi` file to a different location on the computer, and then cancel the interactive installation that you started.

4. Replace the parameters in the following script:

```
msiexec /qn /i "<FULL PATH>\ts-setup.msi" SETUPEXEDIR="<location>" ALLUSERS=1
 SERVERURL=http[s]://<hostname>:[<port>]/ INSTALLDIR="<location>"
```

Silent installation parameters

Parameter	Description
FULL PATH	The full path to the `ts-setup.msi` file.
SETUPEXEDIR	Must be specified for the .msi package to work correctly. The location from which the install script will be run. It can be any directory to which the installation account has write access, for example, "%TEMP%".
SERVERURL	The server to which Spotfire Analyst will connect the first time that the client is started.
INSTALLDIR	The installation directory for Spotfire Analyst. If not specified, the program will be installed in the standard %programfiles% folder.

To see a progress bar during the installation, change "qn" to "qb".

5. Run the script.

Example

```
msiexec /qn /i "C:\Desktop\ts-setup.msi" SETUPEXEDIR="%TEMP%" ALLUSERS=1
 SERVERURL=https://spotfireserver.example.com:8080/ INSTALLDIR="C:\Program
 Files (x86)"
```

Upgrading or downgrading client packages

Newer and older versions of Spotfire Analyst client can be used to connect to the current version of Spotfire Server, in order to upgrade or downgrade the client packages. This makes it easy to ensure the same version is used by clients and the selected deployment area on the server.

Even though it might be possible to connect without upgrading or downgrading (a setting is available to enforce it on the Deployments & Packages page, see Adding software packages to a deployment area on page 245 for more information), it is recommended to run the same version of client and server in production environments. See System Requirements.

It is possible to downgrade all Spotfire installations after version 7.13 back to version 7.11 LTS. For example, if you are running 10.3 LTS, you could downgrade to 7.11, 7.12, 7.13, 10.0, 10.1 and 10.2. Note that if you downgrade to 7.12 you will not be able to downgrade from that version to 7.11 LTS, because the functionality to downgrade is not available in that version. First upgrade to a higher version than 7.12 and, from there, downgrade to 7.11 LTS if that is needed.

Sites

You can create multiple Spotfire environments that share the same Spotfire database, including the library and user directory. These environments, which are called sites, can be configured to reduce latency for multi-geographic deployments. Sites also enable the use of a variety of authentication methods, along with different user directories, within the same deployment.

Each site includes one or more Spotfire Servers along with their connected nodes and services. A site's servers, nodes, and services can only communicate within the site, but because the Spotfire database

is shared among the sites, all of the sites have access to the users, groups, and library in your Spotfire implementation.

If the site will contain more than one server, clustering must be enabled for that site; for more information, see Clustered server deployments.

You assign a Spotfire Server to a site when bootstrapping the server. You can change the assignment afterwards by following the instructions in Moving a server and its nodes to a different site. When you assign a Spotfire Server to a site, any nodes that are connected to the server are automatically included in the site.

As of Spotfire version 7.9, all upgraded servers and nodes belong to the Default site. To assign the upgraded components to a site that you created, use the procedure Moving a server and its nodes to a different site.

The potential reduced latency occurs between the servers and the service instances within a site, resulting in quicker manipulation of data that is already present in the site. To optimize the end-user experience, a best practice when configuring sites is to create scheduled updates so that data and analyses are downloaded from the database before users request them. For more information, see Scheduled updates to analyses.

These are typical uses of Spotfire sites:

- To route user requests from a particular office to the servers and nodes that are physically closest to that office. This reduces the impact of network latency between servers that are located in different geographic regions.
- To enable different authentication methods for different sets of users who share a Spotfire implementation. For example, internal users may use Kerberos authentication while external users such as customers and partners may use username and password authentication.

Administrators who oversee several sites can switch sites from the landing page of the administration interface.

In a deployment that contains sites, the following items are site specific and not shared with any other sites:

- Nodes
- Resource pools
- Schedules
- Scheduled updates and routing rules
- Scheduled Automation Services jobs
- Authentication can be configured to be site specific; see Setting different authentication methods and user directories for sites.
- Public address; set a site's public address when creating the site, or later by using the set-public-address command.

The following items are "global", so shared among all the sites in a deployment:

- Library
- User directory
- Groups
- Deployments
- Server configuration file
- Service configuration files
- LDAP synchronization

- Signing certificates
- Login page RSS feed

Creating sites

Sites are created on the command line, and then you assign a server to a particular site when you bootstrap the server. In the case of a server that has previously been installed and configured, use the **`set-site`** command to assign it to a site.

For general information about sites, see Sites.

Procedure

1. Open a command line as an administrator and go to the `<server installation directory>\tomcat\spotfire-bin` directory.
2. Run the create-site command.

It is recommended to specify the public address (the **`-a`** parameter) when creating a site. If you do not specify the public address now, you can do it later by using the set-public-address command.

Example

```
config create-site -s MySite -a https://server.example.com/
```

where:

`MySite` is the name of the site you create.

`https://server.example.com/` is the public address of the site (optional).

When using the default port (80 for HTTP, 443 for HTTPS), do not specify the port in the public address.

Setting different authentication methods and user directories for sites

You can configure the sites in your implementation to use different authentication methods and, if necessary, different user directories.

Prerequisites

You have created the sites; for instructions, see Creating sites.

For general information about sites, see Sites.

Procedure

1. On any server computer in the implementation, open a command line as an administrator and export the active configuration by using the export-config command. For additional information on using the command line, see Executing commands on the command line.

2. To set different authentication methods, do the following:
   a. To set the global authentication method, run the config-auth command without specifying a site.
   b. To set a different authentication method for a site, run the config-auth command, specifying the site.

   Example

   In this example, all of the sites will use LDAP authentication except for the "Tokyo" site, which will use Kerberos.

   ```
   config config-auth -a BASIC -l
   ```

   ```
   config config-auth -a KERBEROS -s Tokyo
   ```
3. If all the sites will not use the same user directory, run the config-userdir command in a similar manner.
4. Import the configuration file by using the import-config command.
5. Restart the servers.

Moving a server and its nodes to a different site

When moving a server and its nodes from one site to another site, you must edit the `nodemanager.properties` file for each node. This procedure should also be used to move upgraded servers and nodes from the Default site to a site that you created.

Prerequisites

You have created the site to which you want to assign the server; for instructions, see Creating sites.

For general information about sites, see Sites.

Procedure

1. Stop the server and its nodes. For instructions, see Start or stop Spotfire Server and Starting or stopping a node manager.
2. Assign the server to the new site by using the set-site command:
   a. On the computer that is running the server, open a command line as an administrator and go to the `<server installation directory>\tomcat\spotfire-bin` directory.
   b. Run the set-site command.

   Example

   ```
   config set-site -n 1234abcd-ab1-1a23-1234-ab1234c5678 -s Tokyo
   ```

   where:

   `-n value` is the ID of the server.

   `-s value` is the name of the site to which you want to assign the server.

   If you do not know the ID of the server, use the list-nodes command to find the IDs of all the servers and nodes in the environment.
3. Start the server. (Do not start the node managers.)

4. Do the following for each node that is connected to the server:

   a. Open the following file in a text editor or XML editor: `<node manager installation directory>\nm\config\nodemanager.properties`.

      Example of the `nodemanager.properties` file:

      ```
      #Supervisor changed
      #Wed Feb 16 22:27:19 CET 2017
      nodemanager.host.names=Comp_A,10.101.10.10
      nodemanager.communication.port=9443
      server.backend.registration.port=9080
      nodemanager.registration.port=9080
      nodemanager.host=
      server.name=Comp_12
      nodemanager.supervisor.known=Comp_C:9443-9080,Comp_D:9443-9080,Comp_12:9443-9080
      nodemanager.bundle.version=42.0.6127.7990
      server.backend.communication.port=9443
      ```

      The `nodemanager.supervisor.known` property lists the servers in the current site.

   b. Delete the line that begins with `nodemanager.supervisor.known`.

   c. Edit the `server.backend.registration.port`, `server.name`, and `server.backend.communication.port` to point to a Spotfire Server in the site to which you are moving.

      Example of the edited `nodemanager.properties` file:

      ```
      #Supervisor changed
      #Wed Feb 16 22:27:19 CET 2017
      nodemanager.host.names=Comp_A,10.101.10.10
      nodemanager.communication.port=9443
      server.backend.registration.port=9080
      nodemanager.registration.port=9080
      nodemanager.host=
      server.name=Comp_5
      nodemanager.bundle.version=42.0.6127.7990
      server.backend.communication.port=9443
      ```

   d. Save and close the file.

   e. Start the node manager.

   The `nodemanager.supervisor.known` property is added back into the `nodemanager.properties` file. It should contain the names of the servers in the new site.

5. In the administrative interface, verify that the node manager comes online in the correct site.

When you move a node, its service instances are removed from any resource pools they may have previously been assigned to.

Sites administration

Sites are administered in the same way as an ordinary Spotfire environment, with the difference that some features are global and some are site specific.

Because the Spotfire database is shared among all the sites, changes made in **Users & Groups** will be global, and affect all sites.

Communication between the Spotfire Server and the nodes only occurs within each site. For this reason, nodes, services, and routing are site specific and administered individually for each site. You can select which site to administer at the top of the Spotfire Server home page.

For general information about sites, see Sites.

Deleting sites

Sites are deleted on the command line. If the site contains servers and nodes, you must specify a site to move them to.

Procedure

1. Stop the servers and node managers in the site that you want to delete. For instructions, see Start or stop Spotfire Server and Starting or stopping a node manager.
2. Open a command line as an administrator and go to the `<server installation directory>\tomcat\spotfire-bin` directory.
3. Run the delete-site command.
4. Restart any servers and node managers that were in the site.

Example

```
config delete-site -s "East Coast" -i "Default"
```

where:

`East Coast` is the name of the site to delete.

`Default` is the name of the site to which you want to move the deleted site's servers and nodes.

Additional configuration

You can add to or change your Spotfire configuration by using the configuration tool or the command line, or by working directly in the configuration file.

Updating a server configuration in the configuration tool

You can change a Spotfire Server configuration by using the configuration tool.

If you cannot run the configuration tool on the Spotfire Server computer, see Running the configuration tool on a local computer.

Procedure

1. Open the configuration tool and sign in.
2. On the **Configuration** tab, make your changes.
3. Click **Save**.
4. Restart Spotfire Server.

Updating a server configuration on the command line

You can change a Spotfire Server configuration by running a series of commands on the command line.

Procedure

1. Open a command line.

2. Run the export-config command to export the configuration from the Spotfire database to a configuration file; for additional information, see Executing commands on the command line.

```
> config export-config configuration.xml
```

   where "configuration.xml" is optional and the `-f (--force)` option is not applied.

3. Update the configuration in the configuration file using selected commands. Example:

```
> config config-auth --configuration=configuration.xml --auth-method=BASIC --jaas-
database
```

   where "--configuration=configuration.xml" is optional.

4. Run the import-config command to import the updated configuration file into the Spotfire database. Example:

```
> config import-config --comment="Switched to BASIC authentication using the Spotfire
 Database authentication source" configuration.xml
```

   where "configuration.xml" is optional.

5. Restart the server(s).
6. Remove the `configuration.xml` file or restrict access to it.

Do not remove the `bootstrap.xml` file.

Manual configuration

Certain configuration properties in the Spotfire system are rarely used and cannot be set using commands. To use these properties you manually edit the server configuration. You may also want to work directly in the configuration to configure features that require complex commands, such as enabling several authentication options.

There are two ways to manually edit the configuration:

- Export the configuration to a `configration.xml` file and open the file in an XML or text editor.

  Spotfire Server configurations are stored in the Spotfire database and can be exported to a `configuration.xml` file for editing or sharing. The configuration settings can also be exported to file for backup purposes, to be imported into another cluster to set up multiple clusters with similar settings, or to be sent to TIBCO Support for inspection. For instructions, see Manually editing the server configuration in an XML or text editor.

- Open the Spotfire configuration tool and edit the configuration on the XML View page.

  The active configuration file can also be edited in the Spotfire configuration tool, without having to export the file. For instructions, see Manually editing the server configuration in the configuration tool.

Manually editing the service configuration files

The service configuration files give you access to options that are not available in the Spotfire Server administrative interface. You can use the default configuration files as a template to create and import as many customized service configurations as your Spotfire implementation requires. You can then apply the customized configurations to new or existing Spotfire Automation Services and Spotfire Web Player services.

- For information about configuring the TERR service, see TIBCO® Enterprise Runtime for R - Server Edition.

- For information about configuring the Spotfire Service for Python, see TIBCO® Spotfire Service for Python.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the location of the `config.bat` file (`config.sh` on Linux). The default location is `<server installation dir>/tomcat/spotfire-bin`.
2. On the command line, export the service configuration that you want to modify from Spotfire Server by using the export-service-config command. Specify the service's capability and the deployment area, and optionally the configuration name.

   By default, all new services receive a "Default" configuration. The properties of the default configuration cannot be changed, but you can edit the configuration files and import the resulting customized configuration with a specified name.

   If you are editing a service configuration that has been applied to an existing service, you must verify the name of the active service configuration before you export it. If the name of the active configuration is not "Default", you must specify the name in the **`export`** command.

   Example for exporting the "Default" Spotfire Automation Services configuration that is in the Production deployment area:

   ```
   config export-service-config --capability=AUTOMATION_SERVICES --deployment-
   area=Production
   ```

   Example for exporting a customized configuration:

   ```
   config export-service-config --config-name=AutomationServicesConfiguration
   ```

   The following configuration files are exported. By default, these files are saved to the `<server installation dir>\tomcat\spotfire-bin\config\root` directory.

   - `Spotfire.Dxp.Worker.Automation.config` (for Automation Services only)
   - `Spotfire.Dxp.Worker.Core.config`
   - `Spotfire.Dxp.Worker.Host.exe.config`
   - `Spotfire.Dxp.Worker.Web.config`
   - `log4net.config`
3. Edit the exported configuration files in a text editor or XML editor. For details about these files, see Service configuration files.
4. On the command line, import the customized configuration file back into Spotfire Server and name the configuration by using the import-service-config command.

   If the configuration to be imported was created from the default configuration, a name *must* be specified.

   If you are editing already customized configuration files, specifying a name when importing will create a new service configuration. If you import the changed customized configuration without the `--config-name` parameter, the old customized configuration will be replaced.

   ```
   config import-service-config --config-name=ServiceConfiguration
   ```

   When you install a new service or edit an existing one, you can select the customized configuration.

5. Optional: To activate the customized configuration for an existing service, run the following command on the command line:

```
config set-service-config --service-id=value --config-name=ServiceConfiguration
```

Use the list-services command to obtain the service ID.

Activating the configuration for a Spotfire Web Player service causes its web clients to restart.

Viewing the name of the active service configuration

You can view the name of a service's current configuration in the Nodes & Services section of Spotfire Server.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the **Your network** page, under **Select a view**, click **Nodes**, and then select the service whose configuration name you want to view.
3. In the upper-right pane of the page, in the service information list, **Configuration** is the second entry from the bottom:

Service configuration files

There are four files that are used to configure the Spotfire Web Player service and Spotfire Automation Services. Together, these files form service configurations that can be applied to individual services in your Spotfire implementation

- For information on working with these files, see Manually editing the service configuration files.
- For information about configuring the TERR service, see TIBCO® Enterprise Runtime for R - Server Edition.
- For information about configuring Spotfire Service for Python, see TIBCO® Spotfire Service for Python.

For information about the `log4net.config` file, see Web Player service logs.

- Spotfire.Dxp.Worker.Automation.config
- Spotfire.Dxp.Worker.Core.config
- Spotfire.Dxp.Worker.Host.exe.config
- Spotfire.Dxp.Worker.Web.config

Spotfire.Dxp.Worker.Automation.config file

This configuration file is used for configurations that are specific to Automation Services .

For information on working with this file, see Manually editing the service configuration files.

Setting	Defau value	Description
`<Spotfire.Dxp.Automation>`		
`<automation>`		
`maxWaitTimeForTaskBackgroundJobToFinishSeconds`	180	The number of seconds to wait for background thread execution to finish after the task finished executing.
`maxConcurrentJobs`	-1	The number of jobs that are allowed to execute in parallel. If 0 or less, this is set to the number of CPU cores on the machine.  The number of executing jobs can be less than the specified value if the service instance is exhausted. For more information, see `WebPlayer_AverageCpuLoadExhaustedLimit` in Spotfire.Dxp.Worker.Host.exe.config file.
`useKerberos`	False	Set to "True" to run Automation Services jobs as a specific Windows account when delegated Kerberos is enabled in the environment. If set to "False", jobs will be run using the node manager service account. To specify the Windows account, add the following section: `<kerberosIdentity userName="domain\username" password="password" />` and specify the account username and password.
`customAccount`	`""`	To use an existing account instead of automationservices@SPOTFIRESYSTEM, enter the username of the account, including the domain, here (`useKerberos` must be false).
`</automation>`		
`</Spotfire.Dxp.Automation>`		
`<Spotfire.Dxp.Automation.Framework>`		
`<security>`		

Setting	Default value	Description
`allowDeleteOfFilesModifiedLastMinutes`	30	The Send Email task can delete files after they have been sent. To avoid deleting files that should be kept, only files that have been created and modified in the timeframe specified in this setting can be deleted. The default value is 30 minutes. If set to "0", no files can be deleted. If set to "-1", all files can be deleted.
`<allowedFilePaths>`		
`allowAll`	True	By default, Automation Services tasks can read files from, and write files to any directory in the file system. Set this to `False` to only allow tasks to read from and write to directories specified in the `<allowedFilePaths>` section.  To be able to restrict the allowed paths for custom tasks, the custom tasks must use the validation function in the Automation Services API.

`<add path=""/>`		Add an `<add path=""/>` row for each directory the Automation Services tasks should be allowed to read from and write to. Paths can be relative to the Automation Service installation directory on the node, local paths, or network paths. For example:
		`<allowedFilePaths allowAll="false">`
		`  <add path=".\Temp\" />`
		`  <add path="C:\Temp\" />`
		`  <add path="\\MyServer\Spotfire Exported PDF\" />`
		`</allowedFilePaths>`
		  Added allowed paths are compared to all directories and files starting with what was added. For example, if you add `C:\Temp` as an allowed path, both the directory `C:\Temp\` and a file called `C:\Tempfile.txt` would be allowed. If you want to make sure that only a specific folder is allowed, add a backslash at the end, for example `C:\Temp\`.

`</allowedFilePaths>`		
`</security>`		
`</Spotfire.Dxp.Automation.Framework>`		
`<spotfire.dxp.automation.tasks>`		
`<smtp>`		
`port`	25	The port to use when connecting to the SMTP server.
`useTls`	False	Set to "True" to use Transport Layer Security (TLS) when connecting to the SMTP server.
`timeoutSeconds`	100	The maximum number of seconds before the Send command times out.

Setting	Default value	Description
useWindowsDefaultCredentials	False	Set to "True" to use the windows credentials of the account that executes the node manager when accessing the SMTP server. If username and password is set, this is not used.
username		The username to use when authenticating with the SMTP server.
password		The password to use when authenticating with the SMTP server.
useCertificates	False	Set to "True" to use client certificates when accessing the SMTP server.
storeLocation		The store location to take the certificate from [CurrentUser|LocalMachine].
storeName		The name of the store to take the certificate from [AddressBook|AuthRoot|CertificateAuthority|Disallowed|My|Root|TrustedPeople|TrustedPublisher].
serialNumber		The serial number of the certificate.
</smtp>		
<saveAnalysis>		
forceUpdateBehaviorManualWhenEmbeddingData	True	Set to "True" to force embedding of data function-based data sources, such as On-demand.
</saveAnalysis>		
<preferences>		
Spotfire.Automation.SendMail.SMTPHost		Specify the SMTP Host for Email Notification.
Spotfire.Automation.SendMail.FromAddress		Specify the From Address for Email Notification.
Spotfire.Automation.LibraryImport.TimeoutInSeconds	3600	Specify the timeout (seconds) for the library import operation for the Import Library task.
Spotfire.Automation.LibraryExport.TimeoutInSeconds	3600	Specify the timeout (seconds) for the library export operation for the Export Library task.
</preferences>		
</spotfire.dxp.automation.tasks>		

Spotfire.Dxp.Worker.Core.config file

This configuration file specifies settings for the service's communication with the Spotfire Server, and if sections in configuration files should be encrypted.

For information on working with this file, see Manually editing the service configuration files.

Setting	Default Value	Description
`<Spotfire.Dxp.Services.Settings`		
`httpLoggingEnabled="false"`	False	Decides whether troubleshoot logging of communication problems should be on or off. Should only be enabled upon request by Spotfire support.
`webSocketsEnabled="true">`	True	Defines whether or not web clients should use WebSockets in the communication with the service.  If you are going to analyze streaming data in web clients, this variable must be set to `true`.
`cookies autoTransfer=""`		Specify the cookies from the Spotfire Server that should be sent back on all requests in the format of a ; separated list, for example: "ARRAffinity;myCookie;myCookie2".
`<authentication hostsToAuthenticate="" />`		This setting is only applicable when system is setup to use delegated Kerberos. Specify a list of trusted sites/servers that should be allowed to authenticate using Windows credentials. The TIBCO Spotfire Server is automatically added to this list. Also, the top domain of the machine running this service is added to the list (`serv1.b.x.com` is added as `*.x.com`). Add other servers in the format of a ; separated list. To allow wildcard matches, start the host name with a star *. For example: `*.a.x.com;srv1.b.x.com;*.y.com;server3` This will match `<Anything>.a.x.com OR srv1.b.x.com OR <Anything>.y.com OR server3`.
`<Spotfire.Dxp.Worker.Host>`		
`<cryptography encryptConfigurationSections="true" protectSectionEncryptionProvider="DataProtectionConfigurationProvider" /> `		
`<cryptography>`		
`encryptConfigurationSections`	True	Set to `true` to encrypt sections of configuration files containing sensitive information.
`protectSectionEncryptionProvider`	DataProtectionConfigurationProvider	Name of the algorithm used when sections are encrypted.
`</cryptography>`		

Spotfire.Dxp.Worker.Host.exe.config file

Settings in this configuration file affect both Web Player services and Automation Services.

For information on working with this file, see Manually editing the service configuration files.

Setting	Default value	Description
`<Spotfire.Dxp.Data.Cxx.Properties.Settings>`		
`QueryCacheEntryMaxAge`	1.00:00:00 (1 day)	Use this setting to limit the lifetime for query cache data. A large value gives better response times when calculations can be reused, but the data engine query cache increases. For analyses in scheduled updates, it is a good idea to keep the cache as long as the update interval. 00:00:00 = infinity, 1.00:00:00 = 24 hours, 7.00:00:00 = 1 week, 01:00:00 = 1 hour, and so on.
`<Spotfire.Dxp.Web.Properties.Settings>`		
`ProxyUsername`		If you need to use proxy handling for communication from the Web Player service or Automation Services to Spotfire Server, and the proxy server uses username and password authentication, specify the username in the `value` tags.  To use these proxy authentication settings, you must also add a proxy section, including the proxy address, to setting `<system.net><defaultProxy>`
`ProxyPassword`		If the proxy server uses username and password authentication, specify the password in the value tags.
`TibcoSpotfireStatisticsServicesURLs`		A list of URLs to Spotfire Statistics Services.
`TibcoSpotfireStatisticsServicesUsernames`		A list of user names for each of the URLs.
`TibcoSpotfireStatisticsServicesPasswords`		A list of passwords for each of the user names and URLs.

Setting	Default value	Description

`DataAdapterCredentials`		If `WebConfig` is selected as authentication method for a data connector, you must add at least one credentials profile with username and password information for authentication.
		In the data connections that will use the credentials profile for authentication, you must specify the name of the credentials profile.
		You can add multiple profiles with different credentials.
		Credentials profile reference
		Each credentials profile entry should be in this format:
		`<entry profile="credentials_profile_name"><allowed-usages><entry server-regex="database\.example\.com" /></allowed-usages><username>my_username</username><password>my_password</password></entry>`
		`entry profile`
		The name of the credentials profile.
		The name is used to select the credentials profile for authentication in connection data sources.
		`allowed-usages`
		A list of allowed servers and connectors. You can use the credentials profile for authentication only in connections to the allowed servers, or with the allowed connectors.
		If `allowed-usages` is empty, you can use the credentials profile for authentication in connections to any server.
		Enter allowed servers as regular expressions, in the following format:
		`<entry server-regex="database\.example\.com" />`
		⚠ Make sure to specify the allowed servers as valid regular expressions. Values that are not valid regular expressions are ignored. If all servers are invalid, the credentials profile can be used in connections to any server.
		You can also enter allowed connectors. Then you can use the credentials profile for authentication in any connection

Setting	Default value	Description

		that you created with that connector. For example:
		`<entry connector-id="Spotfire.GoogleAnalyticsAdapter" />`
		You can also specify both a connector id and a server in one `allowed-usages` entry, to require a specific combination of connector and server. For example:
		`<entry connector-id="Spotfire.SqlServerAdapter" regex="database\.example\.com">`
		`username`
		The username to use for authentication with the data source.
		`password`
		The password to use for authentication with the data source.
		  Spotfire connectors only require that the user has read privileges in the database. When you create a credentials profile, a recommended practice is to use a database user that only has the minimum required privileges for reading the data that you want to analyze in Spotfire.

`WebPlayer_AverageCpuLoadExhaustedLimit`	90	If a service instance is in the Exhausted state, no new users are routed to that instance. Specify the CPU load limit, as a percentage, that sets the state of the instance to Exhausted. Set to `-1` to disable the Exhausted limit. Note that this setting is applicable to both Web Player services and Automation Services.
`WebPlayer_AverageCpuLoadNotExhaustedLimit`	85	Specify the CPU load, as a percentage, that the instance must get below to leave the Exhausted state. Note that this setting is applicable to both Web Player services and Automation Services.
`WebPlayer_AverageCpuLoadStrainedLimit`	50	If a service instance is in the Strained state, new users are routed to other instances that are not strained or exhausted. If all instances are strained, new users are routed to the strained instance. Specify the CPU load limit, as a percentage, that sets the state of the instance to Strained. Set to `-1` to disable the strained limit. Note that this setting is applicable to both Web Player services and Automation Services.

Setting	Default value	Description
WebPlayer_AverageCpuLoadNotStrainedLimit	45	Specify the CPU load, as a percentage, that the instance must get below to leave the Strained state. Note that this setting is applicable to both Web Player services and Automation Services.
WebPlayer_AverageCpuLoadCountOnlyCurrentProcess	False	Set to "true" to measure only the CPU load created by the instance to which a user is routed. If set to "false", the CPU load is measured for all instances on the node. Note that this setting is applicable to both Web Player services and Automation Services.
Health_UsePagingForHealth	True	Set to "true" to enable the use of information about data paging in determining the health status of the system.
Health_UseTempDiskForHealth	True	Set to "true" to enable the use of information about the temp disk in determining the health status of the system.
Health_UseWcfStatsForHealth	True	Set to "true" to enable the use of Windows Communication Foundation (WCF) statistics in determining the health status of the system.
Health_ExhaustedTempDiskMb	1500	If the amount of available disk space for the temp folder goes below this limit, the state of the instance is set to Exhausted.
Health_LogWarnTempDiskMb	5000	If the amount of available disk space for the temp folder is below this limit, WARN messages will be logged.
Health_StrainedTempDiskMb	16000	If the amount of available disk space for the temp folder goes below this limit, the state of the instance is set to Strained.
Health_LogDebugTempDiskMb	32000	If the amount of available disk space for the temp folder is below this limit, Debug messages will be logged.
Health_WcfCallsPerCoreExhausted	16	If the number of active WCF calls exceeds this limit, the state of the instance is set to Exhausted.
Health_WcfCallsPerCoreStrained	3	If the number of active WCF calls exceeds this limit, the state of the instance is set to Strained.
Health_RecentPagingMinutes	60	If the amount of recently paged out data (the latest value for Health_RecentPagingMinutes) in relation to the process heap is higher than Health_StrainedRecentPagingOutShareOfHeapPercentage (or Health_StrainedRecentPagingInShareOfHeapPercentage for paged in data), the state of the instance is set to Strained. Currently, paging will not result in an Exhausted service.
Health_StrainedRecentPagingOutShareOfHeapPercentage	50	
Health_StrainedRecentPagingInShareOfHeapPercentage	20	
<Spotfire.Dxp.Internal.Properties.Settings>		These settings should not be edited, unless instructed by Spotfire Support.

Setting	Default value	Description
`TempFolder`	Temp	For more information, see Changing the default location of the Web Player temporary files.
`SettingsFolderName`	Spotfire Worker	/the name of the folder containing the setting for the client. These settings should not be edited unless instructed by Spotfire Support.
`SettingsRootFolder`		The path to the folder containing the settings for the client These settings should not be edited unless instructed by Spotfire Support.
`ManifestDownloadTimeoutMilliseconds`	60000	Determines the timeout, in milliseconds, for downloading the manifest.
`ClientLoadBalancing_Enabled`	True	If enabled, and if multiple Spotfire servers are used, clients will load balance requests to all Spotfire servers. These settings should not be edited unless instructed by Spotfire Support.
`AllowedTlsVersions`	Tls, Tls11, Tls12	Determines which versions of the TLS security protocol are allowed. Specify the values separated by a comma ",". For information about the possible values for this setting, refer to the .NET enum `SecurityProtocolType`. If you leave the value for this setting blank, the allowed TLS versions will be `'SystemDefault'`. If you remove the setting from the configuration file, the allowed TLS versions will be the default value; `'Tls, Tls11, Tls12'`.  This setting can also be changed for the Spotfire Analyst installed client, by modifying the configuration file `Spotfire.Dxp.Main.dll.config`.
`<Spotfire.Dxp.Application.Properties.Settings>`		
`Bookmarks_MinimumSynchronizationIntervalSeconds`	60	Specifies the minimum synchronization interval for bookmarks, in seconds.
`WebServerPortAllocationCount`	-1	Determines how many ports the internal web server will bind to. All ports are bound on the loopback interface, localhost. The value for this setting should not be less than the value for `ExportRendererCount`. If a negative value is specified, this setting defaults to the number of processors on the computer.
`WebServerPortFrom`	-1	Determines the first (lowest) port that the internal web server will attempt to bind to. If a negative value is specified, this setting defaults to 8000.
`WebServerPortTo`	-1	Determines the last (highest) port that the internal web server will attempt to bind to. If a negative value is specified, this setting defaults to 65535.

Setting	Default value	Description
ExportRendererCount	-1	Determines how many renderer processes are used to concurrently render pages for PDF export and other tasks. If a negative value is specified, this setting defaults to the number of processors on the computer.
ExportRenderingTimeout	-1	Determines the timeout, in seconds, of an export to PDF operation.
<Spotfire.Dxp.Data.Properties.Settings>		
DataBlockStorage_MemoryLoadExhaustedLimit	98	If a service instance is exhausted, no new users are routed to that instance. Specify the memory load limit, as a percentage, that sets the state of the instance to Exhausted. Set to -1 to disable the exhausted limit.
DataBlockStorage_ MemoryLoadNotExhaustedLimit	93	Specify the memory load, as a percentage, that the instance must get below to leave the Exhausted state.
DataBlockStorage_MemoryLoadStrainedLimit	75	If a service instance is strained, new users are routed to other instances that are not strained or exhausted. If all instances are strained, new users are routed to the strained instance. Specify the memory load limit, as a percentage, that sets the state of the instance to Strained. Set to -1 to disable the strained limit.
DataBlockStorage_MemoryLoadNotStrainedLimit	70	Specify the memory load, as a percentage, that the instance must get below to leave the Strained state.
DataBlockStorageStorageIOSizeKB	64	This setting should not be edited unless instructed by Spotfire Support.
DataOnDemand_MaxCacheTime	01:00:00	Specify the length of time, in the format HH:MM:SS, for data on demand to be cached. This setting is used only if you configured data on demand to be cached on the web clients.
AllowedFilePaths		Provide the full path to directories or files on a local disk that you want to access in the web clients. Specify each file or directory in a separate <string> tag.

Setting	Default value	Description
`LoadOutOfProcessExceptedFilePaths`		By default, Access, Excel, and SAS files are loaded out-of-process, which means that a separate subprocess is launched to handle loading of the file. With this setting, you can specify excepted files or directories that, when accessed in the web clients, should be loaded in-process instead. Provide the full path to directories or files that should be loaded in-process, instead of out-of-process, when accessed in the web clients. Specify each file or directory in a separate `<string>` tag. To be accessed in the web clients, the files or directories must also be included in the `AllowedFilePaths` setting.  Make sure that the specified files are compatible to be loaded in-process on the services from which they will be accessed.
`<Spotfire.Dxp.Data.Access.Properties.Settings>`		
`AllowCustomQueries`	True	Enables custom queries for users on this service.
`<Spotfire.Dxp.Data.Access.Adapters.Settings>`		
`WebAuthenticationMode`	Prompt	Specify the authentication method to use for connectors. Valid options are: • `WebConfig` – Select this to require all users to connect with the credentials specified in the `Spotfire.Dxp.Web.Properties.Settings/DataAdapterCredentials` section. • `Kerberos` – Select this if your system is configured to authenticate users with Kerberos. • `Prompt` – Select this to prompt the users for a username and password for the external data source. • `ServiceAccount` – Select this to make all users connect to the external data source using the computer account or dedicated user account that is used to run the node manager.
`<Spotfire.Dxp.Data.Adapters.MicrosoftCommon.Properties.Settings>`		Settings for the Microsoft SharePoint Online connector. The settings determine what registered app in Microsoft Azure that Spotfire uses to access SharePoint Online data. Use these settings to be able to access data with the Microsoft SharePoint Online connector in the web client.
`ClientId`		The `consumer (client) ID` of your registered app in Microsoft Azure.
`ClientSecret`		The `client secret` from your registered app in Microsoft Azure.

Setting	Default value	Description
`<Spotfire.Dxp.Data.Adapters.SapBw.Properties.Settings>`		The settings in this group are specific to the connector for SAP BW.
`ConstrainedKerberos`	False	Determines whether you can use Kerberos authentication with constrained delegation when you access data from connections to SAP BW.
`EnableInProcessClient`	False	Determines if data connections to SAP BW are run in-process.  When SAP BW connections run in-process, issues in the driver can cause the web player service to stop unexpectedly.
`<system.net>`		
`<defaultProxy>`		If you need to use proxy handling for communication from the Web Player service or Automation Services to Spotfire Server, you must add the following proxy setting inside the `defaultProxy` tag: `<proxy proxyaddress="http://MyProxyServer:3128" scriptLocation="MyScriptLocation" />` The proxy setting is a part of the standard .NET Framework. You can find more information about this configuration at the Microsoft Developer Network (MSDN).  If the proxy server uses username and password authentication, you must also specify the username and password for the proxy server in the `<Spotfire.Dxp.Web.Properties.Settings>` setting.
`<appSettings>`		Configuration of third-party components. These settings should not be edited unless instructed by Spotfire Support.
`<runtime>`		Microsoft .NET configuration settings. See https://docs.microsoft.com/en-us/dotnet/framework/configure-apps/file-schema/index for more information. These settings should not be edited unless instructed by Spotfire Server.
`<startup>`		
`<system.serviceModel>`		

Spotfire.Dxp.Worker.Web.config file

This configuration file specifies Web Player service configurations, some Automation Services configurations, and user interface elements applicable to both the web clients and the library browser on Spotfire Server.

The settings in the sections `<application>`, `<userInterface><pages>`, `<userInterface><closedAnalysis>`, and `<userInterface><errorPage>`, and the setting `maxReceivedMessageSizeMb`, which sets the maximum size for file upload, are applicable both to the web client and the library browser on Spotfire Server. If these settings are changed, you must run the set-service-config command to apply the settings in the web client, and the set-server-service-config command to apply the settings in the library browser on Spotfire Server.

For information on working with this file, see Manually editing the service configuration files.

Setting	Default value	Description
`<spotfire.dxp.web>`		
`<setup>`		
`<javaScriptApi>`		
`enabled`	True	Controls whether the use of the JavaScript API is enabled or disabled.
`domain`		Restricts from which domains it is possible to use the JavaScript API. By default, all domains are allowed. A non-empty domain whitelist indicates that only the listed domains can embed Spotfire files in their web site using the JavaScript API. The list is a comma-separated list of domain names.
`</javaScriptApi>`		
`<errorReporting>`		This section is applicable for both Web Player services and Automation Services.
`emailAddress`	`" "`	Specify the e-mail address for the Spotfire administrator. When a user encounters certain server related errors, a **Report error to your administrator** mailto link is displayed. If the user clicks the link, an e-mail addressed to the administrator, including the error log, is created in the default e-mail application.  To apply this setting, you must enable it on the Spotfire Server by running the set-server-service-config command.

Setting	Default value	Description
`maxMailLength`	1000	Specify the maximum number of characters in the e-mail that is generated when a user clicks the **Report error to your administrator** link.  To apply this setting, you must enable it on the Spotfire Server by running the set-server-service-config command.
`includeDetailedErrorInformation`	False	Set to `true` to enable detailed error information, like call stacks in messages to end users. For security reasons this should not be enabled by default.
`enabledMiniDumpCreationOnError`	True	Create a mini dump file if the service goes down unintentionally.
`miniDumpPath`	" "	Specify the location where the mini dump file should be saved on the computer with the node manager installed. Leave this empty to save the mini dump file to the folder that contains the node manager log files.
`miniDumpSizeLarge`	False	Set to `true` to create a full dump. Note that this can create a very large dump file. This setting should not be edited unless instructed by Spotfire Support.
`miniDumpSizeBoth`	False	Set to `true` to capture both a small and a large dump. `miniDumpSizeBoth` will override `miniDumpSizeLarge` to capture both sizes when enabled. This setting should not be edited unless instructed by Spotfire Support.
`dumpToolPath`	C:\Program Files (x86)\Windows Kits\10\Debuggers\x64\cdb.exe	A tool, such as cdb.exe, can be used to automatically capture dumps for hanging service instance processes. To use the cdb.exe tool to capture dumps, it must be installed. Search for "Windows Software Development Kit (SDK) for Windows" and install it. Make sure to include Debugging Tools for Windows when installing. Then verify that `cdb.exe` is located in this path.
`dumpToolFlagsSmall`	-c ".dump /mhtpFidcu {0};q" -p {1}	These flags will be used if `miniDumpSizeLarge` is set to `False`. For information on the flags, refer to the cdb.exe documentation.

Setting	Default value	Description
dumpToolFlagsLarge	-c ".dump /ma {0};q" -p {1}	These flags will be used if `miniDumpSizeLarge` is set to `True`. For information on the flags, refer to the cdb.exe documentation.
dumpToolFlagsBoth	-c ".dump /mhtpFidcu {0};.dump /mA {1};qd" -p {2}	These flags will be used if `miniDumpSizeBoth` is set to `True`. For information on the flags, refer to the cdb.exe documentation.
`</errorReporting>`		
`<languages>`		This section is applicable for both Web Player services and Automation Services.
`<installedLanguages />`		This section should not be edited. The list of installed languages will be populated automatically.
`<languageMappings>`		You can define a mapping from a language preference configured by users in the browser to one of the languages installed on the service. For example, if your users have French (Canada) [fr-CA] as the highest preference language in their web browser, but the service uses French (France) [fr-FR], you can specify that [fr-FR] should be used even if the end users have not added [fr-FR] to their list of supported languages in the browser.
add browserLanguage		For each mapping from a browser language that is not directly supported, add a setting in the `<languageMappings>` section in the format: `<add browserLanguage="en-GB" installedLanguageToUse="en-US"/>`
`</languageMappings>`		
`</languages>`		

Setting	Default value	Description
`<sbdfCache>`		In order to quickly create and share map chart visualizations that use geocoding tables, and to quickly open SBDF files from the library, it is possible to cache and preload the SBDF files stored in the library. The cache is an in-memory cache that keeps recently opened SBDF files from the library open. If files have not been accessed for a specified time, or if memory is low, they will be removed from memory. This section is applicable for both Web Player services and Automation Services.
`enabled`	True	Set to `true` to enable the cache.
`cacheTimeoutMinutes`	30	Specify the minimum time an SBDF file is stored in the cache. If the preload service is used, this should be a bit longer than the `libraryCheckInterval` setting.
`<preloadSettings>`		
`enabled`	False	Set to `true` to enable the preload service of SBDF files.  The cache must also be enabled for the preload service to work.
`libraryCheckIntervalMinutes`	10	Specify how often the preloading service will check the library for new content.
`librarySearch`	MapChart.IsGeocodingTable::true AND MapChart.IsGeocodingEnabled::true	The search string that specifies which SBDF files to cache. The default search string specifies all geocoding tables in the library; you may want to restrict this to reduce memory consumption.
`</preloadSettings>`		
`</sbdfCache>`		
`<scheduledUpdates>`		
`concurrentUpdates`	2	The maximum number of updates that can be executed at the same time per Web Player service. This is used to limit resources used by the update mechanism. Min value is 1 and max value is 256.

Setting	Default value	Description
`concurrentUpdatesPerCore`	0.0 (setting is not used by default)	The maximum number of updates that can be executed at the same time relative to the number of cores in the computer on which the Web Player service is installed. Setting this value to a positive number overrides the `concurrentUpdates` setting, except if the resulting value (setting × cores) is not within the allowed range of 1.0 - 256.0. Examples: • `0.25` sets `concurrentUpdates` to 1/4 of the number of cores in the computer. • `1.0` sets `concurrentUpdates` to a value that equals the number of cores in the computer.
`updateIntervalSeconds`	60	Specify how often the service should check if any updates should be run. This is set in seconds. Min value is 30, and max value is 3600 (one hour).
`useKerberos`	False	Set to `true` to run scheduled updates as a specific Windows account when delegated Kerberos is enabled in the environment. If set to `false`, scheduled updates will run using the node manager service account. To specify the Windows account, add the following section: `<kerberosIdentity userName="domain\username" password="password" />` and specify the account username and password.
`customAccount`	" "	To use an existing account instead of scheduledupdates@SPOTFIRESYSTEM, enter the username of the account, including the domain, here (`useKerberos` must be false).
`alwaysPreserveState`	False	Set to `true` to preserve the current state of an analysis, such as marking, filtering, active page, etc., when an analysis is updated. Setting this to `true` will override the settings "Remember personalized view for each web client user" in Document Properties (defined in Spotfire Analyst) and the configuration defined under `<performance>` – `<analysis>` – `documentStateEnabled`.
`<forcedUpdate>`		

Setting	Default value	Description
`enabled`	True	It is possible to force updates on users even though the analysis is set to notify the users. This is useful if someone has left an analysis open for a long time and you want to avoid numerous versions of the analysis to be kept simultaneously. To enable forced updates, set this key to `true`.
`maximumRejectedUpdates`	2	Specify the number of times a user can be notified of new updates without accepting them, before the update is forced on the user.
`</forcedUpdate>`		
`<cacheSettings>`		
`enabled`	False	If the Web Player service is restarted, analyses that are scheduled to be pre-loaded must be reloaded. If the data used in the analyses takes a long time to load, so will the analyses. Therefore, it is possible to cache data from scheduled analyses on disk so that the analyses reload faster on restart. Set this to `true` to enable caching of data on disk.
`path`	`" "`	Specify the path on disk where data is to be stored.
`maxDiskSizeMb`	0	Specify the maximum disk space used for the cached data. Set this to `"0"` (zero) to cache data without an upper limit.
`maxAgeMinutes`	1440	Specify how long a cache entry should be kept on disk if it has not been reloaded by scheduled updates.
`</cacheSettings>`		
`</scheduledUpdates>`		
`<application>`		

Setting	Default value	Description
`helpUrl`		You can change the default help link for web client users to point to a locally stored Spotfire help. Specify the location of the locally stored help here. To use this specified help link, you must also set the `useDefaultHelpUrl` setting to `False`.  The URL must reference a Spotfire help. It is not supported to use a URL that is a reference to a website or a non-Spotfire help.
`useDefaultHelpUrl`	True	Set this to `false` and specify a locally stored help in the `helpUrl` setting to change the target of the help link in the web client. To switch back to the default online web client help, set this to `true` again.
`</application>`		
`</setup>`		
`<userInterface>`		
`<pages>`		
`showLogout`	True	Specify if the **Log out** menu item is displayed. If `true`, the menu item is displayed in the top right menu of the web client.
`showAbout`	True	Specify if the **About** menu item is displayed. If `true`, the menu item is displayed in the top right menu of the web client.
`showHelp`	True	Specify if the **Help** menu item is displayed. If `true`, the menu item is displayed in the top right menu of the web client.
`showUserName`	True	Specify if the user name should appear in the web client user interface, for example in the Modified By section in the library browser and the Analysis Information dialog.
`</pages>`		
`<diagnostics>`		This section is applicable for both Web Player services and Automation Services.
`errorLogMaxLines`	2000	Specify the maximum number of lines from the error log files to display in Monitoring and diagnostics. The range is 1000 - 50000.

Setting	Default value	Description
showRunGC	True	Specify if a garbage collection link should be visible in Monitoring and diagnostics, for web service instances.
</diagnostics>		
<analysis>		
showToolTip	True	Specify if highlighting tooltips should be shown in visualizations in the web client. Setting this value to false will increase performance.
showClose	True	Specify if the **Close** menu item is displayed. If true, the menu item is displayed in the top right menu of the web client.
showToolBar	True	Specify if the toolbar containing the menu and other controls is displayed in the web client.
showAnalysisInformationTool	True	Specify if the **Analysis Information** menu item is displayed. If true, the menu item is displayed in the top right menu of the web client.
showExportFile	True	Specify if the **Download as DXP file** menu item is displayed. If true, the menu item is displayed in the top right menu of the web client.
showExportVisualization	True	Specify if the **Export Visualization Image** menu item is displayed. If true, the menu item is displayed in the top right menu of the web client.
showUndoRedo	True	Specify if the **Undo** and **Redo** menu items are displayed and if undo is available in the visualization. If true, the menu item is displayed in the top right menu of the web client.
showDodPanel	""	Specify the behavior of the Details-on-Demand (DoD) panel. If empty (""), the DoD panel is displayed if the author of the analysis file chooses to display the DoD panel. If true, the DoD panel is always displayed. If false, the DoD panel is never displayed.

Setting	Default value	Description
showFilterPanel	""	Specify the behavior of the Filters panel. If empty (""), the Filters panel is displayed if the author of the analysis file chooses to display the Filters panel. If true, the Filters panel is always displayed. If false, the Filters panel is never displayed.
showPageNavigation	True	Specify if the Page tabs (or page links) in analyses are displayed. If you set this to false only the currently active Page as saved in the analysis will be displayed.
showStatusBar	True	Specify if the status bar is displayed.
showPrint	True	Specify if the **Print** menu item is displayed. If true, the menu item is displayed in the top right menu of the web client.
allowRelativeLinks	False	Specify if incomplete links in the Spotfire Web Player should be treated as relative to the library root directory. If false, incomplete links will be prepended with http://.
showShareWithTwitter	True	Specify if users should be able to share analyses on Twitter.
showCollaboration	True	Specify if the collaboration tool is displayed.
</analysis>	True	
<customHeader>		
source	Default	Specify in which of the server and web player UIs any added custom header is to be used. The available values of the setting are default, local, and none. For more information, see Adding Custom Headers to Spotfire UIs in the TIBCO Spotfire® Cobranding manual.
</customHeader>		
<closedAnalysis>		
showOpenLibrary	True	Specify if the Open Library link is displayed on the Closed Analysis page.

Setting	Default value	Description
showReopenAnalysis	True	Specify if the Reopen Analysis link is displayed on the Closed Analysis page.
redirectToLibrary	True	Specify if the Closed Analysis page is displayed after an analysis is closed.
</closedAnalysis>		
<errorPage>		
showOpenLibrary	True	Specify if the Open Library link is displayed on an error page.
showReopenAnalysis	True	Specify if the Reopen Analysis link is displayed on an error page.
</errorPage>		
</userInterface>		
<performance>		
<gcConfiguration>		This section is applicable for both Web Player services and Automation Services.
sustainedLowLatencyMode	True	Enabling sustainedLowLatencyMode should lead to fewer pauses during blocking GC, it may also lead to higher memory usage since GC now becomes less aggressive. When this setting is disabled, the Interactive latency mode is used.
</gcConfiguration>		
<recoverMemory>		This section is applicable for both Web Player services and Automation Services.
enabled	True	Enabling recoverMemory will help the system in the case where memory is exhausted and the last user session is removed. This state may occur if GC was not triggered by the system when freeing up large resources. The action can be specified with an integer depending on the service's memory status: 0. Do nothing. 1. Run garbage collection GC2. 2. Recycle the process.
actionWhenOk	0	Specify action when memory is OK.
actionWhenStrained	1	Specify action when memory is strained.

Setting	Default value	Description
`actionWhenExhausted`	2	Specify action when memory is exhausted.
`recycleIfScheduledAndCacheEnabled`	False	Set to `True` to allow actions (garbage collection or process recycling) to be triggered even if analyses are cached by scheduled updates, but only if scheduled updates caching is enabled.
`recycleEvenIfScheduledAnalyses`	False	Set to `True` to allow actions (garbage collection or process recycling) to be triggered even if analyses are cached by scheduled updates, even if scheduled updates caching is not enabled.
`triggerEvenIfUsersLoggedIn`	True	Actions (garbage collection or process recycling) may be triggered even if users are logged in.
`allowGcEvenIfAnalysesLoaded`	False	Set to `True` to allow GC even if analyses are open.
`minMinutesBetweenGc`	60	Specify the minimum number of minutes between garbage collections.
`minMinutesBeforeRecycle`	300	Specify the minimum number of minutes before the process is recycled.
`recycleWhenOutOfDiskEnabled`	True	Specifies whether the instances of a Web Player service are recycled when the service's temporary disk space falls below a specified amount, for a specified time period. When set to "true", and `recycleWhenOutOfDiskAfter` is a non-zero period of time, the Web Players are recycled if available disk space remains "very low" for that time span. For more information, see Changing the settings that determine when Web Player services are recycled due to low temporary disk space.
`recycleWhenOutOfDiskAfter`	01:00:00	If the available temporary disk space is very low for this period of time, the process is recycled. The value of "very low" disk space is set in the `Health_ExhaustedTempDiskMb` property in the `Spotfire.Dxp.Worker.Host.exe.config` file.
`</recoverMemory>`		
`<documentCache>`		

Setting	Default value	Description
purgeInterval	300	Specify the number of seconds between searches to identify unused, open documents (templates) to be purged. The range is 60 to 3600.
itemExpirationTimeout	00:00:00	Specify the length of time, in the format HH:MM:SS, that a document can remain in the cache when no open analysis is using that document template. Maximum value is 47.00:00:00.
</documentCache>		
<analysis>		
antiAliasEnabled	True	Specify if anti-aliasing is enabled. It is recommended that you leave anti-aliasing enabled in order to produce visualizations that are clear and sharp. All graphics in the web client are rendered with anti-aliasing enabled. However, anti-aliasing does impose a slight performance impact. The performance impact may become noticeable for visualizations that consist of a very large amount of graphical objects.
useClearType	True	Specify if ClearType is enabled. It is recommended that you leave ClearType enabled in order to produce clear and sharp text in visualizations. All graphics in the Spotfire Web Player are rendered with ClearType enabled. However, ClearType imposes a slight performance impact. The performance impact may become noticeable for certain visualizations.
documentStateEnabled	True	Specifies that the state of files is maintained between sessions. If this value is set to "true", when users resume working on a file, the file will be in the state in which that user left the file.
closedTimeout	120	Specify how long, in seconds, an analysis session will stay alive when a ping fails. The range is 60 to 4000000 (~46 days).
checkClosedInterval	60	Specify how often, in seconds, a check should be made to determine if an analysis has been closed in the web client. The range is 60 to 300.

Setting	Default value	Description
`inactivityTimeout`	02:00:00	Specify the length of time, in the format HH:MM:SS, that an analysis session can be alive when no user activity has been detected, excluding pings. The range is 00:01:00 to Infinite.
`checkInactivityInterval`	300	Specify how often, in seconds, a check should be made to determine if an analysis session has had no user activity, excluding pings. The range is 60 to 12*3600.
`regularPollChangesInterval`	500	Specify the base interval, in microseconds, from when a change is made on the web client to when the client polls for a status update. The range is 200 to 1000.
`maxPollChangesInterval`	3000	Specify the maximum value, in microseconds, by which the poll interval in `regularPollChangesInterval` is increased for each try until this value is reached. The range is 1000 to 10000.
`pollLoadInterval`	1000	Specify the interval, in microseconds, between polls when an analysis file is loading. The range is 1000 to 10000.
`needsRefreshInterval`	15	Specify the frequency, in seconds, with which the web client should ping or poll to keep the analysis alive. The range is 10 to 60.
`privateThreadPoolEnabled`	True	This setting should not be edited unless instructed by TIBCO Spotfire Support.
`privateThreadPoolWorkerCount`	1	This setting should not be edited unless instructed by TIBCO Spotfire Support.
`toolTipDelay`	1000	Specify the length of time, in microseconds, that the client must wait before requesting a visualization highlighting tooltip from the server. The range is 200 to 3000.
`undoRedoEnabled`	True	Specify if the Undo and Redo functionality is enabled.
`maxRenderTimeMs`	60000	Specify the time limit, in milliseconds, for each request or render job is allowed to create an image on the web client for a visualization. You can use this setting to prevent long-running requests or jobs from making the web client unresponsive.

Setting	Default value	Description
maxAnalysisShutdownInformations	1024	When an analysis is closed, the reasons why it was closed are stored and used when the analysis is re-opened. This value specifies the maximum number of entries stored.  This setting should not be changed.
</analysis>		
<application>		This section is applicable for both Web Player services and Automation Services.
checkUserSessionTimeoutIntervalSeconds	120	How often to check whether a user has timed out on the service.
userSessionTimeout	00:20:00	How long a user is cached on the service.
maxConcurrentWebServiceCallsPerCall	16	Specify how many active web service calls are allowed per CPU core on the service instance.
maxReceivedMessageSizeMb	64	Specify the maximum size of files uploaded to the service (Mb).
maxReaderQuotasSizeKb	256	Specify the maximum size of request and response messages sent to and from the service.
requestTimeoutSeconds	3600	Specify the timeout, in seconds, for requests between the Spotfire Server and the service. This might need to be increased if large files or data sets are uploaded to the service.
</application>		
<performanceCounterLogging>		This section is applicable for both Web Player services and Automation Services.
enabled	True	Enable or disable the logging of the specified performance counters. The result of this logging can be found in the `PerformanceCounterLog.txt` file specified in the `log4net.config` file.
cpuAverageTimeSpan	120	Specify the number of seconds to use for a rolling average when calculating the CPU load. The calculated CPU load is used to determine if the service instance is exhausted, strained, or ok.
logInterval	120	Specify the number of seconds between each performance counter logging at `INFO` level.

Setting	Default value	Description
logWcfCounters	True	When enabled, this setting logs the performance counters related to Windows Communication Foundation.
dontLogRepeatedValues	True	Set to true to not log repeated performance counter values. An exception is that INFO level counters with a non-zero value will be logged every time. Enabling this setting will make the log files much smaller.
counters		Add performance counters you wish to log, at both INFO and DEBUG level, separated by a comma ",". Each counter consists of three parts: category, counter, and instance, separated by a semi-colon ";". Both standard Windows performance counters, as well as a set of internal TIBCO counters, may be included.
debugLogInterval	15	Specify the number of seconds between each performance counter logging at DEBUG level.
debugCounters		Add additional performance counters you wish to log at DEBUG level, separated by a comma ",".
</performanceCounterLogging>		
<statistics>		This section is applicable for both Web Player services and Automation Services.
flushInterval	60	Specify the number of seconds between each logging.
enabled	True	When true, enables logging of all the other statistics for the service. The result of this logging can be found in the other log files specified in the log4net.config file.
</statistics>		
<hierarchicalClustering>		This section is applicable for both Web Player services and Automation Services.
maxInteractiveElements	2000	Specify the maximum number of rows or columns of a hierarchical clustering that can be started interactively in the web client.
maxElements	30000	Specify the maximum number of rows or columns of a hierarchical clustering that can run on the web client. Scheduled updates can run hierarchical clustering up to this size.

Setting	Default value	Description
`maxInteractiveJobs`	2	Specify the maximum number of interactive clustering jobs running in parallel.
`cpuFactorInteractiveJobs`	0.8	Specify an estimate of the number of threads that clustering will use for interactive jobs on a multi-core server running the Web Player service.
`cpuFactorLargeJobs`	0.5	Specify an estimate of the number of threads that clustering will use for scheduled update jobs on a multi-core server running the Web Player service.
`nativeMemory`	500	Specifies a memory limit, in MBytes, for the clustering algorithm. The default value `500` (MBytes) matches `maxElements = 30000`.
`</hierarchicalClustering>`		
`</performance>`		
`</spotfire.dxp.web>`		

Customizing the service logging configuration

`Log4Net.config` specifies the logs and logging levels for the Web Player service and Automation Services. To edit this configuration, you must export its contents to an `XML` file, edit it, import it, and then apply the configuration.

This task walks you through editing the configuration for the Web Player service. You can also edit the configuration for Automation Services, which includes an additional configuration file.

- For an example of editing Automation Services, see Manually editing the service configuration files.
- For a list of the log files and their properties you can customize, see Service logs.

For the TERR service and Spotfire Service for Python, JMX is the only way to capture logs. For more information, see the following.

- "Monitoring the TERR service using JMX" in TIBCO® Enterprise Runtime for R - Server Edition.
- "Monitoring Spotfire Service for Python" in TIBCO® Spotfire Service for Python.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the path of the `config.bat` file (`config.sh` on Linux).
   The default file path is `<installation dir>/tomcat/spotfire-bin`.

2. Export the configuration using export-service-conf and passing commands for the service to customize.
   For example:

```
config export-service-config --tool-password=mypassword
 --capability=WEB_PLAYER --deployment-area=Production c:\temp\config
```

   - Provide the appropriate password for the configuration tool.
   - The deployment area is usually Production. Check the administration interface page Nodes & Services if you are not sure.
   - If the directory where you want to write the configuration files already exists, you can overwrite the contents by using the `--force` flag.

   The configuration is exported to the specified directory, creating a `root` subdirectory that contains the following configuration files.

   - `log4net.config`
   - `Spotfire.Dxp.Worker.Core.config`
   - `Spotfire.Dxp.Worker.Host.exe.config`
   - `Spotfire.Dxp.Worker.Web.config`

3. Browse to the directory, and then, using a text editor, open and edit the configuration file `log4net.config`.

   In the configuration file, each potential log file is specified by an `<appender>` section. Edit each section for the logs to create. For more information about the logs this file can create, see Web Player service logs.

   a) Set the logging level.

   b) Specify the file path to write the log.

   c) Save and close the configuration file.

4. Optional: Customize the user and session statistics, and the performance counter logging, specified in the file `Spotfire.Dxp.Worker.Web.config`, which is also exported and written to the `root` subdirectory.

   You can customize the performance counters at both the `INFO` and the `DEBUG` levels. See Service log levels for more information.

5. Return to the command line and import the custom configuration using import-service-config, passing in the configuration name, the tools password, and the path for the configuration.
   For example:

```
config import-service-config --tool-password=mypassword
--config-name=SampleConfig c:\temp\config
```

   The configuration is successfully imported.

6. Set the custom configuration using set-service-config, passing in the service ID and the configuration name.
   For example:

```
config set-service-config --tool-password=mypassword
```

```
--service-id="VALUE" --config-name=SampleConfig
```

Use the list-services command to get the service ID. In some cases, you must enclose the service ID in double quotation marks.

A warning is displayed indicating that setting a new service configuration causes all running instances of the service to restart, and you must indicate whether you want to continue. If you press `Y`, the service restarts and the new configuration is set.

Result

The configuration setting for the Web Player service is displayed in Nodes & Services, and the log files should be written as specified.

Customize statistics and performance counter logging

You can configure the collection of user and session statistics and the performance counters in the file `Spotfire.Dxp.Worker.Web.config`.

The `Spotfire.Dxp.Worker.Web.config` file is exported and imported with other service configuration logging files as described in the task Customizing the service logging configuration.

To customize the information to collect, in the file `Spotfire.Dxp.Worker.Web.config`, find and edit the `<performance>` and `<statistics>` sections. For detailed information about the nodes, see the reference topic for Spotfire.Dxp.Worker.Web.config.

Service log levels

For events occurring for a service, Spotfire Server can provide a log entry that specifies a level of severity. The level applied can provide you with clues about the nature of the log entry.

You can set the log level for each log file you write. The following table lists the log levels and their descriptions. If you set logging the lowest (most severe level), notice that only fatal problems are logged. For each added level of reporting, levels are concatenated, so at the highest, most thorough level, your logs contain detailed information at all levels.

For information about Web Player service logs and their properties, see Web Player service logs.

Log level	Comment
`OFF`	Specifies that no log should be created.
`FATAL`	Specifies that fatal problems should be logged.
`ERROR`	Specifies that fatal problems and errors should be logged.
`WARN`	Specifies that fatal problems, errors, and warnings should be logged.
`INFO`	Specifies that fatal problems, errors, warnings, and information should be logged.
`DEBUG`	Specifies the a fine-grained and detailed logging of events.
`TRACE`	Specifies the an even finer-grained and detailed level of detail for logging of events. Use with caution, because it can degrade server performance if it runs for long.

For a list of server and node logging levels, see Server and node logging levels.

Configuring a specific directory for library import and export

You can change the directory that Spotfire uses for library import and export if the default directory is inconvenient. For most purposes this setting does not need to be changed.

Procedure

- You can set a new library directory by using either the configuration tool or the command line:
  - In the configuration tool, the **Library Directory** panel is at the bottom of the **Configuration** tab.
  - On the command line, use the config-import-export-directory command.

Enabling cached and precomputed data for scheduled update files

Disk caching and precomputations of data shorten the time it takes for a scheduled update file to reopen in a Spotfire Web Player after the Web Player is restarted. This feature is disabled by default. It is enabled at the service level by editing the `Spotfire.Dxp.Worker.Web.config` file for each installed web client service.

You then have the option of turning the feature off for individual files (see Disallowing cached and precomputed data in individual scheduled update files).

Procedure

1. Open a command line and export the service configuration by using the export-service-config command.
2. Open the `Spotfire.Dxp.Worker.Web.config` file in a text editor or XML editor and locate the following section. By default, the exported configuration file is saved to the `<installation dir>\tomcat\spotfire-bin\config\root` directory.

```
<scheduledUpdates concurrentUpdates="2" updateIntervalSeconds="60">
        <forcedUpdate enabled="true" maximumRejectedUpdates="2"/>
        <cacheSettings enabled="false" path="" maxDiskSizeMb="0" maxAgeMinutes="1440"/>
      </scheduledUpdates>
```

3. In the line `<cacheSettings enabled="false" path="" maxDiskSizeMb="0" maxAgeMinutes="1440"/>`, make these changes:
   - Set `cacheSettings enabled` to "true".
   - Set `path` to the path on disk where the data is to be stored.

   For information on the other settings, see Spotfire.Dxp.Worker.Web.config.
4. Import the configuration back into Spotfire Server by using the import-service-config command.
5. Assign the edited service configuration to the Spotfire Server by using the set-service-config command.
   Example:

```
config set-service-config --service-id=6610a31b-1a2a-4497-b146-cee797f9b6a7
```

Use the list-services command to obtain the service ID.

Disabling the attachment manager cache

By default the Spotfire attachment manager caches library content and the results of information link executions when downloading or saving large amounts of data. You can disable the attachment manager cache by editing the `configuration.xml` file

Procedure

1. Export and open the Spotfire Server configuration file; for general instructions, see Manually editing the Spotfire Server configuration file.
2. In the `configuration.xml` file, locate the following section and set `<content-caching-enabled>` to "false":

```
<library>
    <import-export-path>default</import-export-path>
    <content-caching-enabled>true</content-caching-enabled>
    <max-number-concurrent-imports-and-exports>3</max-number-concurrent-imports-and-
exports>
  </library>
```

3. Then locate the `<information services>` section and set `<result-caching-enabled>` to "false".
4. Import the server configuration file and restart the server(s); for instructions, see Manually editing the Spotfire Server configuration file.

Tips for running antivirus software

You can reduce the performance impact of running antivirus software on computers that run Spotfire Servers and node managers by excluding certain files from the system checks.

For performance reasons, the following are best practices for using antivirus software on server and node manager computers:

- Do not run full system scans during business hours because the I/O operations affect system performance.
- Exclude certain files from system scans, as indicated in the sections below.

Computers running Spotfire Server

In the `<server installation dir>/tomcat/temp` directory, including subdirectories, exclude files with these extensions:

- `*.cab`
- `*.config`
- `*.css`
- `*.gif`
- `*.gz`
- `*.ico`
- `*.js`
- `*.png`
- `*.svg`
- `*.tmp`
- `*.xml`

In the `<server installation dir>/tomcat/logs` directory, exclude files with these extensions:

- `*.log`
- `*.txt`
- *.1, *.2, ..., *.*n*

In the `<server installation dir>/tomcat/application-data/library` directory, including subdirectories, exclude files with this extension:

- *.zip

Computers running Spotfire node manager

In the `<node manager installation dir>/nm/logs` folder, exclude files with these extensions:

- *.log
- *.txt
- *.1, *.2, ..., *.*n*

In the `<node manager installation dir>/services/<ServiceFolder>/Temp` folders, including subfolders, exclude the following:

- Files with the following extensions:
  - *.dxp
  - *.tmp
- Files without an extension

Connectors

With the connectors that are available in Spotfire, users can connect to, and analyze data from, a variety of data sources. This section describes how to configure the connectors for use in Spotfire Analyst, TIBCO Spotfire® Business Author, TIBCO Spotfire® Consumer, and TIBCO Spotfire® Automation Services.

The following connectors are currently available:

- Amazon Redshift
- Apache Drill
- Apache Spark SQL
- Attivio
- Cloudera Hive
- Cloudera Impala
- Dremio
- Google Analytics
- Google BigQuery
- Hortonworks
- IBM DB2
- IBM Netezza
- Microsoft SQL Server
- Microsoft SQL Server Analysis Services
- OData
- Oracle
- Oracle Essbase

- Oracle MySQL
- Pivotal Greenplum
- Pivotal HAWQ
- PostgreSQL
- Salesforce
- SAP BW
- SAP HANA
- Snowflake
- Teradata
- Teradata Aster
- TIBCO Cloud™ Live Apps
- TIBCO ComputeDB
- TIBCO Data Virtualization
- TIBCO Spotfire Data Streams
- Vertica

Setting up connectors

Before you can use a data source connector on a Spotfire client, the connector must be installed on the server and the data source driver must be installed on the client computer.

Prerequisites

Client packages have been deployed to Spotfire Server. The connectors are included in the distribution file named `Spotfire.Dxp.sdn`. For information on package deployment, see Deploying client packages to Spotfire Server.

After deployment, make sure to update the clients with the deployed packages. This is done by restarting any open Spotfire clients, logging in as usual, and then clicking **Update now**.

These are the additional required steps for setting up data source connectors.

Procedure

1. On the following computers, install the data source drivers that correspond to the connectors that will be used in your implementation:
   - All computers running Spotfire Analyst.
   - All computers running a node with Spotfire Web Players or Spotfire Automation Services for which connectors should be available.

   For information about the required drivers and where to find them, see the TIBCO Spotfire Connectors System Requirements page.

   If you have installed a 32-bit version of the Spotfire Analyst, then you must use the 32-bit version of the data source driver. For Spotfire Web Player services, always use the 64-bit driver.
2. If the connectors should be available for users of Spotfire Web Players or in Spotfire Automation Services, additional configuration on the server is necessary; see Configuring connectors for use with web clients and Spotfire Automation Services.
3. Set the access rights for the users; for details, see Access to the connectors.

4. Some connectors require additional configuration; see, for example, Configuring the Google Analytics connector and Installing Oracle Essbase Client on client computers.

Configuring connectors for use with web clients and Spotfire Automation Services

If connectors should be available for users of Spotfire web clients, or in Spotfire Automation Services, some configuration on the Spotfire Server is necessary.

This is a suggested workflow; detailed descriptions for each step are available in separate topics.

Procedure

1. Optional: Create a configuration that the service will use, and assign it to the deployment area that the web clients or Automation Services use. For instructions, see Preconfiguring Spotfire Web Player services (optional) on page 142 or Preconfiguring Spotfire Automation Services (optional) on page 144, depending on the type of service that you are configuring.

   If you have to configure the authentication mode for any of the deployed connectors, this step is required. See Authentication modes on page 190 for more information.

2. Install a service and make sure to select the same deployment area as in #unique_418/unique_418_Connect_42_step_a57c6949a87d42d9901fddf7e31d7485. For detailed instructions on installation of services, see Installing Spotfire Web Player instances on page 143 or Installing Spotfire Automation Services instances on page 144, depending on the type of service you are configuring.

   If you created a configuration in #unique_418/unique_418_Connect_42_step_a57c6949a87d42d9901fddf7e31d7485, select that configuration when you install the service.

3. After the service has been installed successfully, test that it is now possible to work with data from the connectors.

   Note that some connectors require additional configuration. See for example Configuring the Google Analytics and Google BigQuery connectors on page 196 and Installing Oracle Essbase Client on client computers on page 195.

   Create an analysis in Spotfire Analyst, and configure connections with the connectors that should be available in the web clients. Then save the analysis to the library. Verify that you can successfully open the analysis in a web client.
   Create a Spotfire Automation Services job with tasks that use the connectors that should be available for Spotfire Automation Services. Verify that you can run the job successfully.

Authentication modes

You may have to change the authentication mode for some connectors so that they are available for use with Spotfire web clients. This is done in the `Spotfire.Dxp.Worker.Host.exe.config` file.

To change the authentication mode for a connector on a Spotfire Web Player service, you must modify an existing configuration or create a new configuration and assign it to the deployment area on which the `Spotfire.Dxp.sdn` distribution file has been deployed. Instructions are available in Preconfiguring Spotfire Web Player services (optional) on page 142, but details specific to the connectors are listed here.

- The authentication mode settings are located in the section `<Spotfire.Dxp.Data.Access.Adapters.Settings>`. To edit the configuration file, you must first export it from Spotfire Server using the export-service-config command. For instructions, see Preconfiguring Spotfire Web Player services (optional) on page 142.

- These are the available authentication modes:
  - Prompt
  - ServiceAccount
  - Kerberos
  - WebConfig

  By default, all the connectors use the Prompt mode. To read more about the settings, see Configuration file examples.
- If you are unsure of what a certain connector is called in the configuration file, see Connector names in configuration file.

Connector configuration examples

By default, all Spotfire connectors are listed in the configuration file, `Spotfire.Dxp.Worker.Host.exe.config`, and all connectors use Prompt as authentication mode.

This is the connector section of the configuration file:

```
<Spotfire.Dxp.Data.Access.Adapters.Settings>
 <setting name="WebAuthenticationMode" serializeAs="Xml">
  <value>
   <adapters>
    <adapter name="Spotfire.SqlServerAdapter" mode="Prompt"/>
    <adapter name="Spotfire.TeradataAdapter" mode="Prompt"/>
    <adapter name="Spotfire.OracleAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SsasAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SapBwAdapter" mode="Prompt"/>
    <adapter name="Spotfire.EssbaseAdapter" mode="Prompt"/>
    <adapter name="Spotfire.CompositeAdapter" mode="Prompt"/>
    <adapter name="Spotfire.MySqlAdapter" mode="Prompt"/>
    <adapter name="Spotfire.NetezzaAdapter" mode="Prompt"/>
    <adapter name="Spotfire.PostgreSqlAdapter" mode="Prompt"/>
    <adapter name="Spotfire.VerticaAdapter" mode="Prompt"/>
    <adapter name="Spotfire.TeradataAsterAdapter" mode="Prompt"/>
    <adapter name="Spotfire.HanaAdapter" mode="Prompt"/>
    <adapter name="Spotfire.GreenplumAdapter" mode="Prompt"/>
    <adapter name="Spotfire.ImpalaAdapter" mode="Prompt"/>
    <adapter name="Spotfire.ClouderaHiveAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SparkSqlAdapter" mode="Prompt"/>
    <adapter name="Spotfire.HortonworksAdapter" mode="Prompt"/>
    <adapter name="Spotfire.DB2Adapter" mode="Prompt"/>
    <adapter name="Spotfire.PivotalHdAdapter" mode="Prompt"/>
    <adapter name="Spotfire.ODataAdapter" mode="Prompt"/>
    <adapter name="Spotfire.RedshiftAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SalesforceAdapter" mode="Prompt"/>
    <adapter name="Spotfire.GoogleAnalyticsAdapter" mode="Prompt"/>
    <adapter name="Spotfire.TeiidAdapter" mode="Prompt"/>
    <adapter name="Spotfire.DataStreamsAdapter" mode="Prompt"/>
    <adapter name="Spotfire.TIBCOComputeDBAdapter" mode="Prompt"/>
    <adapter name="Spotfire.DrillAdapter" mode="Prompt"/>
    <adapter name="Spotfire.DremioAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SnowflakeAdapter" mode="Prompt"/>
    <adapter name="Spotfire.LiveAppsAdapter" mode="Prompt"/>
    <adapter name="Spotfire.GoogleBigQueryAdapter" mode="Prompt"/>
    <adapter name="Spotfire.SharepointAdapter" mode="Prompt"/>
   </adapters>
  </value>
 </setting>
</Spotfire.Dxp.Data.Access.Adapters.Settings>
```

The effect that a certain authentication mode has for users who are logging in to a web client depends on the authentication method that was selected for the connection in the analysis. All authentication alternatives are not available for all connectors.

Prompt

Prompt is the default authentication mode. When it is used, web client users are prompted for their username and password when they log in to analyses that contain connections.

Example: `<adapter name="Spotfire.SparkSqlAdapter" mode="Prompt"/>`

ServiceAccount

ServiceAccount is used as authentication mode for connectors that are configured for anonymous authentication (for example Cloudera Hive, Cloudera Impala, Hortonworks, and OData). Web client users are connected to the external data source using the computer account or dedicated user account that is running the node manager.

Example: `<adapter name="Spotfire.ClouderaHiveAdapter" mode="ServiceAccount"/>`

Kerberos

To use Kerberos as authentication method, the following must be true:

- Spotfire Server is configured to use delegated Kerberos.
- In the analysis' connection login dialog, Kerberos is selected as authentication method.

For more information about Kerberos configuration, see Kerberos authentication on page 87.

Example: `<adapter name="Spotfire.SqlServerAdapter" mode="Kerberos"/>`

WebConfig

When WebConfig is used as authentication method, you can configure data sources in analyses so that web client users log in automatically with credentials stored in a credentials profile.

Example: `<adapter name="Spotfire.SparkSqlAdapter" mode="WebConfig"/>`

To use WebConfig authentication mode, you must add a credentials profile, which stores a username and password for authentication, in the web client configuration. This is done in the `DataAdapterCredentials` settings section in the configuration file Spotfire.Dxp.Worker.Host.exe.config file on page 160. See the next section, DataAdapterCredentials.

You must also specify the credentials profile in the connection data source in the analysis. If you do not specify a credentials profile in the analysis, then web client users must enter their credentials.

DataAdapterCredentials

If WebConfig is selected as `WebAuthenticationMode`, users log in with a credentials profile that is stored in the web client service configuration. A credentials profile consists of a profile name, a username, and a password. Optionally, you can specify a list of allowed servers and/or connectors in the `allowed-usages` element, which determines conditions for what you can use the credentials profile for. See Spotfire.Dxp.Worker.Host.exe.config file on page 160.

```
<entry profile="profile_name">
 <allowed-usages>
  <entry server-regex="database\.example\.com" />
 </allowed-usages>
 <username>user</username>
 <password>password</password>
</entry>
```

In the example below, two credentials profiles have been added:

```
<Spotfire.Dxp.Web.Properties.Settings>

 <setting name="DataAdapterCredentials" serializeAs="Xml">
  <value>
   <credentials>
    <entry profile="Sales_Dept">
     <allowed-usages>
      <entry server-regex="database\.example\.com" />
     </allowed-usages>
     <username>EMEA\SalesUsers</username>
     <password>MySalesPassword</password>
    </entry>
    <entry profile="Executive">
     <allowed-usages>
      <entry server-regex="another-database\.example\.com" />
     </allowed-usages>
     <username>EMEA\ExecUsers</username>
     <password>MyExecPassword</password>
    </entry>
   </credentials>
  </value>
 </setting>

</Spotfire.Dxp.Web.Properties.Settings>
```

For integrated security, the username should be in the DOMAIN\user format as in the example with `EMEA\SalesUsers` and `EMEA\ExecUsers`. The profile is an arbitrary string.

To use the credentials in an analysis, enter the profile name in Spotfire Analyst, on the **Credentials** page of the Data Source Settings dialog. When a credentials profile is specified both in the configuration file and in an analysis in Spotfire Analyst, web client users are not prompted for username and password to the connection when they open the analysis. Instead, the username and password that are defined in the credentials profile of the configuration file are used to log in to the data source.

Connector names in configuration file

This list describes how to refer to the different connectors in the configuration file `Spotfire.Dxp.Worker.Host.exe.config`.

Official name	Name in configuration file
Amazon Redshift	`RedshiftAdapter`
Apache Drill	`DrillAdapter`
Apache Spark SQL	`SparkSqlAdapter`
Attivio	`TeiidAdapter`
Cloudera Hive	`ClouderaHiveAdapter`
Cloudera Impala	`ImpalaAdapter`
Dremio	`DremioAdapter`
Google Analytics	`GoogleAnalyticsAdapter`
Google BigQuery	`GoogleBigQueryAdapter`
Hortonworks	`HortonworksAdapter`
IBM DB2	`DB2Adapter`
IBM Netezza	`NetezzaAdapter`
Microsoft SharePoint Online	`SharepointAdapter`
Microsoft SQL Server	`SqlServerAdapter`
Microsoft SQL Server Analysis Services	`SsasAdapter`
OData	`ODataAdapter`
Oracle	`OracleAdapter`
Oracle Essbase	`EssbaseAdapter`
Oracle MySQL	`MySqlAdapter`
Pivotal Greenplum	`GreenplumAdapter`
Pivotal HAWQ	`PivotalHdAdapter`
PostgreSQL	`PostgreSqlAdapter`
Salesforce	`SalesforceAdapter`
SAP BW	`SapBwAdapter`
SAP HANA	`HanaAdapter`
Snowflake	`SnowflakeAdapter`
Teradata	`TeradataAdapter`
Teradata Aster	`TeradataAsterAdapter`

Official name	Name in configuration file
TIBCO Cloud™ Live Apps	`LiveAppsAdapter`
TIBCO ComputeDB	`TIBCOComputeDBAdapter`
TIBCO Data Virtualization	`CompositeAdapter`
TIBCO Spotfire Data Streams	`DataStreamsAdapter`
Vertica	`VerticaAdapter`

Access to the connectors

After you configure the connectors, you must specify access rights to make the connectors available for users of any Spotfire client.

In Spotfire, the access rights to data from connectors are controlled by the following items:

- The data source authentication. See the official help for the data source of interest for more information. For a short summary of which authentication modes are available for a specific connector, you can view the help section for the connector in the Spotfire Analyst User's Guide.
- The licenses that are enabled for the end user groups. For more information, see Groups and licenses.

If the steps in Configuring connectors for use with web clients and Automation Services are performed on the Spotfire Web Player service, and an analysis using that connection is created, then users of Spotfire web clients can connect to the data source directly.

Installing Oracle Essbase Client on client computers

To use the Oracle Essbase connector, you must also install Oracle Essbase Client on each computer that will run the connector.

Prerequisites

Ensure that you have access to the appropriate Oracle Essbase Client installer and unzip any zipped files on your computer (for example, `ClientInstallers-11122.zip`).

For more information about the supported Oracle Essbase versions, see TIBCO Spotfire Connectors System Requirements.

Procedure

1. In the extracted archive, locate the `EssbaseClient` directory containing the installation program `EssbaseClient.exe`.
2. Double-click `EssbaseClient.exe`.
3. Select the appropriate language and continue.
4. In the installer pane, click **Next**.
5. Make a note of the destination directory; you need it for creating the appropriate environment variables. Click **Next**.
6. In the **Custom Setup** pane, ensure that both **Essbase Client** and **Essbase Client C API** are selected to be installed before you click **Next**.

The Essbase Client C API is not selected by default. You must select it manually.

7. Click **Install**, and then click **Finish** when the installation is completed.

In the **Installed Programs** list of the Control Panel, you can find a listing for Oracle® Hyperion Essbase Client. Use this entry if you must uninstall Oracle Essbase. Also, remember to remove the created environment variables that are listed in Creating environment variables.

Creating environment variables

You must create the required environment variables to access the Essbase Client C API.

The environment variables must be exactly as specified, and they must point to the correct paths. Make sure that no additional blank spaces are added.

Procedure

1. Open the **System Properties** of your computer. (On Windows 7 this is reached from **Control Panel** > **All Control Panel Items** > **System** > **Advanced system settings**.)
2. On the **Advanced** tab, click **Environment Variables**.
3. On client computers, under **System variables**, click **New**, and then create the variable `EPMHOME` and set its value to the home path for the Oracle Enterprise Management System (for example, `C:\oracle\Middleware\EPMSystem11R1`).

   This home path contains the directories `bin`, `bin-32`, `common`, and `products`.

   Always use System variables, if possible. For computers running Spotfire Web Player services, Spotfire Automation Services services, the TERR service, or Spotfire Service for Python, the environment variables must be defined as System variables.
4. Create the variable `ARBORPATH` and set it to the destination folder chosen in the installer (for example, `C:\oracle\Middleware\EPMSystem11R1\products\Essbase\EssbaseClient` (or `%EPMHOME%\products\Essbase\EssbaseClient`)).
5. Create the variable `ESSBASEPATH` and set it to `%ARBORPATH%`.
6. Add the following to the `PATH` variable (or create the `PATH` variable): `%ARBORPATH%\bin;%EPMHOME%\bin;`

Configuring the Google Analytics and Google BigQuery connectors

To enable the Google Analytics and Google BigQuery connectors for use in Spotfire web clients, you must configure your Google Cloud Platform project.

Procedure

- For detailed instructions, see the topic **Enabling Google connectors in Spotfire web clients** in the Spotfire Analyst User's Guide.

Enabling federated authentication for the Salesforce connector in web clients

To enable the use of the authentication method **Log in with Salesforce** , which is required to use federated authentication, when you use the Salesforce connector in Spotfire web clients, you must first create a new app in your Salesforce instance. Then, in Spotfire, you configure the Salesforce connector to connect through the new app.

Creating an app in Salesforce for connecting to web clients

You must create a new app in your Salesforce instance so that you can allow access to Salesforce from Spotfire web clients.

Prerequisites

You must have a Salesforce account with administrator rights.

Your Spotfire Server must be configured to use the HTTPS protocol.

Procedure

1. In a web browser, go to https://login.salesforce.com, and log in with a Salesforce account that has administrator rights.
2. In Salesforce Classic, go to **Setup**.
3. In the navigation panel on the left, under Build, expand **Create** and click **Apps**.
4. On the Apps page, under **Connected Apps**, click **New** to start creating a new connected app.
5. On the New Connected App page, enter the required basic information: **Connected App Name**, **API Name**, and **Contact Email**.
6. Under **API (Enable OAuth Settings)**, select the check box **Enable OAuth Settings**.
7. In the **Callback URL** text field, enter the following URLs:
   - https://<spotfire_server>/spotfire/wp/oauth2/code

     The above URL is used to allow access from web clients on your Spotfire Server. Salesforce requires that you use the HTTPS protocol for this URL, and not HTTP.
   - http://localhost:55932
   - http://localhost:55933
   - http://localhost:55934

     The above URLs are used to allow access from Spotfire installed clients. You can use any port numbers you want in the URL (if you do not want to use the suggested port numbers above) but it must be the same as the port numbers that you specify for the RedirectPort preference in Spotfire.
8. In the **Available OAuth Scopes** list, click to select the following items and then click **Add**:
   - Access and manage your data (api)
   - Provide access to your data via the Web (web)
   - Perform requests on your behalf at any time (refresh_token, offline_access)
9. Make sure that the check box **Require Secret for Web Server Flow** is selected.
10. To finish creating your new Salesforce app, click **Save**.

Configuring the Salesforce connector in Spotfire

After creating your new Salesforce app, you must configure the Salesforce connector in Spotfire to connect through the new app.

Procedure

1. Open Spotfire Analyst, and log in as a user with administrator rights.
2. Click **Tools > Administration Manager**.

3. In the Administration Manager dialog, go to the **Preferences** tab, and click a group for which you want to enable Salesforce connectivity in the web clients.
4. In the list of preferences, expand **Connectors** and click **Salesforce**.
5. Click **Edit**.
6. In the Edit Preferences dialog, in the **ConsumerKey** field, add the Consumer Key from your Salesforce app.
7. In the **ConsumerSecret** field, add the Consumer Secret from your Salesforce app.
8. In the **RedirectPorts** field, enter 55932, 55933, 55934 (separate each entry with a comma or semicolon).

The port numbers must be the same as the port numbers that you used for the installed client URLs when you created the Salesforce app.

If you opted to use port numbers that are different from the default 55932, 55933, and 55934, enter those port numbers instead.

9. To save your changes and close the Edit Preferences dialog, click **OK**.

Result

For the changes to take effect, users must log out of Spotfire and then log in again.

Configuring the Microsoft SharePoint Online connector

To be able to use the Microsoft SharePoint Online connector in your Spotfire web clients, you must add the `application (client) id` and `client secret` of a registered app in Microsoft Azure in the web player service configuration.

Procedure

1. In your registered app in Microsoft Azure, add the following address as a Web Redirect URI:

```
https://<spotfire_server>/spotfire/wp/oauth2/code
```

2. Then add the application (client) ID and client secret in the `<Spotfire.Dxp.Data.Adapters.MicrosoftCommon.Properties.Settings>` settings in the Spotfire.Dxp.Worker.Host.exe.config file.

```
<Spotfire.Dxp.Data.Adapters.MicrosoftCommon.Properties.Settings>
      <setting name="ClientId" serializeAs="String">
        <value>My application (client) ID</value>
      </setting>
      <setting name="ClientSecret" serializeAs="String">
        <value>My Client Secret</value>
      </setting>
</Spotfire.Dxp.Data.Adapters.MicrosoftCommon.Properties.Settings>
```

For detailed instructions, see the topic *Enabling the Microsoft SharePoint Online connector in Spotfire web clients* in your version of the Spotfire Analyst User's Guide.

Information Services

Information Services is a server-side method for accessing data from JDBC-compliant data sources. The connection to the external data source is done by the Spotfire Server on behalf of the Spotfire client.

The prerequisite for using Information Services to access data is that you install and configure the corresponding JDBC drivers and data source templates for your data source on your Spotfire Server.

- Included JDBC drivers and data source templates

- Installing database drivers for Information Services
- Adding a data source template

Once these preparations are set up on the Spotfire Server, you use the Information Designer tool in your Spotfire Analyst client to create information links for accessing data from your JDBC-compliant data source.

- For more information about how to create information links with the Information Designer tool in Spotfire Analyst, see the Spotfire Analyst User's Guide.

Installing database drivers for Information Designer

To be able to access data from a JDBC-compliant data source with Information Services, you must install the appropriate JDBC driver on the computer that is running Spotfire Server.

If you have a clustered server deployment, you must install the driver on all computers that run Spotfire Server in the cluster.

Procedure

1. Download the database driver.
2. Place the driver in the following directory: `<installation dir>/tomcat/custom-ext`.

   This is the recommended directory for most JDBC drivers. There might be some JDBC drivers that you must install to a different directory. For information on some known exceptions, see JDBC Data Access Connectivity on the TIBCO Community
3. Restart Spotfire Server.

What to do next

To connect to an external data source, you must also add and/or enable a data source template that matches the database and the specific database driver.

You can add and enable data source templates in two different ways:

- Add a data source template with the configuration tool
- Add a data source template with the command add-ds-template

The database connection URL, used by the server to connect to the database, may differ for different database drivers; see Database drivers and database connection URLs.

Adding a data source template with the configuration tool

To be able to use Information Services to access data from a JDBC-compliant data source, you must have the corresponding data source template. You can add a data source template to the Spotfire Server configuration with the TIBCO Spotfire Server Configuration Tool.

Prerequisites

You must have installed the corresponding database driver on the computer running TIBCO Spotfire Server.

Procedure

1. To open the TIBCO Spotfire Server Configuration Tool, follow the instructions in Opening the configuration tool.

2. In the TIBCO Spotfire Server Configuration Tool, select the **Configuration** tab.
3. On the Configuration tab, in the navigation pane to the left, select **Data Source Templates**.
4. To add a new data source template, click **New**.
5. In the Data Source Template dialog, enter the name of the data source, and enter the data source template.
6. Click **OK**.
7. To enable the template, select the **Enabled** check box to the right of the name of your data source.
8. To save your changes, click **Save Configuration**. In the dialog, click Next, add a comment, and then click **Finish**.
9. For your changes to take effect, restart the TIBCO Spotfire Server service.

Result

Your data source template is available to use in the Information Designer in Spotfire Analyst. You can use it to create information links for accessing data from the corresponding external data source type.

Data source templates

When you use Information Services to access data from a JDBC data source, you must have a data source template for that data source. When you have installed the appropriate driver and data source template on the Spotfire Server, you can use Information Designer tool in Spotfire Analyst to create information links for accessing data from that data source.

Some data source templates and JDBC drivers are included in the Spotfire Server installation. As a Spotfire administrator, you can add custom data source templates. This way, you can enable users to create information links to more types of data sources.

You can add, enable, and edit data source templates in two different ways:

- Add, enable, or edit a data source template with the configuration tool.
- Add a data source template with the command add-ds-template.

Included drivers and data source templates for Information Services

Some data source templates and JDBC drivers for Information Services are included in the TIBCO Spotfire Server installation.

You can add JDBC drivers and data source templates for other JDBC-compliant data sources that you want to access data from.

Included JDBC drivers for Information Services

Driver
PostgreSQL
SQL Server 2005 or newer
DB2 - DataDirect
MySQL - DataDirect
Oracle - DataDirect

Driver
SQL Server - DataDirect
Sybase - DataDirect
Redshift

Included data source templates for Information Services

The following data source templates, together with the required JDBC drivers, are included and enabled by default:

- DB2 (DataDirect)
- MySQL (DataDirect)
- Oracle (DataDirect)
- Redshift
- SQL Server (DataDirect)
- SQL Server (2005 or newer)
- Sybase (DataDirect)

The following data source templates, together with the required JDBC drivers, are included but disabled by default. You must enable the templates before you can use them:

- Oracle (service name, DataDirect)
- Oracle (service name, DataDirect, delegated Kerberos)
- SQL Server (DataDirect, delegated Kerberos)
- SQL Server (2005 or newer, delegated Kerberos)
- PostgreSQL

The following data source templates are included, but they are disabled by default and the required JDBC drivers are not included. You can use the templates if you install the appropriate driver and enable the template:

- TIBCO Data Virtualization
- DB2
- DB2 TYPE4
- MySQL
- MySQL5
- Netezza
- Oracle
- Oracle (delegated Kerberos)
- Oracle (service name)
- Oracle (service name, delegated Kerberos)
- SAS/SHARE
- SQL Server (JTDS)
- SQL Server 2000

- Sybase
- Sybase (JTDS)
- Teradata

Data source template commands

You can use these command-line commands to handle data source templates.

If you want to	Use this command	Notes
Add a new data source template	add-ds-template	
Enable, modify, or disable a data source template	modify-ds-template	For a data source template to become available in the Information Designer, it must be enabled.
Remove a data source template	remove-ds-template	Verify that no data sources use the data source template before you remove it. If a data source template is removed, all data sources using that template stop working.

XML settings for data source templates

The following table defines all the available XML settings for data source templates; only the first three are required. All other settings use their default values if not specified.

Setting	Description	Default value
type-name	A unique name for the configuration.	
driver	The JDBC driver Java class used for creating connections.	
connection-url-pattern	A pattern for the connection URL. The URL syntax is driver specific.	
ping-command	A dummy command to test connections.	SELECT 1
connection-properties	JDBC connection properties.	
metadata-provider	Java class that provides database metadata.	BasicJDBCMetadataProvider
sql-filter	Java class that generates SQL.	BasicSQLFilter
sql-runtime	Java class that handles SQL execution.	BasicSQLRuntime

Setting	Description	Default value
fetch-size	A fetch size specifies the amount of data fetched with each database round trip for a query. The specified value is shown as the default value in Information Designer. May be changed at instance level.	10000
batch-size	A batch size specifies the amount of data in each batch update. The specified value is shown as the default value in Information Designer. May be changed at instance level.	100
max-column-name-length	The maximum length of a database column name. This limit is used when creating temporary tables.	30
table-types	Specify which table types to retrieve.	TABLE, VIEW
supports-catalogs	Tells if the driver supports catalogs.	true
supports-schemas	Tells if the driver supports schemas.	true
supports-procedures	Tells if the driver supports stored procedures.	false
supports-distinct	Tells if the driver supports distinct option in SQL queries.	true
supports-order-by	Tells if the driver supports order-by option in SQL queries.	true
column-name-pattern	Determines how a column name is written in the SQL query.	"$$name$$"
table-name-pattern	Determines how a table name is written in the SQL query.	"$$name$$"
schema-name-pattern	Determines how a schema name is written in the SQL query	"$$name$$"
catalog-name-pattern	Determines how a catalog name is written in the SQL query.	"$$name$$"
procedure-name-pattern	Determines how a procedure name is written in the SQL query.	"$$name$$"
column-alias-pattern	Determines how a column alias is written in the SQL query.	"$$name$$"
string-literal-quote	The character used as quote for string literals.	SQL-92 standard

Setting	Description	Default value
max-in-clause-size	The maximum size of an SQL IN-clause. Larger lists are split into several clauses that are OR:ed together.	1000
condition-list-threshold	A temporary table is used when executing an SQL query, where total size of a condition list is larger than this threshold value. A Data Base Administrator may prefer a lower value than the default. Depends on the maximum SQL query size.	10000
expand-in-clause	If true, an SQL `IN`-clause will be expanded into `OR` conditions.	false
table-expression-pattern	Determines how a table expression is written in the SQL query; `catalog` and `schema` may be optional (surrounded by brackets).	[$$catalog$$.][$$schema$$.]$$table$$
procedure-expression-pattern	Determines how a procedure expression is written in the SQL query.	[$$catalog$$.][$$schema$$.]$$procedure$$
procedure-table-jdbc-type	Integer representing the jdbc type identifying a table returned form a procedure as defined by java.sql.Types.	0
procedure-table-type-name	Display name for tables from procedure. This is currently not visible to the user in any UI.	null
date-format-expression	An expression that converts a date field to a string value on the format: `YYYY-MM-DD`, for example, 2002-11-19. Used in `WHERE` and `HAVING` clauses. The tag `$$value$$` is a placeholder for the date field.	$$value$$
date-literal-format-expression	An expression that converts a date literal on the format `YYYY-MM-DD` to a date field value. Used in `WHERE` and `HAVING` clauses. The tag `$$value$$` is a placeholder for the date literal.	'$$value$$'
time-format-expression	An expression that converts a time field to a string value on the format: `HH:MM:SS`, for example 14:59:00. Used in `WHERE` and `HAVING` clauses. The tag `$$value $$` is a placeholder for the time field.	$$value$$

Setting	Description	Default value
time-literal-format-expression	An expression that converts a time literal on the format HH:MM:SS to a time field value. Used in `WHERE` and `HAVING` clauses. The tag `$$value$$` is a placeholder for the time literal.	'$$value$$'
date-time-format-expression	An expression that converts a datetime field to string value on the format: `YYYY-MM-DD HH:MM:SS`, for example 2002-11-19 14:59:00. Used in `WHERE` and `HAVING` clauses. The tag `$$value$$` is a placeholder for the date-time field.	$$value$$
date-time-literal- format-expression	An expression that converts a date-time literal on the format YYYY- MM-DD HH:MM:SS to a date-time field value. Used in WHERE and HAVING clauses. The tag $$value$$ is a placeholder for the date-time literal.	'$$value$$'
java-to-sql-type-conversions: • String • Integer • Long • Float • Double • Date • Time • DateTime	Type conversions needed when a join data source creates a temporary table for result from a subquery. For String conversion %s will be replaced by the size of the string. A match-length attribute may be specified (see MySQL). Different String types may be needed dependant of the length of the string. Note that there must be a `VARCHAR` conversion for when the length of the string is unknown (255 in the example here). When several VARCHAR mappings are specified, the mapping that first matches the match-length is used.	VARCHAR($$value$$) VARCHAR(255) INTEGER BIGINT REAL DOUBLE PRECISION DATE TIME TIMESTAMP
temp-table-name-pattern	Determines how to format a temporary table name in an SQL command.	$$name$$
create-temp-table-command	SQL commands for creating a temporary table. This is used to store filter values (when more than `condition-list-threshold`) and to store result from subqueries. The syntax may vary between databases. `$$name$$` is a placeholder for the table name. `$$column_list$$` is a placeholder for a column list on the format (name type, name type, ...).	CREATE TEMPORARY TABLE $$name$$ ($$column_list$$)

Setting	Description	Default value
drop-temp-table-command	SQL commands for deleting a temporary table. The syntax may vary between databases. `$$name$$` is a placeholder for the table name	DROP TABLE $$name$$
data-source-authentication	Default value data source authentication. (boolean). This value can be set (overridden) in the Information Interaction Designer.	false
lob-threshold	Threshold when LOB values used as parameters in a WHERE clause, must be written in temporary tables. The default means no limit.	-1
use-ansi-join	The default generated SQL creates joins with where statements. If this setting is set to true, an attempt is made to rewrite it to standard ANSI format. If this setting is set to false, no attempt to rewrite inner joins will be made and outer joins depend on the value set for `use-ansii-style-outer-join`.	false
use-ansii-style-outer-join	The default generated SQL uses the Oracle way with "(+)" to indicate joins. If this setting is set to true an attempt is made to rewrite it to standard ANSI format, making it possible to run on non Oracle databases.  If `use-ansi-join` is set to true, then this setting has no effect.	false
credentials-timeout	Defines the time in seconds user credentials are cached on the server for a particular data source. Value must be between 900 (15 minutes) and 604800 (1 week). Applicable only if `data-source-authentication` is set to true.	86400 (24 hours)

JDBC connection properties

The optional <`connection-properties` > parameter block in the configuration can be used to define JDBC connection properties parameters to be used when connecting to the data sources of the given type. A typical use case is to specify encryption and integrity checksum algorithms for secure database connections.

Each connection property consists of a key-value pair. The syntax for specifying JDBC connection properties for a connection pool is shown in the configuration example below.

If you need different JDBC connection properties for different data sources of the same type, just duplicate the <jdbc-type-setting> configuration, rename the configurations for each variant needed, and define the proper JDBC connection properties. Make sure to update any already existing data sources so that they are of the correct type.

Example: Defining JDBC connection Properties for data source of type oracle. This example creates an encrypted connection to the database.

```
<jdbc-type-settings>
 <type-name>oracle</type-name>
 <driver>oracle.jdbc.OracleDriver</driver>
 <connection-urlpattern>jdbc:oracle:thin:@&lt;host&gt;:&lt;port1521&gt;:&lt;sid&gt;</
connection-url-pattern>
 <ping-command>SELECT 1 FROM DUAL</ping-command>
 <connection-properties>
 <connection-property>
 <key>oracle.net.encryption_client</key>
 <value>REQUIRED</value>
 </connection-property>
<connection-property>
 <key>oracle.net.encryption_types_client</key>
 <value>( 3DES168 )</value>
 </connection-property>
 <connection-property>
 <key>oracle.net.crypto_checksum_client</key>
 <value>REQUIRED</value>
 </connection-property>
 <connection-property>
 <key>oracle.net.crypto_checksum_types_client</key>
 <value>( MD5 )</value>
 </connection-property>
 </connection-properties>
 ...
</jdbc-type-settings>
```

Advanced connection pool configuration

Information Services uses the same underlying connection pool implementation as Spotfire Server uses for connecting to its own database. The following special parameters are available to configure some of the aspects of that connection pool.

Special parameter	Corresponding common parameter
spotfire.pooling.data.source.scheme	pooling-scheme
spotfire.pooling.data.source.connection.timeout	connection-timeout
spotfire.pooling.data.source.login.timeout	login-timeout
spotfire.kerberos.login.context	kerberos-login-context

For more information, see Database connectivity.

All these parameters should be added as JDBC connection properties. However, they are never used as real JDBC connection properties and are never sent to a database server.

Example: Configuring a connection pool for Oracle databases

```
<jdbc-type-settings>
 <type-name>oracle</type-name>
 <driver>oracle.jdbc.OracleDriver</driver>
 <connection-urlpattern>jdbc:oracle:thin:@&lt;host&gt;:&lt;port1521&gt;:&lt;sid&gt;</
connection-url-pattern>
 <ping-command>SELECT 1 FROM DUAL</ping-command>
 <connection-properties>
 <connection-property>
 <key>spotfire.pooling.data.source.scheme</key>
 <value>WAIT</value>
```

```
 </connection-property>
 <connection-property>
<key>spotfire.pooling.data.source.connection.timeout</key>
 <value>1800</value>
 </connection-property>
 <connection-property>
 <key>spotfire.pooling.data.source.login.timeout</key>
 <value>30</value>
 </connection-property>
 </connection-properties>
 ...
</jdbc-type-settings>
```

Kerberos authentication for JDBC data sources

Configuring Kerberos authentication for JDBC data sources is similar to configuring Kerberos for the connection to the Spotfire database.

For more information, see Using Kerberos to log in to the Spotfire database.

This is an example of configuring a connection pool for Oracle databases:

```
<jdbc-type-settings>
  <type-name>oracle</type-name>
  <driver>oracle.jdbc.OracleDriver</driver>
  <connection-urlpattern>jdbc:oracle:thin:@&lt;host&gt;:&lt;port1521&gt;:&lt;sid&gt;</
connection-url-pattern>
  <ping-command>SELECT 1 FROM DUAL</ping-command>
  <connection-properties>
   <connection-property>
    <key>spotfire.kerberos.login.context</key>
    <value>DatabaseKerberos</value>
   </connection-property>
   <connection-property>
    <key>oracle.net.authentication_services</key>
    <value>( KERBEROS5 )</value>
   </connection-property>
  </connection-properties>
  ...
</jdbc-type-settings>
```

Creating an Information Services data source template using Kerberos login

The default Information Services Data Source templates that are included with Spotfire Server are not configured to use Kerberos. You must therefore create a new data source template based on one of the default templates.

Procedure

1. List the existing data source templates by using the list-ds-template command and select one that matches the database you are setting up, for example Oracle.
2. Export the definition of the selected data source template by using the export-ds-template command.
3. Open the exported definition file in a text editor.
4. Add the JDBC connection property key `spotfire.connection.pool.factory.data.source` with the value `kerberos.data.source` within the `connection-properties` element. If there is no `connection-properties` element, create one.

   There may also be other connection properties you must add; consult the documentation of the database server for more information. For general instructions about adding connection properties, see JDBC connection properties.

   Example:

   ```
   <jdbc-type-settings>
    <type-name>oracle</type-name>
   ```

```
 <driver>oracle.jdbc.OracleDriver</driver>
 <connection-urlpattern>jdbc:oracle:thin:@&lt;host&gt;
:&lt;port1521&gt;:&lt;sid&gt;</
connection-url-pattern>
 <ping-command>SELECT 1 FROM DUAL</ping-command>
<connection-properties>
 <connection-property>
 <key>spotfire.connection.pool.factory.data.source</key>
 <value>kerberos.data.source</value>
 </connection-property>
<connection-property>
 <key>oracle.net.authentication_services</key>
 <value>(KERBEROS5)</value>
 </connection-property>
</connection-properties>
```

5. Use the add-ds-template command to add the new data source template with a suitable name, such as "oracle_kerberos", using the modified template definition.
6. Import the configuration and restart the server.

What to do next

Verify the data source template

Verifying a data source template

Procedure

1. Log in to Spotfire Analyst as an administrator.
2. Select **Tools** > **Create Information Link**
3. Click **Setup Data Source**.
4. Enter a name for the data source connection.
5. Specify the type of data source.
6. Enter the **connection URL** and **max/min-values** for the connection pool.
7. Enter a username and a password to connect to the database.

   This does not apply to Kerberos.
8. Click **Save**.
9. In the left pane, click the **Data sources** tab.

Result

The data source name should appear in the tree to the left, ready for use.

Setting up MySQL5 vendor driver

For the MySQL5 vendor driver to work with MySQL data sources that include TIMESTAMPS that can potentially be null, you must edit the template.

Procedure

1. In the MySQL5 data source template, locate the following section:

```
<connection-properties>
 <connection-property>
 <key>useDynamicCharsetInfo</key>
 <value>false</value>
 </connection-property>
```

```
</connection-properties>
```

2. Within the `connection-properties` tag, add the following code:

```
<connection-property>
 <key>noDatetimeStringSync</key>
 <value>true</value>
</connection-property>
<connection-property>
 <key>zeroDateTimeBehavior</key
 <value>convertToNull</value>
</connection-property>
```

Information Services settings

Information Services provides end users with the ability to access and pivot data from multiple databases simultaneously, without having to know anything about installing database drivers, underlying data schemas or SQL.

End users' access to data from multiple sources can be configured and controlled through settings in Information Services. Below is a list of common settings with short descriptions. These are server settings, and you cannot use the settings for individual data sources.

For instruction on changing the settings, see Manually editing the Spotfire Server configuration file.

Setting	Description
`information-services.jdbc.oracle.use-faster-schema-listing`	List all Oracle users as schema list.
`information-services.dat.no-sbdf`	Use Spotfire text data format or Spotfire binary data format when transferring data from Spotfire Server to a Spotfire client.
`information-services.runtime-query-validation`	Validate information link prior to execution.
`information-services.dat.data-block-queue-size`	Maximum number of queued (not yet consumed by client) data blocks per job.
`information-services.dat.idle-limit`	Maximum idle time in seconds before a job is garbage collected.
`information-services.dat.max-field-size`	Maximum size (in Megabytes) for a data cell.
`information-services.dat.max-jobs`	Maximum number of concurrent jobs.
`information-services.dat.max-timeout`	Maximum value of timeout parameters; must be at least 60 seconds less than the idle limit.
`information-services.dat.pivot.thread-pool-size`	Maximum number of pivot worker threads.
`information-services.dat.reshape.max-memory-usage`	Maximum memory available to a reshape operation.
`information-services.dat.retrieve-timeout`	Maximum time allowed for retrieve requests, in seconds.
`information-services.dat.thread-pool-size`	Maximum number of job worker threads.
`information-services.ds.credentials-cache-timeout`	The default expiration time in seconds for cached data source authentication credentials.
`information-services.ds.credentials-provider`	The class used to provide credentials for data sources that require authentication.

Setting	Description
`information-services.jdbc.connection-login-timeout`	Login timeout for JDBC database connections.
`information-services.jdbc.oracle.temp-table-grantee`	Selecting privileges on temporary tables used during query execution will be granted to this user or role. The temporary tables are only valid during the query transaction.
`information-services.jdbc.use-inner-select-in-clause`	This setting affects the behavior when the number of filter values sent to a jdbc data source exceeds the condition-list-threshold. If set to false (default): all data rows matching any duplicate filter values will be duplicated, If set to true: data rows matching any duplicates will not be duplicated (the same behavior as when the number of filter values is below the condition-list-threshold limit), but there is a large performance penalty.
`information-services.parameter-expression-validation`	Determines whether parameters in information link queries should be validated when executed. The default value is `true`. Sometimes this validation results in errors, for example if you use extensions that create custom SQL parameters, or parameters of the type 'undefined'. To disable the validation process, set the value of this setting to `false`. ⚠ When validation is disabled, it is possible to execute arbitrary SQL.

Information Services commands

To perform an Information Services task, use the related command.

All commands in this group work on the `configuration.xml` file, which must be imported using the import-config command for any changes to take effect.

For instructions on using the commands, see Executing commands on the command line.

Task	Command
Add a new data source template.	add-ds-template
Clear the default join database configuration.	clear-join-db
Configure the default join database.	create-join-db
Export the definition of a data source template.	export-ds-template
List the data source templates.	list-ds-template
Modify a data source template.	modify-ds-template
Remove a data source template.	remove-ds-template
Show the configured default join database.	show-join-database

Default join database

The default join database is used for creating temporary tables and joining the final result when running an information link.

Most often using the standard Spotfire database for the default join database will work fine. However, in certain situations you may want to configure another database to be used. For example, if you prefer to run these operations as a specific user on the database, or if you want to use a database that is specifically optimized for temporary tables.

To set up a default join database use the command create-join-db.

Default join database settings

Option	Description
`Type`	Sets the type of database and driver you want to use as the default join database. Refers to a data source template.
`Connection URL`	The connection URL to the database.
`Number of Connections`	A minimum and maximum number of connections to use when accessing the database.
`Username and Password`	The username and password that will be used to access the database.

Post-installation steps

After Spotfire Server is installed and configured, the Spotfire administrator must complete these setup tasks before end users can access and work in Spotfire.

1. If you have not done so yet, install Spotfire Analyst on a computer for the administrator to use.

   Steps 4 and 5 in this list require Spotfire Analyst.

2. Set up groups and assign *licenses* to them; see Groups and licenses for details.
3. Set up users and assign them to appropriate groups; see Users for details.
4. Optional: Assign preferences to groups; use the Administration Manager in Spotfire Analyst to accomplish this.
5. Set up the Spotfire library by using Spotfire Analyst.
6. Optional: Import geocoding tables into the library so that data can be displayed on maps; see Enabling geocoding tables for map charts.

Enabling geocoding tables for map charts

To display data on a Spotfire map, the data must be "geocoded". This involves matching the data to location identifiers in a set of data tables that are known as a geocoding hierarchy. These geocoding tables must be imported into the library before they can be used.

Prerequisites

Spotfire Analyst is installed.

Procedure

1. Copy the file `<Spotfire Server installation kit>/geoanalytics/geoanalytics.part0.zip` to the library folder that is used for importing and exporting files. (By default, this is `<server installation directory>/tomcat/application-data/library`.)
2. Log in to Spotfire Analyst as a Spotfire Administrator or Library Administrator.
3. Click **Tools** > **Library Administration**.
4. Click **Import** and then browse to and select the file `geoanalytics.part0.zip`.
5. Click **OK** twice, and then in the Select Destination Folder dialog, either select an existing folder or create a new one (for example, you can create a "GeoAnalytics" folder).
6. Click **OK**, wait for the dialog to display the words "Import done", and then click **Close**.

Administration

Administrators perform most management tasks in Spotfire Server, including creating users and groups, setting licenses, deploying software updates, and managing and monitoring software configurations.

Setting preferences, however, and managing the library, take place in Spotfire Analyst.

Opening Spotfire Server

You can access Spotfire Server through a browser on any computer in the domain.

There are two ways to open Spotfire Server:

- On the computer running Spotfire Server, click **Start**, go to the Spotfire Server folder, and click **TIBCO Spotfire Server**.
- On any computer in the domain, go to http://*servername*:*port*/spotfire.

If you work in a clustered environment, it does not matter which server in the cluster you use. Changes made to one server are stored in the Spotfire database and are available to all servers. If your clustered deployment includes a load balancer, use the load balancer hostname in place of `servername` in the second method.

Nodes, services, and resource pools

In Spotfire Server you can enlarge or scale down your implementation as needed, as well as create and manage *resource pools*. Resource pools are used in *routing rules* to direct Spotfire traffic to specific service instances.

For more information, see Nodes and services introduction, Node manager installation, and Routing rules.

Creating resource pools

If you want a certain analysis, or all analyses requested by certain users, to open on specific instances of the Spotfire Web Player, create a resource pool that contains the selected instances and use it in a routing rule.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Resource pools" page, click **Create resource pool**.
3. In the "Create new resource pool" dialog, enter a name for the pool, and select the check box of each Spotfire Web Player instance that you want to add to the pool.

   Each Spotfire Web Player instance can belong to only one resource pool.
4. Click **Create**.
   The new pool appears in the Resource pools list.

Adding resources to resource pools

To respond to changing needs in your organization, you can adjust the contents of resource pools at any time.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. Click the **Resource pools** tab.
3. In the "Resource pools" table, locate the pool that you want to change and then click the plus sign on the right side of its row.
4. In the "Add instances to resource pool" dialog, select the check box for each instance that you want to add.
5. Click **Add**.

Removing resources from resource pools

To respond to changing needs in your organization, you can adjust the contents of resource pools at any time.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. Click the **Resource pools** tab.
3. In the "Resource pools" table, locate the pool that you want to change and then click the down arrow in its "AVAILABLE" box.
   This displays a list of the instances that the resource pool currently contains.
4. Above the list of instances, on the right, click the pencil icon.
   Check boxes are displayed to the left of each instance.
5. Select the check boxes of the instances that you want to remove from the pool, and then click **Remove**.
   The removed instance(s) are added to the "Unassigned instances" section.

Changing the name of a resource pool

You can rename a resource pool directly in the "Resource pools" list.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. Click the **Resource pools** tab and then, in the list of resource pools, click the name you want to change.
3. Make your changes, and then click the check mark.

Deleting resource pools

You can delete any resource pool that is no longer being used in a routing rule.

Prerequisites

Make sure that the resource pool is not in use by reviewing the "Resource pool" column of the Rules list in **Scheduling & Routing**.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.

2. Click the **Resource pools** tab.
3. In the "Resource pools" table, locate the pool that you want to delete and then click the trash icon on the right side of its row.

Updating node managers

When you add a node manager software update (hotfix) to the appropriate deployment area, an **Update** button is displayed in the information pane for each affected node.

Prerequisites

The software update is in the node manager's deployment area; for instructions, see Adding software packages to a deployment area.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, click **Nodes**, and then select the node that you want to update.
   In the upper-right pane there is an **Update** button.
3. Click **Update**, and then in the confirmation dialog click **Update** again.
   A message indicates that the update has started, and then the **Status** line indicates that the node is offline.

Result

When the **Roll back** button appears in the upper-right pane, the update is complete.

If you want to cancel the update and return to the previous node manager version, see Rolling back a node manager update.

Rolling back a node manager update

After updating a node manager, you have the option of undoing the update and returning to the previous version of the node manager.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, click **Nodes**, and then select the node manager that was updated.
   In the upper-right pane there is a **Roll back** button.
3. Click **Roll back**, and then in the confirmation dialog click **Roll back** again.
   A message indicates that the rollback has started, and then the **Status** line indicates that the node is offline.

Result

When the **Update** button reappears, the rollback is complete.

Updating services

When you add an update for a service to the appropriate deployment area (or make any other change to a deployment, such as deleting a package or changing the deployment area of a service), an **Update service** button becomes available in the information pane for each affected service.

Updating a service will lead to a restart of the service and all instances under the service (including any instances that were previously stopped).

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, click **Nodes**, and then select the service that you want to update.
   In the upper-right pane there is an **Update service** button. You can scan the Packages pane for the orange notes that indicate exactly what has changed from the current deployment.
3. Click **Update service**, and then in the confirmation dialog click **Update**.
   In the upper-right pane, the **Status** line indicates that the update has started. The Activity page shows the progress of the update.

Result

When the update is complete, the **Status** line indicates "Service installed successfully". The new service duplicates the settings of the old service, including its name, resource pool, and port. No further requests will be routed to the old service.

If you want to cancel the update and return to the previous service version, see Rolling back a service update.

If you delete the old service you will not be able to roll back the service.

When the update is successful and you are sure that you want to keep the new version, you should delete the old service version. Because Spotfire Server stores a maximum of two versions of a service, if you perform another update on the same service, the first version will be deleted automatically if it is still being stored.

Rolling back a service update

After updating a service, you have the option of undoing the update and returning to the previous version of the service.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, click **Nodes**, and then select the service that was updated.
   The **Show old service** link is visible in the upper-right corner of the page.
3. Click **Show old service**.
   In the upper-right pane, information about the old service appears (in a paler font) to the right of information about the new service. A **Roll back** button becomes available in the upper-right corner of the page.
4. Click **Roll back**, and then in the confirmation dialog click **Roll back** again.
   The **Status** line indicates "Instances are being modified".
5. When the **Status** line indicates "Service is available but the functionality is limited until rollback is confirmed", click **Confirm rollback** in the upper-right corner of the page. In the confirmation dialog, click **Roll back**.

Result

The **Status** line indicates "Service installed successfully".

Shutting down a service instance

If you want to shut down a service instance because it is not needed, for example, or because you want to run it on a different node, you can shut the service down without disturbing the work of end users. You can also shut it down immediately.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, select **Nodes**.
3. In the left pane, expand the entries under the node and select the service instance that you want to shut down.
4. In the right pane, click **Shut down** and then do one of the following:
   - If you want the instance to continue running for a while, click **Schedule** and then enter the number of hours and minutes you want Spotfire Server to wait before shutting it down.

     Before the shutdown, any users on that service instance are notified that the instance will be shutting down; this gives them time to save their work. The instance is then shut down when the user or users close the analysis, or at the scheduled time, whichever is earlier. If no one is using the instance, the instance is shut down immediately.
   - If you want the instance to shut down immediately, whether or not it is being used, click **Immediately**.

     End users who are on this service instance will lose any unsaved work.

Revoking trust of a node

You may want to remove the authorization of a node because you are upgrading your hardware, for example, or down-scaling your network, or if you see an unusual error and want to reset the computer. This immediately shuts down any services that are running on the node, and disables all management options for the node except re-trusting it.

Procedure

1. Log in to Spotfire Server and click **Nodes & Services**.
2. On the "Your network" page, under **Select a view**, select **Nodes**.
3. In the left pane, select the node whose trust you want to revoke, and in the upper-right pane click **Revoke trust**.

Result

The node moves from the "Your network" page to the "Untrusted nodes" page.

Users

There are three types of Spotfire user accounts: local, external, and system. All of these accounts are registered in the Spotfire database. The administrator organizes the users into groups; the licenses that provide access to Spotfire features, as well as preference settings and other options, are applied at the group level.

For basic information, see Users introduction and Groups and licenses introduction.

Local user accounts

If your environment is configured for authentication towards the Spotfire database (as opposed to an external user directory), the administrator enters the user accounts directly in the Spotfire database. These accounts are called *local user accounts*.

There are several ways to do this:

- To add users manually, see Creating Spotfire users.
- To import users from a file, you can use the Administration Manager tool in Spotfire Analyst, or the **`import-users`** command. For instructions, see the Administration Manager User's Manual or the import-users topic.

  If you want to change servers, you can export the current users and groups from the Spotfire database on one server and then import and reuse the information on a different server. This can be done in the Administration Manager in Spotfire Analyst, or by the **`export-users`** or **`export-groups`** command. For instructions, see the Administration Manager User's Manual, or the export-users or export-groups topic.

This is an example of a local user listed on the Users page of the Spotfire administration interface:

As indicated, local users belong to the SPOTFIRE domain.

Local user accounts are administered in Spotfire Server.

For administration on the server, use the procedures in this section of the manual, or the related commands that are listed, if available, at the top of a procedure.

For information about configuring authentication towards the Spotfire database, see Saving basic configuration data (authentication towards Spotfire database).

External user accounts

If your environment is configured for authentication towards an external user directory such as LDAP, or an external authentication provider such as Google, these external user accounts are added and administered in that context rather than in the server. Changes are then copied to the Spotfire database during synchronization.

The specific administration tasks that are accomplished in the external context rather than in Spotfire depend on how your Spotfire environment is configured. For example, if you acquire users, but not groups, from an external source, assigning the users to groups takes place in Spotfire.

External users keep their domain name from the external directory, and the domain name appears as part of their username throughout the Spotfire interface.

If you want to change servers, you can export the current users and groups from the Spotfire database on one server and then import and reuse the information on a different server. This can be done in the Administration Manager in Spotfire Analyst, or by using the **`export-users`** or **`export-groups`** command. For instructions, see the Spotfire Administration Manager User's Manual, or the export-users or export-groups topics.

This is an example of an external user listed on the Users page of the Spotfire administration interface:

The format of the username will vary depending on your user directory or authentication provider.

For information about configuring external authentication, see User authentication.

System user accounts

Spotfire contains five system user accounts that are used internally in the Spotfire environment.

These accounts, which are listed below, are present at installation and cannot be deleted or renamed.

- Automation Services System Account
- Monitoring System Account
- Node Manager System Account
- SBDF (Spotfire Binary Data File) Cache System Account
- Scheduled Updates System Account

This is an example of a system account listed on the Users page of the Spotfire administration interface:

As indicated, system accounts belong to the SPOTFIRESYSTEM domain.

There is also a Guest system account that belongs to the ANONYMOUS domain. This account is used only to provide access to public information on the web client through anonymous authentication. For more information, see Configuring anonymous authentication.

Creating Spotfire users

If your Spotfire environment is configured for Spotfire database authentication, you can manually add new users using the Spotfire Server web interface.

For more information, see Users introduction and Users.

Related command: create-user

Procedure

1. Log in to Spotfire Server. (For instructions on accessing the server, see Opening Spotfire Server.)
2. Click **Users & Groups**.
3. Under **Manage**, select **Users**.
4. On the Users page, click **Create user**.
5. On the Create user page, enter the username and password.

The username must be unique within your Spotfire environment. The display name, which is used in the interface to identify the user, defaults to the username but it can be changed.

6. Click **Save**.
   Now you can create another user, or click **Cancel** to view the complete users list.

Result

To view the profile of the new user, right-click the user's name and then click **Edit**.

Adding a user to one or more groups

A user can belong to one or many groups. A user who is an explicit member of a group is also, by inheritance, a member of that group's parent groups. All users automatically belong to the Everyone group and cannot be removed.

For more information, see Groups and licenses introduction and How licenses work.

Related commands: add-member, copy-group-membership

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. On the Users page, right-click the user that you want to add to groups, and then click **Edit groups**.
4. Click **Add to groups**.
5. In the Select groups dialog, select the check box next to the groups to which you want to add the user, and then click **Add**.

Result

The selected groups are displayed in the user's Groups list.

Viewing user profiles

User profiles contain the basic information about a user, plus their assigned deployment area and their composite name. Depending on the type of user and how your environment is configured, some of the information on a user's Profile page will be editable.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. Right-click the name of the user whose profile you want to edit, and then click **Edit profile**.

Result

On the Profile page, under **Deployment area**, the deployment area or areas that the user has access to are listed. These are the areas that are assigned to the groups that the user belongs to, and indicate which software packages the user has access to. For more information, see Deployments and deployment areas.

For external and system users, the **Composite name** is automatically formed from the user's domain and username.

The **Display name** is used in the interface to identify the user.

The **Invalidate persistent sessions** button is available if the user selected the **Keep me logged in** check box on the login page. For more information, see Persistent Spotfire sessions.

Viewing user licenses

The licenses and license features that are enabled for a user determine the Spotfire features that are available to the user. Licenses are set for groups, never for individual users.

For more information, see Groups and licenses.

Related command: show-licenses

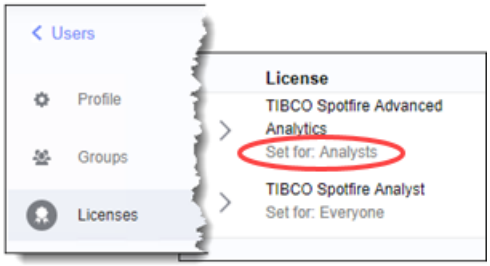
Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. Right-click the name of the user whose licenses you want to see, and then click **View licenses**.

Result

The Licenses page shows the licenses that are enabled for the user. This is the sum of all the licenses that are set for the groups to which the user belongs, and for the parent groups of those groups.

As shown in the following example, the "Set for" line indicates that the TIBCO Spotfire Advanced Analytics license was enabled for the Analysts group, and the Spotfire Analyst license was enabled for the Everyone group.



Removing a user from one or more groups

You can remove a user from a group to remove the user's access to the licenses that are enabled for that group, and any groups that are parent to that group.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. On the Users page, right-click the user that you want to remove from groups, and then click **Edit groups**.
4. On the user's Groups page, select the check box next to the groups from which you want to remove the user, and then click **Remove from group**.

Result

The removed groups no longer appear in the user's Groups list.

Changing a user's name, password, or email

Administrators can change these settings for *local users*. Externally synchronized users are managed in that context and not within the Spotfire system.

These settings cannot be edited for system user accounts.

Related command: set-user-password

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. On the Users page, right-click the user that you want to edit, and then click **Edit profile**.
4. On the user's Profile page, edit any of the available user settings.

   The **Invalidate persistent sessions** button is available if the user selected the **Keep me logged in** check box on the login page. For more information, see Persistent Spotfire sessions.
5. When you've finished, click **Save**.

Persistent Spotfire sessions

A user session is considered "persistent" when the server retains the user's successful login after the user has stopped working or closed the program. This makes it possible for the user to reopen Spotfire and continue working without logging in again.

The "persistent sessions" feature applies only to username/password authentication.

Persistent sessions can create a security risk. Spotfire provides several ways to control or remove this session information from the server.

- In the administration interface, on a user's Settings page, you can invalidate any current persistent sessions for the selected user. For instructions on accessing this page, see Changing a user's name, password, or email.
- You can use the invalidate-persistent-sessions command to invalidate persistent sessions for all users, or for a specified user.
- You can use the config-persistent-sessions command to set how persistent sessions work for web-based users in your environment, or to disable the feature.
- You can use the config-login-dialog command to set how persistent sessions work for users who log in to Spotfire Analyst.

Disabling user accounts

Disabling a user account makes it impossible for the user to log in to Spotfire, but keeps their record in the system for reference or for enabling them again in the future.

Related command: enable-user

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. On the Users page, right-click the user account that you want to disable, and then click **Disable**.

   To disable several user accounts at the same time, select each account, right-click one of them, and then click **Disable**.

Result

The disabled account appears greyed out in the Users list.

Deleting users from the system

To permanently remove users and their records from your Spotfire implementation, delete them. However, if you want to deny users access to Spotfire but keep their records in the system, you can disable their accounts instead.

Externally synchronized users are managed in that context and not within the Spotfire system.

Related command: delete-user

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. Under **Manage**, select **Users**.
3. On the Users page, right-click the user that you want to delete, and then click **Delete**.

To delete several users at the same time, select each user, right-click one of them, and then click **Delete**.

Groups and licenses

Users' group membership determines the Spotfire features that they have access to, and therefore their role in the Spotfire environment.

Spotfire comes with a set of system groups that correspond to common user roles, such as Library Administrator or Script Author.

For groups that are synchronized from an external source such as an LDAP directory, certain tasks, including adding and removing members of the synchronized group, take place in the external environment and not within the Spotfire system.

For more information, see Groups and licenses introduction.

Access to library folders can be set for both groups and individual users; for more information, see the Spotfire Analyst User's Guide. Preferences are also set at the group level; for more information, see Preferences introduction. The tools for administering preferences and library access are available in Spotfire Analyst.

How licenses work

Licenses determine the Spotfire features and functionality that are available to users. Licenses are set at the group level, so when creating the groups that your organization requires, you should also set the licenses that apply to each group.

For more information, see Groups and licenses introduction.

Groups can contain other groups as well as individual users, making it possible to build a hierarchy of groups. Within the hierarchy, in addition to any licenses that are assigned directly in the groups that a user belongs to, the user also inherits the licenses of all the groups above them in the hierarchy.

The idea is to create a hierarchy of groups, where you assign licenses as high up in the hierarchy as possible. The top groups contain the basic Spotfire functionality that all users need. Then, for more specialized groups farther down in the hierarchy, you can enable more advanced licenses to supplement the basic licenses that they inherit.

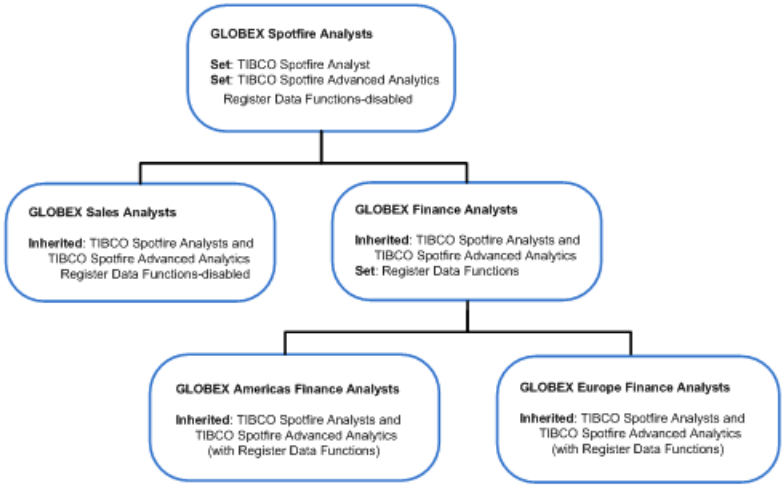
If a certain feature is enabled in one group, and disabled in another group, a user that is a member of both groups (whether directly or through inheritance) *will* have access to the feature.

If a group's parent groups contain conflicting settings, you can control which group has precedence by setting a primary group; for more information, see Assigning a primary group to a subgroup.

The following three examples demonstrate how inheritance works among groups, and gives an idea of how these features can be used. In the examples, "set" features are enabled directly in the indicated group, as opposed to being inherited.

Finally, A group hierarchy template offers some best practices to consider when structuring your groups.

Example 1



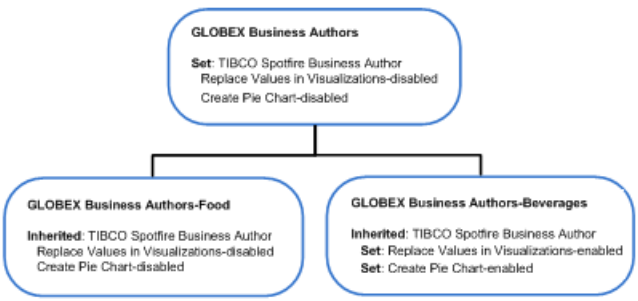
In the first example, the group GLOBEX Spotfire Analysts contains all the analysts at the company. The administrator sets the Spotfire Analyst and Spotfire Advanced Analytics licenses for that group so that all analysts have access to the features that are included in those licenses. But not all analysts should be able to register data functions, so the administrator does not enable that feature of the Advanced Analytics license.

Finance Analysts do need to register data functions, so the administrator creates separate groups for Sales and Finance Analysts, and sets the Register Data Functions feature for the Finance group.

The administrator then decides to split the Finance Analysts into two subgroups, and does not add any users to the GLOBEX Finance Analysts group. Instead, the administrator creates the subgroups Americas Finance Analysts and Europe Finance Analysts, and adds the appropriate users at that level. Members of both of these groups inherit the Register Data Functions feature from their parent group.

Because library permissions can also be set for groups, the administrator will later provide separate library sections so that the Americas Finance Analysts can share analyses in one section, and the Europe Finance Analysts in another.

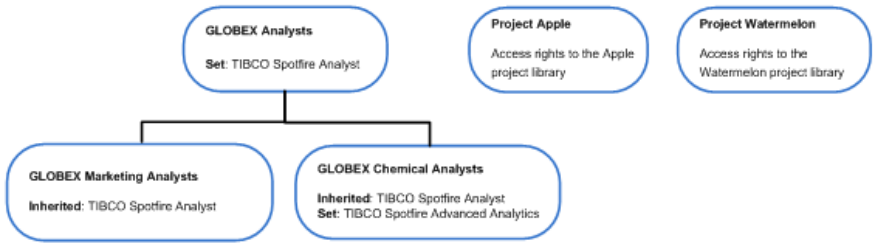
Example 2



In the second example, the administrator assigns the Spotfire Business Author license to the GLOBEX Business Authors group, but disables the Replace Values in Visualizations and Create Pie Chart features because the average Business Author should not be allowed to perform these tasks.

However, for the group Business Authors-Beverages, the administrator explicitly enables these two features because the members of this group require the features to do their job. The users working with food analyses do not get access to these features; they have only the licenses and features that are set for their parent group.

Example 3

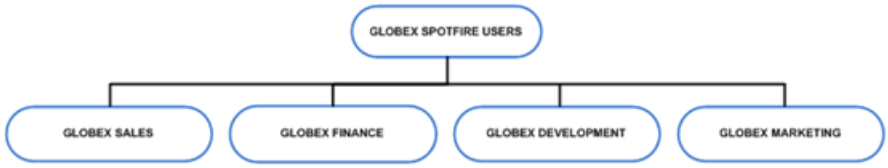


Not all groups need to be placed in the same hierarchical, top-down tree. It can, for example, be powerful to handle licenses in a hierarchical tree, but parallel to that create a number of separate top-level groups that correspond to another property of your company, such as projects. These groups could be used to handle Spotfire library privileges. (Library access rights are configured in Spotfire Analyst.)

A group hierarchy template

This group hierarchy may serve as a starting point for designing a logical structure to meet the needs of your organization. Ease of maintenance is paramount, so design a hierarchy that can quickly accommodate personnel changes and shifting user responsibilities.

In many cases, best practice is to begin by separating out the functional units of the organization. This is an example:



Next, design a structure to be repeated in each second-level unit. The following example features a special group for Administrators, a Center of Excellence (COE) to support users of various ability, and an IT group to handle the most technical tasks.

The third level, consisting of Consumers, Business Authors, and Analysts, corresponds to how Spotfire is provided to the organization. Under Business Authors and Analysts there will be groups with different responsibilities that require different licenses to be enabled.

Of course no example will fit every company. Large companies often work with Spotfire specialists to design an implementation that covers present needs and is flexible enough to handle growth and change.

System groups

Spotfire comes with a set of system groups that correspond to common user roles, such as Library Administrator and Script Author. System groups are created at installation and cannot be removed.

Users who require the permissions that are granted by one or more of these groups must either be added to the appropriate group, or to a subgroup of that group.

Group name	Description	Notes
Administrator	Membership grants administrator privileges on Spotfire Server, including the ability to manage users and groups. Members are given all permissions described below, in addition to administration of preferences in the Administration Manager tool in Spotfire Analyst.	The Spotfire Administration license is automatically assigned to this group and cannot be removed.
Anonymous User	This group is used for anonymous authentication. It contains the guest@ANONYMOUS user.	
API User (obsolete)		Only appears on upgraded systems.
Automation Services Users	Membership grants permission to schedule Automation Services jobs in the Spotfire Server administration interface, and to execute Automation Services jobs on the server by using the administration interface, the Job Builder, or the Client Job Sender. To use the Job Builder, users must also have the "Tibco Spotfire Extensions" license or the "Automation Services Job Builder Tool" feature, which is part of the TIBCO Spotfire Extensions license.	By default, the user account Automation Services System Account is a member of this group. ⚠ Do not remove this account unless you are sure of what you are doing. It is also possible to configure Automation Services to use a Kerberos account or a custom Spotfire account.

Group name	Description	Notes
Custom Query Author	Membership grants permission to save custom queries as trusted to the library. Only trusted custom queries will run in web clients.  An authorized custom query author MUST ALSO have the Custom Query in Connections feature, which is part of the TIBCO Spotfire Analyst license.	
Deployment Administrator	Membership grants permission to deploy packages to the server by using the Deployments & Packages area. Members can deploy to any area on the server, as well as delete any existing deployment.	
Diagnostics Administrator	Membership grants permission to view logs and diagnostics, set logging configurations, download troubleshooting bundles, and so on. Members of this group can access the Monitoring & Diagnostics area of the server.	
Everyone	This group always contains all users in the Spotfire implementation except for the Anonymous users (guests).	No users can be removed from this group, but you can set licenses for the group if you want to.
Impersonator (obsolete)		Only appears on upgraded systems.
Library Administrator	Membership grants full permission to the library, including the ability to create new top level folders. It overrides all folder permissions set in the library, granting full control over content.  Library administrators must also have the "Library Administrator" license.	Users and groups that require administrative privileges in the library must belong to this group or the Administrator group.
Scheduled Updates Users	The user account that executes scheduled updates must be a member of this group.	By default, the user account scheduledupdates@SPOTFIRESYSTEM is a member of this group.  It is also possible to configure scheduled updates to use a Kerberos account or a custom Spotfire account.
Scheduling and Routing Administrator	Membership grants permission to create scheduled updates and routing rules. Members of this group can access the Scheduling & Routing area of the server.	For Spotfire implementations that are upgrading to version 7.5, the old "WebPlayer Administrator" group has been added as a subgroup to the Scheduling and Routing Administrator group to facilitate migration. However, in all new implementations, only the Scheduling and Routing Administrator role is required for creating scheduled updates and routing rules.

Group name	Description	Notes
Script Author	Membership grants permission to save scripts and data functions as trusted in the Spotfire library. Script authors also need the "Access to extensions" feature or the "Author scripts" feature, which are part of the Spotfire Extensions license.	Scripts and data functions that are executed by the Web Player or Spotfire Analyst can essentially do anything that deployed packages can do. Therefore, only trusted users should be granted this permission.
System Account	This group contains the system accounts that are used internally in the Spotfire environment.	This group cannot be edited.
Web Player Administrator	Legacy group included for Spotfire implementations that are upgrading to version 7.5. See the "Scheduling and Routing Administrator" notes, above.	

License feature reference

The following topics describe the features that are contained in each Spotfire license. When enabling licenses, individual features of the license can be enabled or disabled.

When users run Spotfire, license features that are not enabled for them will either not appear in the interface, or appear greyed out.

Related commands: list-licenses, show-licenses

Spotfire Administrator license

The Administrator license, along with membership in the Administrator group, provides full administrative control over all aspects of the Spotfire environment. This license also controls basic access to the Spotfire library.

Feature	Description
Administration	Provides access to all administrative tasks, including access to the Administration Manager tool and its functionality. The Administration Manager tool is available in Spotfire Analyst. In addition to having this license, administrators must also be members of the Administrator group for full administrator control of the library. Full control includes accessing and browsing the folders, modifying and saving items, and also changing permissions for folders and their contents. For details about the Administrator group, see System groups.
Library Administration	This license provides access to the Library Administration tool and its functionality. The tool is available in Spotfire Analyst. Users who require administration control over only certain parts of the library should not be members of the Administrator group; for these users, the Library Administration license feature is enough. This is the recommended option for users who will administer specific sections of the library in your company. Administrators who need full control over all content in the library must also be members of the Library Administrator group. Membership in the Library Administrator group overrides all folder permissions in the library. (By default, the Administrator group is a part of the Library Administrator group, so members of the Administrator group have full control over the library.)

Spotfire Advanced Analytics license

The Advanced Analytics license controls access to additional analytic features such as custom expressions, the box plot visualization, various statistical features, and advanced aggregation methods. Most users of Spotfire Analyst will need this license as well as the Spotfire Analyst license.

Feature	Provides permission to
Create Box Plot	Create a new Box Plot visualization.
Advanced Aggregation Methods	Access more aggregation methods, in addition to the standard Sum, Average, Count, and Unique Count.
Custom Expressions	Access the Custom Expressions option on the column selectors.
Set Column From Marked	Access the Column from Marked option on the column selectors.
Line Similarity	Access the Line Similarity tool.
K-Means Clustering	Access the K-Means Clustering tool.
Hierarchical Clustering	Use hierarchical clustering.
Data Relationships	Access the Data Relationships tool.
Curve Draw	Add a curve to a visualization by entering an expression for the curve.
Curve From Data Table	Add a curve to a visualization by importing it from a data table.
Straight Line	Add a Straight Line Fit to visualizations.
Polynomial Curve	Add a Polynomial Curve to visualizations.
Logistic Regression Curve	Add a Logistic Regression Curve to visualizations.
Power Curve	Add a Power Curve to visualizations.
Gaussian Curve	Add a Gaussian Curve to visualizations.
Logarithmic Curve	Add a Logarithmic Curve to visualizations.
Exponential Curve	Add an Exponential Curve to visualizations.

Feature	Provides permission to
Register Data Functions	Create, edit, import, and export data functions and expression functions that are in the Spotfire library. Specifically, users can: • Register new data functions. • Edit existing scripts, including data function parameters and output. • Refresh data functions and expression functions. • Delete data functions and expression functions from the library (if they are not embedded in an analysis). • Synchronize data functions. • Add data tables, and then add data function transformations to that data table. If this license feature is not granted, the **Tools** > **Register Data Function** and the **Data** > **Data Function Properties** menu items are not available in the Analyst client. Without this feature, users cannot create, edit, or delete data functions or expression functions. Even if users do not have this feature, however, their existing analyses that contain predicted columns will continue to function and update based on the data as necessary. (That is, the underlying data functions continue to work.)  This feature should be granted in conjunction with the Execute Data Functions feature, so that users can test (execute) as well as register data functions and expression functions. If you do not enable this feature, you should not enable the Predictive Modeling and Insert Predicted Columns features.
Execute Data Functions	Create, edit, run, and remove data functions and expression functions. Specifically, users can: • Edit existing scripts, including data function parameters and output. • Specify whether to run data functions locally or on a server with TIBCO Spotfire Statistics Services • Access and change settings for the type of data used by data functions. If this license feature is not granted, data functions cannot be accessed from the Files and data flyout, and the **Settings** button in the **Data** > **Data Table Properties** dialog is not available. Even if users do not have this feature, however, their existing analyses that contain predicted columns will continue to function and update based on the data as necessary. (That is, the underlying data functions continue to work.)  If you do not enable this feature, you should not enable the Register Data Functions, Predictive Modeling, and Insert Predicted Columns features.
Predictive Modeling	Create, edit, and manage Regression and Classification Models. Even if users do not have this feature, however, their existing analyses that contain results from the predictive modeling tools will continue to function and update based on the data as necessary. (That is, the underlying data functions continue to work.) If this feature is enabled, you should also enable the Register Data Functions and Execute Data Functions features.
Insert Predicted Column	Use predictive models in analyses to add predicted columns to data tables. Even if users do not have this feature, their existing analyses that contain predicted columns will continue to function and update based on the data as necessary. (That is, the underlying data functions continue to work.) If this feature is enabled, you should also enable the Register Data Functions and Execute Data Functions features.

Spotfire Analyst license

The Spotfire Analyst license is the primary license for users running the full Spotfire client.

All the base functionality of Spotfire is contained here, except for the features that are listed for the Spotfire Enterprise Player license. It is therefore vital that any user who has the Spotfire Analyst license also get the Spotfire Enterprise Player license.

Spotfire Analyst users might also need the TIBCO Spotfire Business Author license feature, under TIBCO Spotfire Business Author to be able to move, edit or delete files from the web view of the library (even if they have Browse + Modify + Access rights to a library folder and mainly work with Spotfire Analyst).

Feature	Provides permission to
Action Links	Create action links.
Advanced Data Properties	Use additional functionality in the Column Properties dialog, such as Details, Edit, Freeze Columns, Insert, Column Properties, Sort Order, and Hierarchies.
Advanced Document Properties	Access the **File** > **Document Properties** dialog, the **Tools** > **Options** dialog, and also allows the user to assign a filtering scheme to a page.
Advanced Filter Panel Properties	Hide, move, and group filters, and access the Filtering Scheme Properties dialog and the Organize Filters dialog.
Advanced Visualization Properties	Change all visualization properties by using the Visualization Properties dialog. This feature also enables everything in the Change Column Selector and Simple Visualization Properties features.
Alerts for Streaming Data	Open a web user interface where it is possible to view and configure alert rules for streaming data tables from Spotfire Data Streams.
Change Column Selector	Interact with column selectors and data table selectors in visualizations. It also allows users to adjust binning sliders and hierarchy sliders, and to alter the order of multiple columns on an axis.
Change Exportability for Web Client Table Data	Specify whether the data in a Table, Summary Table, or Cross Table should be able to be exported if the analysis is opened in a Spotfire web client. If disabled, a default value or preference will determine if the data in the table can be exported from web clients. Set these preferences on the Preferences tab in the Administration Manager tool.
Create 3D Scatter Plot	Create new 3D scatter plot visualizations.
Create Annotations	Create annotations within analyses.
Create Bar Chart	Create new bar chart visualizations.
Create Combination Chart	Create new combination chart visualizations.
Create Cross Table	Create new cross table visualizations.
Create Graphical Table	Create new graphical table visualizations.
Create Heat Map	Create new heat map visualizations.

Feature	Provides permission to
Create Information Link	Create new information links using predefined elements.  This feature gives users access to the **Data** > **Information designer** menu option, from which they can use data sources and elements other people have configured to create a new information link. However, to configure data sources and elements, users must also have the Spotfire Information Modeler license.
Create KPI Chart	Create new KPI chart visualizations.
Create Line Chart	Create new line chart visualizations.
Create Map Chart	Create new nap chart visualizations.
Create Page	Create new pages.
Create Parallel Plot	Create new parallel coordinate plot visualizations.
Create Pie Chart	Create new pie chart visualizations.
Create Scatter Plot	Create new scatter plot visualizations.
Create Summary Table	Create new summary table visualizations.
Create Table	Create new table visualizations.
Create Text Area	Create new text areas.
Create Treemap	Create new treemap visualizations.
Create Waterfall Chart	Create new waterfall chart visualizations.
Custom Query in Connections	Author custom queries for data access using data connections.  To publish trusted custom queries to the library, a user must also be a member of the Custom Query Author group. Only custom queries saved as "trusted" to the library by such a member will execute in web clients.
Customize Toolbar	Customize the toolbar by adding shortcuts and arranging them in the desired order on the toolbar.
Data Action Recommendations	Enable the option to get recommendations on data actions. For example, this could be to clean up the column values by removing whitespace characters.
Data Panel	Use the Data in analysis flyout (the Data panel).
Error Bars	Display error bars in visualizations.
Insert New Column	Add and edit calculated columns.
Lists	Work with Lists.
Load Data on Demand	Load data from data connections and information links on demand.
Manage Data Tables	Add and work with data tables in an analysis, as well as add data to start a new analysis. Enables the possibility to add new data tables, to view and edit data table properties, to see the Data canvas (the source view in which to edit data table structures), as well as seeing the Manage Data Connections tool.

Feature	Provides permission to
Merge Data	Insert and merge additional data into a data table. This is done by selecting to add more data as new rows or new columns after choosing it in the Files and data flyout, or by inserting rows or columns between nodes in the Data canvas.
Open Custom Made File	Open custom made files.
Open Database Data	Open data from databases.
Open Empty Data	Open an analysis file even if a data source is unavailable.
Open File Data	Open local data files such as *.txt, *.csv,*.xlsx, *.xls, *.stdf, and *.sbdf files, and to paste data from the clipboard.  Users can always open *.dxp files if the Open File feature that is included in the Spotfire Enterprise Player license is enabled for them.
Open Information Links	Open data from information links.
Open Library Data	Open data files stored in the library.
Open Library Data Connection	Open data from data connections in the web client (provided that the data connection was previously stored in the library).
Recommendations (before 10.0)	Access to the previous Recommended visualizations dialog.
Replace Data	Enable the **Replace data table** button on the toolbar of the **Data canvas**, and also the **Replace data source** option for a selected data node. The available sources from which data can be replaced are determined by the 'Open...' license features.
Replace Values in Visualizations	Replace values in the context of a table visualization or in the Details-on-Demand. This is applicable to in-memory data only.
Simple Data Properties	Access the **Data** > **Column Properties** dialog.
Simple Filter Panel Properties	Access the context menu that is used for switching filter types and resetting filters.
Simple Visualization Properties	Change visualization properties using the axis selectors, and drag and drop columns from other visualizations or the Filters panel. This feature also enables everything in the Change Column Selector feature.
Tags	Work with Tags.
Use Custom Made Tools	Open custom made tools.

Spotfire Business Author license

The Business Author license contains the functionality for creating and editing analyses from the web client. (For technical reasons, some license features under this license might also be needed for administrative tasks in the web client.)

Feature	Provides permission to
Advanced Visualization Properties	Change all visualization properties by using the Visualization Properties panel. This feature also enables everything in the Change Column Selector and Simple Visualization Properties features.)
Change Column Selector	Interact with column selectors and data table selectors in visualizations. It also allows users to adjust binning sliders and hierarchy sliders, and to alter the order of multiple columns on an axis.
Create Annotations	Create annotations within analyses.
Create Bar Chart	Create new bar chart visualizations.
Create Combination Chart	Create new combination chart visualizations.
Create Cross Table	Create new cross table visualizations.
Create KPI Chart	Create new KPI chart visualizations.
Create Line Chart	Create new line chart visualizations.
Create Map Chart	Create new map chart visualizations.
Create Page	Create new pages.
Create Parallel Plot	Create new parallel coordinate plot visualizations.
Create Pie Chart	Create new pie chart visualizations.
Create Scatter Plot	Create new scatter plot visualizations.
Create Table	Create new table visualizations.
Create Treemap	Create new treemap visualizations.
Create Waterfall Chart	Create new waterfall chart visualizations.
Customize Toolbar	Customize the toolbar by adding shortcuts and arranging them in the desired order on the toolbar.
Data Action Recommendations	Enable the option to get recommendations on data actions. For example, this could be to clean up the column values by removing whitespace characters.
Data Panel	Use the Data in analysis flyout (the Data panel).
Error Bars	Display error bars in visualizations.
Insert New Column	Add and edit calculated columns.
Manage Data Tables	Add and work with data tables in an analysis, as well as add data to start a new analysis. Enables the possibility to add new data tables and to see the Data canvas (the source view in which to edit data table structures).
Merge Data	Insert and merge additional data into a data table. This is done by selecting to add more data as new rows or new columns after choosing it in the Files and data flyout, or by inserting rows or columns between nodes in the Data canvas.
Open Empty Data	Open an analysis file even if a data source is unavailable.

Feature	Provides permission to
Open File Data	Open local data files such as *.txt, *.csv,*.xlsx, *.xls, *.stdf, and *.sbdf files.  Users can always open *.dxp files if the Open File feature that is included in the Spotfire Enterprise Player license is enabled for them.
Open Information Links	Open data from information links.
Open Library Data	Open data files stored in the library.
Open Library Data Connection	Open data from data connections in the web client (provided that the data connection was previously stored in the library using Spotfire Analyst).
Recommendations (before 10.0)	Access to the previous Recommended visualizations dialog.
Replace Data	Enable the **Replace data** button on the toolbar of the **Data canvas**, and also the **Replace data source** option for a selected data node. The available sources from which data can be replaced are determined by the 'Open... ' license features.
Replace Values in Visualizations	Replace values in the context of a table visualization or in the Details-on-Demand. This is applicable to in-memory data only.
Simple Filter Panel Properties	Access the context menu that is used for switching filter types and resetting filters.
Simple Visualization Properties	Change visualization properties using the axis selectors. This feature also enables everything in the Change Column Selector license feature.
TIBCO Spotfire Business Author	Access the TIBCO Spotfire Business Author.  You also need this license to be able to move, edit or delete files from the web view of the library (even if you have Browse + Modify + Access rights to a library folder and mainly work with Spotfire Analyst).

Spotfire Consumer license

The Consumer license provides access to Spotfire Consumer, which users require to open analyses in the web browser.

Note that any group with this license enabled must also have the following two features enabled. These features are included in the Spotfire Enterprise Player license.

- Open File
- Open from Library

Feature	Provides permission to
TIBCO Spotfire Consumer	Access Spotfire Consumer, which users require to open analyses in the web browser.
External updates of analysis files in Spotfire web clients	Perform updates of analysis files through the Web Player Updater Service API.
Create Collaboration Conversations	Create new conversations within analyses.
View Collaboration Conversations	View conversations within analyses.

Feature	Provides permission to
Create Multiple Windows	View an analysis in multiple windows or browser tabs simultaneously.

Spotfire Diagnostics license

The Diagnostics license provides access to the Application Profiler, which is used for gathering information about analysis files.

Feature	Provides permission to
Application Profiler	Use the Application Profiler tool, which is available in Spotfire Analyst (click **Tools** > **Diagnostics** > **Application Profiler**). This tool is used to scan the Spotfire library and acquire information about the available analyses. You can automatically run tests on a large set of DXP files, and gather information such as how long an analysis takes to load, and how long a visualizations takes to render. You can also use the tool to search for errors.

Spotfire Connectors license

The Connectors license provides permission to add connections to the listed products and services and, except where noted, to analyze data both in-database and in-memory.

Features

- Amazon Redshift
- Apache Drill
- Apache Spark SQL
- Attivio
- Cloudera Hive
- Cloudera Impala
- Dremio
- Google Analytics

  Feature provides permission to analyze data only in-memory.
- Google BigQuery
- Hortonworks
- IBM DB2
- IBM Netezza
- Microsoft SharePoint Online

  Feature provides permission to analyze data only in-memory.
- Microsoft SQL Server
- Microsoft SQL Server Analysis Services
- OData

  Feature provides permission to analyze data only in-memory.
- Oracle

- Oracle Essbase
- Oracle MySQL
- Pivotal Greenplum
- Pivotal HAWQ
- PostgreSQL
- Salesforce

  Feature provides permission to analyze data only in-memory.
- SAP BW
- SAP HANA
- Snowflake
- Teradata
- Teradata Aster
- TIBCO Cloud LiveApps
- TIBCO ComputeDB
- TIBCO Data Virtualization
- TIBCO Spotfire Data Streams
- Vertica

Spotfire Enterprise Player license

The Enterprise Player license contains the functionality for running the Spotfire Enterprise Player. It enables features such as saving files to disk or to the library, opening files from the library, opening linked data, and exporting to various formats. (For technical reasons, some license features under this license are also needed for users running Spotfire Consumer.)

Feature	Provides permission to
Bookmarks	Work with bookmarks.
Capture Private and Public Bookmarks	Work with private and public bookmarks.
Capture Private Bookmarks	Work with private bookmarks.
Change Password	Change one's own password.
Create Collaboration Conversations	Collaborate with others by creating new conversations.
Create Multiple Windows	View an analysis in multiple windows or browser tabs simultaneously.
Details on Demand	Display the Details-on-Demand panel.
Export Data	Export or copy the data used in visualizations.
Export Image	Export visualizations as images.
Export to PDF	Export analysis files to PDF.

Feature	Provides permission to
Export to PowerPoint	Export analysis files to Microsoft PowerPoint.
Find	Allows a user to open the Find tool.
Open File	Open local DXP files. Without this feature, users cannot open any files.
Open from Library	Open files from the Spotfire library.
Open Linked Data	Open analysis files that contain linked data.
Print	Print visualizations to paper.
Save Spotfire Analysis Files	Save analysis files.
Share link	Share links to analyses.
Save to Library	Save files to the Spotfire library.
Share to tibbr®	Share visualizations via tibbr®.
Support Diagnostics and Logging	Access support diagnostics and logging.
Undo/Redo	Undo and redo actions.
View Collaboration Conversations	View collaboration conversations.
Web Page Panel	Use the Web Page panel.

Spotfire Extensions license

The Extensions license provides access to several tools that are available in Spotfire Analyst. It also enables users to author scripts and to use custom extensions.

Feature	Provides permission to
Access to Extensions	View and use custom extensions that developers at your company may add to the Spotfire environment. When new custom license features are deployed on the server, they appear as features under this license. An administrator can then enable or disable each license as necessary.
Author Scripts	Author scripts.  To save scripts and data functions as "trusted" in the library, a user must also be a member of the Script Author group. Only scripts that are saved as "trusted" can be run in the web client, with an exception for simple TERR-based data functions that can be executed in restricted mode. For more information about the Script Author group, see System groups.
Automation Services Job Builder Tool	Access the Automation Services Job Builder tool, which is used to create automated procedures that carry out multi-step tasks

Feature	Provides permission to
Qualification Compare Tool	Access the Qualification Compare tool, which allows users to to verify that a Spotfire analysis gives the same results on different installations of Spotfire. The tool is designed especially for clinical and other regulated industries. An administrator must deploy the Spotfire Qualification package to make the tool available. The Spotfire Qualification license, which is also required, is sold separately.
Qualification Export Tool	Access the Qualification Export tool, which allows users to export an analysis with the Qualification tool. An administrator must deploy the Spotfire Qualification package to make the tool available. The Spotfire Qualification license, which is also required, is sold separately.

Spotfire Information Modeler license

The Information Modeler license provides permissions to set up and edit data sources that are used by information links.

Feature	Provides permission to
Administration	Modify data sources, joins, and other elements when working with information links.  To access this feature, users must also have the Create Information Link feature enabled. This feature is part of the Spotfire Analyst license.
Prompt for Parameter	View the Prompt for Parameter dialog. This feature is intended for people who create and test parameterized information links. When using this feature, a Prompt for Parameter dialog appears for any parameterized information link that is not provided with a value for every parameter.

Creating groups

You can create a group at the top level of the groups hierarchy, or as a subgroup of an existing group. A subgroup inherits access to all the licenses of its parent group or groups. (To import and export groups, use the Administrator Manager in Spotfire Analyst.)

Note that inherited licenses are disabled by default, and will need to be enabled.

For more information, see Groups and licenses.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, click **Create group**.
3. On the Create group page, enter a name for the group.
4. Do one of the following:
   - To create a group at the top level, click **Save**.
   - To create a subgroup, click **Add to groups**, select the check box for each group to which you want to add the new group, and then click **Add**.

What to do next

Set licenses for the group

(Optional) Assign preferences to the group. Preferences are set in the Administration Manager in Spotfire Analyst.

Setting licenses

Administrators assign licenses to groups so that members of that group, and members of that group's sub-groups, have access to the features contained in the license. Each license contains features that can be individually enabled and disabled.

For more information, see How licenses work and License and feature reference.

For basic information, see Groups and licenses introduction.

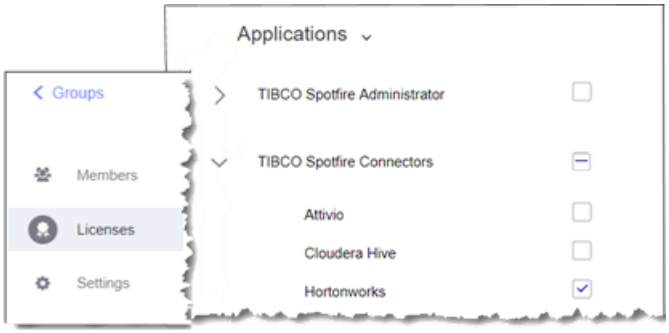
Related commands: set-license, remove-license, show-licenses

Prerequisites

- An administrator has designed the group hierarchy for your Spotfire implementation.
- The group you want to edit is available.
- The Spotfire distribution file, `Spotfire.Dxp.sdn`, is deployed on the server.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to edit, and then click **Edit licenses**.
   A list of all the available licenses is visible on the Licenses page. As indicated in the following example, a dash in the check box for the TIBCO Spotfire Connectors license means than only some of the features under the license are enabled for the group.



   A check mark instead of a dash at the license level means that all of the features under the license are enabled.
3. Select the check boxes for the licenses and features that you want to enable for the group, and then click **Save**.

Adding members to a group

You can add any number of Spotfire users or groups to a group at the same time.

For more information, see Groups and licenses.

For basic information, see Users introduction and Groups and licenses introduction.

Related command: add-member

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group to which you want to add to members, and then click **Edit members**.
   The next page lists the current members of the group.
3. Click **Add** > **Add groups** or **Add** > **Add users**.
4. In the Select dialog, select the check box next to the users or groups that you want to add to the group, and then click **Add**.

Result

The added members are displayed in the list of current members.

Assigning a primary group to a subgroup

When a group has several parent groups, different values may be set for the same license or preference item in two or more parent groups. To ensure that the child group inherits the default settings of a particular parent group, set that group as the primary group.

For more information, see Groups and licenses.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to edit, and then click **Edit settings**.
3. On the Settings page, under **Primary group**, select a primary group.
4. Click **Save**.

Assigning a deployment area to a group

For users to have access to a deployment, you must assign the deployment area that contains the deployment to the appropriate groups. If no deployment area is set for a group, the group members are assigned the default deployment area.

For general information, see Deployments and deployment areas.

Related command: manage-deployment-areas

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to edit, and then click **Edit settings**.
3. On the Settings page, under **Deployment area**, select a deployment area.
4. Click **Save**.

Renaming a group

You can rename only those groups that were added to Spotfire Server after installation. The groups that Spotfire creates automatically, such as Administrator and Script Author, cannot be renamed. Also, externally synchronized groups cannot be renamed in the server.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to edit, and then click **Edit settings**.
3. On the Settings page, under **Group name**, enter the new name.
4. Click **Save**.

Removing members from a group

Removing members from a group removes the members' access to the licenses of that group and its parent groups. Users and groups are removed from a group in the same way.

Externally synchronized groups are managed in that context and not within the Spotfire system.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to edit, and then click **Edit members**.
3. In the members list, right-click the member that you want to remove, and then click **Remove from group**.

   To remove several members at the same time, select each member, right-click one of them, and then click **Remove from group**.

Result

The members that you removed no longer appear in the members list.

Deleting groups from the system

Deleting a group does not delete any of its members from Spotfire; only the group itself is deleted. All users and groups that are members of the deleted group remain in the system. Subgroups that lose their parent group are automatically placed at the top level of the group hierarchy.

Notes: There is no recursive delete function that deletes an entire branch of the hierarchy.

You cannot delete any of the groups that Spotfire creates at installation.

Externally synchronized groups are managed in that context and not within the Spotfire system.

Procedure

1. Log in to Spotfire Server and click **Users & Groups**.
2. On the Groups page, right-click the group that you want to delete, and then click **Delete**.

   To delete several groups at the same time, select each group, right-click one of them, and then click **Delete**.

Result

The deleted groups no longer appear in the groups list.

Deployments and deployment areas

To deploy Spotfire software, the administrator places software packages in a *deployment area* and assigns the deployment area to particular groups.

If a new deployment is available when a user logs in to a Spotfire client, the software packages are downloaded from the server to the client.

Deployments are used:

- To set up a new Spotfire system.
- To install a product upgrade, extension, or hotfix provided by Spotfire.
- To install a custom tool or extension.

A group of software packages (.spk files) can be bundled together into a *distribution* (.sdn file). A distribution can be copied to create a new deployment area, or downloaded for deployment to another Spotfire Server.

Every user is associated with at least one deployment area; by default, this is the Production area that is created when you install Spotfire Server, but you can designate any area as the default.

Some users have access to more than one deployment area because they belong to several groups that are associated with different deployment areas. In this case, users are prompted to choose a deployment area when they log in to the Spotfire client.

Whether a user has access to a particular feature contained in a distribution depends on the licenses that are assigned to that user's groups. For more information, see Groups and licenses introduction.

Administrators usually create a Test deployment area to use as a staging server; when the new software has been thoroughly tested in their Spotfire environment, the distribution is copied to a production area.

Creating a new deployment area

Deployment areas contain software packages that you make available to certain groups. You can create a new deployment area for a Spotfire update or extension, for custom tools created in your organization, and so on.

For general information, see Deployments and deployment areas.

Procedure

1. Log in to Spotfire Server. (For instructions on accessing the server, see Opening Spotfire Server.)
2. Click **Deployments & Packages**.
3. In the **Deployment areas** pane, click **Add**.
4. In the **Add area** dialog, enter a name for the new area.

   Deployment area names are case insensitive and have a maximum length of 25 characters. These are the valid characters:
   - a - z
   - 0 - 9
   - The underline character _
   - The dash character -
5. Click **Add area**.

Result

The new deployment area is displayed in the **Deployment areas** list.

Adding software packages to a deployment area

When Spotfire releases updates, or if your company creates custom tools or other software elements, the administrator adds these to a deployment area so that they can be uploaded to Spotfire Server. Then the server distributes the new software to the appropriate groups, as selected by the administrator.

For general information, see Deployments and deployment areas.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the left pane, under **Deployment areas**, select a deployment area.

   It is recommended that you first test the software on a deployment area that is not in production.
3. Optional: If the deployment area contains any software packages that are not currently needed, delete them. (For instructions, see Removing packages from a deployment area.)
4. In the "Software packages" pane, click **Add packages**.
5. In the "Add packages" dialog, click **Choose File**, locate and select the file you want to add, and click **Open**.
6. In the "Add packages" dialog, click **Upload**.
   The added packages are displayed in the "Software packages" pane.

   If you want to start over again, you can return to the last saved version of the deployment area by clicking **Revert all**.
7. To confirm that the packages are error-free, in the "Software packages" pane click **Validate area**.
8. To save the new packages, click **Save area**.
9. In the "Save deployment" dialog, if you want the Spotfire clients to automatically accept the update when they are opened (rather than having the user decide when to accept the update), select the **Force client update** check box.
10. Click **Save area**.

Copying a distribution to another deployment area

You can copy a distribution from one deployment area to another when you are ready to move it from a test area to a production area, or if you want to create a new deployment based on an existing one.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. Under **Deployment areas**, select the deployment area that contains the distribution you want to copy.
3. In the Information pane to the right, click **Copy distribution**.
4. In the "Copy distribution" dialog, do one of the following:
   - Select the existing deployment area to which you want to add the distribution, and then click **Copy**.
   - Create a new deployment area to hold the distribution by clicking the **To new area** tab, entering a name for the area, and clicking **Copy**.

Result

When you select the deployment area in the "Deployment areas" pane, the copied software packages are displayed under **Software packages**.

Exporting a distribution

You can download a local copy of a distribution (.sdn file) for deployment to another Spotfire Server.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. Under **Deployment areas**, select the area that contains the distribution that you want to export.
3. In the Information pane to the right, click **Export distribution**.

Changing the default deployment area

The default deployment area is available to all groups for which no deployment area has been set. During installation, Spotfire Server adds a "Production" deployment area and sets it as the default, but you can change the default area to give users access to new software packages.

For general information, see Deployments and deployment areas.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the "Deployment areas" pane, select the deployment area you want to set as the default.
3. In the upper-right pane, click **Make default**.

Renaming a deployment area

You can rename any deployment area in your system.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the "Deployment areas" pane, select the deployment area you want to rename.
3. In the Information pane to the right, click **Rename**.
4. In the "Rename deployment area" dialog, enter a new name.

Deployment area names are case insensitive and have a maximum length of 25 characters. These are the valid characters:

- a-z
- 0-9
- The underline character _
- The dash character -

5. Click **Rename**.

Removing packages from a deployment area

You can edit the contents of any of your deployment areas.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the "Deployment areas" pane, select the deployment area from which you want to remove packages.
3. In the "Software packages" pane, select the check boxes for the packages you want to remove, and then click **Remove packages**.
4. Click **Validate area**, and when the area is validated, click **Save area**.

Clearing all packages from a deployment area

If you want to create a new deployment in an existing deployment area, you can clear the area of its contents.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the "Deployment areas" pane, select the deployment area that you want to clear.
3. In the "Software packages" pane, select all the packages in the area.
4. Click **Remove packages**.
5. Click **Validate area**, and when the area is validated, click **Save area**.

Deleting a deployment area

You can delete a deployment area that is no longer needed. The software packages in that area will be removed as well.

Procedure

1. Log in to Spotfire Server and click **Deployments & Packages**.
2. In the "Deployment areas " pane, select the check box in front of the deployment area you want to delete.

   It is not possible to delete the area that is set as the default deployment area.
3. In the "Deployment areas " pane, click **Delete**.

Scheduled updates to analyses

For analyses that contain links to large amounts of data, downloading fresh data can take a significant amount of time. Scheduled updates save time by downloading the latest data before users need it.

Based on settings in Spotfire Server, or on messages that the server receives from an external source, selected analyses can be preloaded with fresh data, stored on specific Spotfire Web Player instances, and then made available to users as needed.

For example, in the case of sales data that is tallied at the end of the day, you could schedule the update to occur overnight so that users can quickly access the analysis first thing in the morning, when they log in. Or, in the case of a large analysis that users tend to refer to several times during the day, you could schedule an update every 20 minutes.

You can trigger updates in two ways:

- In Spotfire Server you can create rules that specify the analysis to preload, when to do it, whether the new data is automatically displayed to the end user, and so on.
- Using TIBCO Enterprise Message Service™ (EMS) or a web service, you can create "event-driven updates" that are triggered by an external process. For more information about event-driven updates, see Creating a scheduled update by using TIBCO EMS or, to use a web service, consult the Web Services API documentation.

When scheduling an update in Spotfire Server, you can configure the following options:

- The days of the week that the update runs.
- The times of day between which the updated analysis is available to end users.
- How often the server checks for new data.
- The *resource pool* on which to preload the analysis, and the number of Spotfire Web Player instances that should be available for users opening the analysis.
- Whether the updated data is automatically displayed in the user's copy of the analysis, or the user decides when to refresh the information.
- Whether to allow cached and pre-computed data when the analysis is reopened.

On the Overview page, the "Scheduled updates" pane gives you the basic status of your scheduled updates.

In the **Rules** list you can identify scheduled updates (as opposed to *routing rules*) by their **Type** (**File**) and the fact that a schedule is displayed under **Schedule** in the list.

You can also view the Activity and Notifications pages in Scheduling & Routing to monitor job status.

Creating scheduled updates by using Spotfire Server

In Spotfire Server, you can configure and run automated data updates to existing analysis files. This saves time for end users because they do not have to wait for the new data to download when they open the analysis.

Prerequisites

- The analysis file to be updated must be in the Spotfire library.
- The scheduled updates user service account (scheduledupdates@SPOTFIRESYSTEM) must have the following library permissions:
  - **Browse & Access** permissions to the analysis.
  - Permissions to access the folder(s) that hold the information link object.
  - Permission to access the data source object.

  To set library permissions, use the tools in Spotfire Analyst.

  Alternatively, you can use the copy-library-permissions command to copy library permissions from another user or group.

The following tasks are optional, but you may want to complete them before creating the scheduled update:

- If you want this update to run according to a schedule (or several schedules) that you plan to reuse, create the schedules first; for instructions, see Creating schedules.
- If you want the updated file to open on specific instances of the Spotfire Web Player, create a *resource pool* containing those instances; for instructions, see Creating resource pools.

If you are creating a scheduled update for an analysis that is based on data from a prompted or personalized information link, see Scheduled updates with prompted or personalized information links.

For general information, see Scheduled updates to analyses.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. In the Rules pane, click **Create rule**.
3. Under **Type**, select **File**, and then click **Next**.
4. Enter a name for the rule and select the file that you want to update.
5. Under **Select resource pool**, do one of the following:
   - If you do not want to set a specific resource pool on which to open the analysis, leave the **System Default** routing selected.
   - If you want the analysis to open on a specific resource pool, select it.

     If a scheduled update rule indicates that a file should open on a specific resource pool, this rule overrides any routing rules (for a group or an individual user) that specify a different resource pool for the user who opens the updated file.
6. Optional: Set a priority. This setting comes into effect if two or more scheduled updates are scheduled to occur at the same time. **0** is the highest priority.
7. To set a schedule, do one of the following:
   - To update the analysis based on a schedule that has already been created or several schedules, select **Use saved schedule** and then, in the "Select schedule" dialog, select the schedule or schedules that you want to use.
   - To create a "unique schedule" for this rule (a schedule that will not be available for reuse), select **Use custom schedule**. For instructions on setting up the schedule, see Creating a reusable schedule.

     Analyses are always updated and loaded at the beginning of each scheduled start time, in addition to the reloads that are set in the **Check for updates every** field. If a scheduled update is scheduled for 24 hours a day/7 days a week, with **Check for updates every** set to 0, the analysis is loaded only once, when the rule is initially executed.
8. If you want the rule to be disabled initially, select the **Disable rule** check box in the bottom right of the dialog. You can enable the rule later, on the Scheduling & Routing page.
9. Optional: If you want to do one of the following, click **Additional properties**:
   - Set the number of Spotfire Web Player instances for this rule.
   - Switch the client update method from automatic to manual.
   - Disallow cached and pre-computed data.

   For details, see Additional settings for scheduled updates.
10. In the "Create rule" dialog, click **Save**.

    If you are unable to save the information you entered, and your library files are stored externally on Amazon Web Services S3 (AWS), see Forcing Java to use IPv4.

Result

The rule is displayed in the **Rules** list.

Additional settings for scheduled updates

In addition to basic information about the analysis that you want to update and when you want the update to occur, several additional property settings are available in Spotfire Server.

Setting the number of Spotfire Web Player instances to make available for a scheduled update

By default Spotfire Server uses one of the available Spotfire Web Player instances when users open a scheduled update file. To load balance or to change the resource load of a particular analysis, the administrator can set the number of instances on which the updated analysis can open.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. Do one of the following:
   - If you want to change this property for an existing scheduled update, under **Rules** select the update and click **Edit**.
   - If you are creating a new scheduled update, at the bottom of the second Create rule dialog, click **Additional properties**.
3. In the Additional properties dialog, under **Number of instances** select a number.
4. Click **Update** and then **Save**.

Switching the scheduled update method from automatic to manual

When the scheduled update method is set to manual, users decide when to incorporate new data in the analysis.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. Do one of the following:
   - If you want to set this property for an existing scheduled update, under **Rules** select the check box next to the update rule and click **Edit**.
   - If you are creating a new scheduled update, at the bottom of the second Create rule dialog, click **Additional properties**.
3. In the Additional properties dialog, under **Update method**, indicate how users should receive the updated data:
   - **Automatic**—The new data is automatically displayed in the analysis when a user opens it.
   - **Manual**—A Refresh icon on the title bar of the analysis indicates that an updated version is available. When the user clicks the icon, the analysis is updated.
4. Click **Update** and then **Save**.

Disallowing cached and precomputed data in individual scheduled update files

If your Spotfire environment is set up to use disk caching and precomputations of data to shorten the time it takes for an updated analysis to reopen in a Spotfire Web Player after the analysis closes, this setting may prevent the latest data from appearing in the reopened analysis. You can turn this setting off for individual scheduled update files.

By default, cached and precomputed data is *not* enabled. To enable this feature, see Enabling cached and precomputed data for scheduled update files.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. Do one of the following:
   - If you want to change these properties for an existing scheduled update, under **Rules** select the update, click **Edit**, and then click **Additional Properties**.
   - If you are creating a new scheduled update, at the bottom of the second Create rule dialog, click **Additional Properties**.
3. In the Additional properties dialog, under **Caching**, clear the check boxes of the settings you want to turn off.
4. Click **Update** and then **Save**.

Result

The analysis will always reflect the latest data but it may reopen more slowly.

Scheduled updates with prompted or personalized information links

Scheduled updates are intended mainly for use with analyses that were set up using ordinary information links to load data. If you set up scheduled updates for an analysis that is based on data from a prompted or personalized information link, there are special issues to consider.

When a user opens an analysis that is based on a prompted information link, the user selects a certain view of the data to be loaded. In the same way, when a user opens an analysis that is based on a personalized information link, the data loaded is determined by the permissions of the user who logs in.

However, when a scheduled update of this file occurs, the update causes the analysis to reload based on the prompted values that were specified when the file was originally saved, and the permissions of the user that the administrator set up to programmatically run the scheduled update. This means that users with an analysis already open will see a different selection of data the next time that they update the analysis because the scheduled update has in fact updated the underlying data on the server.

You should be especially careful when setting up scheduled updates for analyses with personalized information links. If the user you specify for the scheduled updates has access to more data than the intended end users of the analyses, these end users may see more data than they have access to; they will see all the data that is available to the user specified for scheduled updates.

Editing scheduled updates

You can edit most properties of a scheduled update at any time. To change the analysis file or the resource pool in a scheduled update, however, you must first disable the rule.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. In the **Rules** pane, select the scheduled update that you want to edit.
3. Optional: If you want to change the rule's analysis file or resource pool, click **Disable**.
4. In the **Rules** pane, click **Edit** and make your changes.
5. Click **Save**.

6. Optional: If you disabled the rule in step 3, click **Enable** to make it active again.

Creating schedules

You can create and save schedules that you plan to reuse in scheduled updates to analyses. If a schedule will only be used once, you can set it when you create the update rule.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. In the "Saved schedules" pane, click **Create schedule**.
3. In the "Create schedule" dialog, enter a name for the schedule.
4. Under **Repeat**, indicate the days on which you want the update to run by selecting the appropriate check boxes.
5. Under **Start** and **End**, enter the times between which the updated analysis should be available to end users (on the days that you indicated in the previous step).
6. Under **Time zone**, select the time zone for the times that you entered in the previous step.
7. Under **Check for updates every**, select how often you want Spotfire Server to check whether the analysis file or its underlying data has changed. If the analysis or data has changed, the server updates the pre-loaded file.

   Analyses are always updated and loaded at the beginning of each scheduled start time, in addition to the reloads that are set in the **Check for updates every** field. If a scheduled update is scheduled for 24 hours a day/7 days a week, with **Check for updates every** set to 0, the analysis is loaded only once, when the rule is initially executed.
8. Click **Save**.

Result

The new schedule is displayed in the **Saved schedules** list.

Manually updating a file outside of its update schedule

If you do not want to wait for a file to be updated according to its schedule, you can trigger an update manually.

Prerequisites

There is a scheduled update for the file that you want to manually update.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. On the **Overview** page, under **Rules**, select the file.
3. Click **Reload**.

Copying routing rules and schedules from one site to another

You can copy all the routing rules and saved schedules from one site in your Spotfire environment to another site in the same environment by using the `copy-rules-to-site` command. This is helpful when setting up local access points for users who are located in different regions.

This procedure copies rules that were created in the Spotfire Server administration interface. Scheduled updates that are triggered externally, for example by TIBCO Enterprise Message Service (EMS), are not copied.

Procedure

1. Open a command line as an administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
2. On the command line, enter the **`copy-rules-to-site`** command, specifying the options needed. Example:

```
config copy-rules-to-site --source-site-name=NewYork --target-site-name=SanFran rule-conflict-resolution=replace --use-default-resource-pool=true --disabled=false --test-run=false
```

   For information on the command options, see copy-rules-to-site.

Result

In this example, the rules and saved schedules from the NewYork site are reproduced in the SanFran site. On the computer where you ran the command, the `impex.rules.log` file, which provides information about the copy process, is available in the following directory: `<installation dir>/tomcat/logs`.

Exporting routing rules and schedules for import in a different Spotfire environment

You can export the routing rules and saved schedules, including scheduled Automation Services jobs, from a Spotfire Server to a JSON file. Then, to prepare for a rolling update or to test and validate a new version of Spotfire, you can import the JSON file on a different Spotfire environment.

This procedure exports rules that were created in the Spotfire Server administration interface. Scheduled updates that are triggered externally, for example by TIBCO Enterprise Message Service (EMS), are not exported. Likewise, scheduled Automation Services jobs that were not created in the administration interface are not included.

Procedure

1. Open a command line as an administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
2. On the command line, enter the **`export-rules`** command, specifying the options needed to export the data to a JSON file.
   Example:

```
config export-rules --bootstrap-config="C:\Work\Spotfire\bootstrap.xml" --tool-password=Spotfire --keystore-file "C:\Work\nm\trust\keystore.p12" --force
```

   For information on the command options, see export-rules.

Result

In this example, the `rules.json` file containing your scheduled updates, Automation Services jobs, and routing rules is available in the `<server installation dir>/tomcat/spotfire-bin` directory.

Importing routing rules and schedules from a different Spotfire environment

After you have exported the routing rules and saved schedules from a Spotfire Server to a JSON file, you can import the JSON file in a different Spotfire environment to prepare for a rolling update, for example, or to test and validate a new version of Spotfire.

Prerequisites

- You have exported the rules and schedules from the original server to a JSON file; for instructions, see Exporting routing rules and schedules.
- At least one server in the target environment is running.
- The analysis files referred to in the rules have been added to the target environment.
- The Automation Services jobs referred to in the scheduled jobs have been added to the target environment.
- The users and groups referred to in the rules have been created in the target environment.
- If you want the target environment to use resource pools that are named the same as the resource pools in the original environment, and you want the import to use the same resource pool assignments as the original environment, create the resource pools before importing the file.

Your other options are to assign the imported rules to the default resource pool, or to another resource pool; for details, see the **`-r`** option and the **`-u`** option in import-rules.

Procedure

1. Open a command line as an administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
2. On the command line, enter the **`import-rules`** command, specifying the options needed. Example:

```
config import-rules --bootstrap-config="C:\Work\Spotfire\bootstrap.xml" --keystore-
file="C:\Work\nm\trust\keystore.p12" --rule-conflict-resolution=replace --schedule-
conflict-resolution=rename --use-default-resource-pool=true --test-run=false
```

   For information on the command options, see import-rules.

If you want the chance to address import errors up front, you can enable the **`--test-run`** option. This option provides a preview of any import errors before the actual import takes place.

Result

In the previous example, the rules and saved schedules are imported and assigned to the default resource pool. On the server where you ran the command, the `impex.rules.log` file, which provides information about your import, is available in the following directory: `<installation dir>/tomcat/logs`.

Disabling or deleting scheduled updates and routing rules

Disabling a scheduled update or other rule makes the rule inactive until you activate it again. Deleting a rule removes it from the database.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. Select the check box next to the rule or rules that you want to disable or delete.
3. Click **Disable** or **Delete**.
   If you disabled a rule, it appears grayed out in the list.

Deleting schedules

Deleting a schedule removes it from the database and cancels any scheduled updates that use the schedule.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. Select the check box next to the schedule or schedules that you want to delete.
3. Click **Delete**.

If deleting the schedule will cancel any scheduled updates, Spotfire Server lists the affected rules.

Creating scheduled updates by using TIBCO EMS

You can create scheduled updates that are triggered by messages from TIBCO Enterprise Message Service (EMS). In Spotfire Server, the external updates configuration takes place in the server, and the updates are sent to the server. Spotfire Server then sends the updates to the appropriate web player service(s).

Prerequisites

- EMS is installed on a computer.
- The following files, which are located in your TIBCO EMS installation in the `lib` folder, must be copied to the Spotfire Server classpath on the server computer. If your implementation is clustered, the files must be copied to each computer in the cluster. If your implementation includes sites, the files must be copied to each server in the sites that will receive scheduled updates via EMS.
  - `jms.jar` or `jms-2.0.jar` (depending on the version)
  - `tlbjms.jar`
  - `tibcrypt.jar`

Procedure

1. On the Spotfire Server command line, use the config-external-scheduled-updates command to configure the server to accept the EMS messages. (For details on using the Spotfire command line, see Executing commands on the command line.) Include the following parameters:

   - Set the `ems-enabled` value to "true".
   - Set the server and port to the computer and port on which EMS is currently running. Use this configuration:

     ```
     <server-url>tcp://localhost:7222,tcp://localhost:7222</server-url>
     ```

     This enables the reconnect parameters. For more information about this value, see "Fault Tolerance" in the TIBCO EMS documentation.
   - Set the `client-id` value to indicate which server or site will handle the scheduled updates:
     - If your Spotfire implementation includes a clustered server deployment (but not sites), set the `client-id` to a unique value in the cluster. In this case, the first server to connect to EMS will handle all the scheduled updates received via EMS.
     - If your Spotfire implementation includes sites, each site that will receive scheduled updates via EMS must have its own client-id.

   Command example

   ```
   config config-external-scheduled-updates -e true -s tcp://localhost:7222 -i clientId1 -t
    scheduled_updates -S "first site"
   ```

   Example of the resulting section in the server configuration file (`configuration.xml`):

   ```
   </external-updates>
       <external-updates site="first site" operation="override">
         <ems-enabled>true</ems-enabled>
         <server-url>tcp://localhost:7222</server-url>
         <client-id>clientId1</client-id>
         <topic>scheduled_updates</topic>
         <reconnect-attempt-count>10</reconnect-attempt-count>
         <reconnect-attempt-delay-milliseconds>1000</reconnect-attempt-delay-milliseconds>
         <reconnect-attempt-timeout-milliseconds>1000</reconnect-attempt-timeout-
   milliseconds>
         <keep-alive-minutes>10</keep-alive-minutes>
       </external-updates>
   ```

2. In EMS, create the message. Include the following parameters:

   - Path (required)
   - ClientUpdate
   - KeepAliveMinutes
   - ResourcePoolName

   If the following statements are true, the resource pool value in the existing rule takes precedence:
   - There is an existing rule for the same file.
   - The existing rule was created in Spotfire Server.
   - The existing rule specifies a resource pool.
   - The existing rule is enabled.

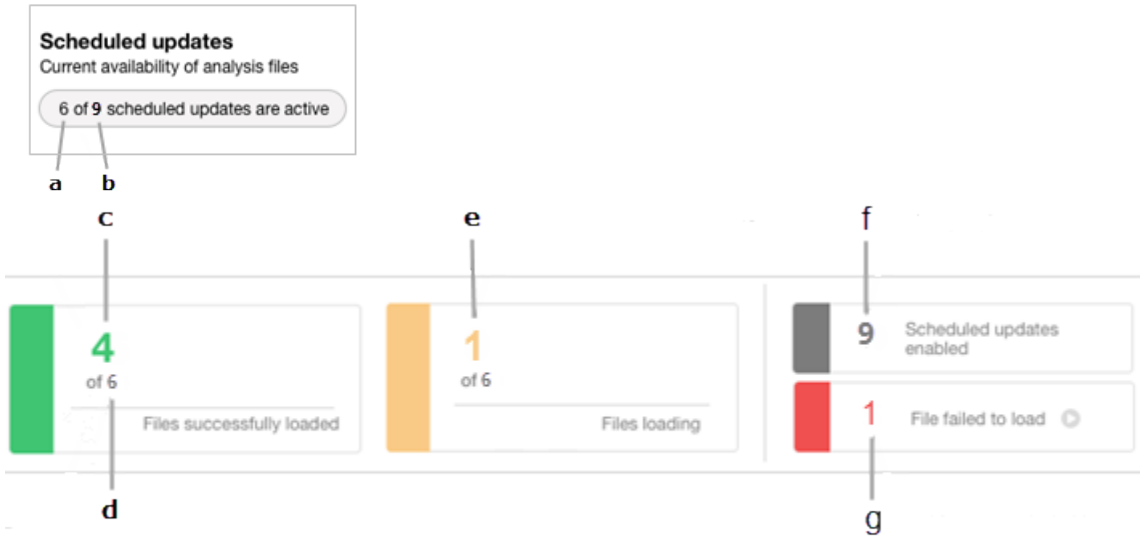
   For the `ClientUpdate` parameter, the value (`manual` or `automatic`) that is defined in the external rule takes precedence. If the external update does not specify a value, or if the specified value is invalid, the value from an enabled rule is used, if available.

3. Send the EMS request. For details, see the TIBCO EMS documentation.

Scheduled updates monitoring

The Scheduling & Routing area of Spotfire Server provides several ways of monitoring the success of your scheduled updates.

The "Scheduled updates" pane at the top of the Overview page summarizes the current state of your scheduled updates:



Details about the Scheduled updates summary

a	**Number of active scheduled updates** The number of scheduled update rules that are enabled and currently within their schedule window. This means that the files that are attached to these rules are scheduled to be loaded now, so that end users can view them without waiting for the data to download.
b	**Number of enabled rules** The total number of file rules that are enabled in your Spotfire implementation. This includes file rules without schedules.
c	**Number of scheduled update rules that ran successfully** The number of scheduled update files that end users can currently view without waiting for the data to download. These analyses have been updated (if new data was available) and loaded on at least one Spotfire Web Player instance. This does not guarantee that the file was loaded on the number of Spotfire Web Player instances that is specified in the rule.
d	The same as **a**.
e	**Number of scheduled update files that are currently being loaded** The number of scheduled update files that are currently being loaded and so not yet available to end users. Scheduled update files that are waiting to load are not counted.
f	The same as **b**.

g	**Number of failed scheduled updates** The number of unsuccessful scheduled updates. (The analysis files attached to these rules should have been updated and loaded on at least one Spotfire Web Player instance.) After a scheduled update fails, it is included in this number until it is scheduled to load again, or until it is manually reloaded.

You can click the large boxes in the Scheduled updates pane to view the scheduled update rules that each box refers to.

On the Activity page, you can view the status, date, and time of each file update attempt. Click the arrow to the left of the line to view additional details, any messages that were generated, and a link to relevant logs.

Important messages are listed on the Notifications page. An information symbol on the **Notifications** tab, and on the Scheduling & Routing image on the main server page, indicates that there is a new notification.

Changing the priority of a rule

Spotfire Server uses rule priorities if two or more rules are executed at the same time.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
   On the **Overview** page, under **Rules**, the scheduled updates and routing rules are listed in priority order.
2. Select the rule whose priority you want to change and then do one of the following:
   - Drag the rule to a new position in the list.
   - On the right end of the row, click the **More** menu (**...**) and then select **Move to top** or **Move to bottom**.
   - Click **Edit** and then, in the " Edit rule" dialog, enter a new priority number under **Set a priority**.

Changing the number of retries for failed scheduled updates

By default, Spotfire Server retries a failed scheduled update ten times. Using the command-line interface, you can set a different limit for the number of times that a scheduled update is retried if it initially fails.

This property was previously set by using the `stopUpdatesAfterRepeatedFail` setting in the `Spotfire.Dxp.Worker.Web.config` file.

Procedure

1. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. Use the config-scheduled-updates-retries command to set the retry limit.
   Example:

```
config config-scheduled-updates-retries --stop-updates-after-repeated-fail-enabled=true
 --fails-before-stop=X
```

   where X is the number of times to retry the update.

3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server service.

Changing how often the scheduled job history is cleared

If your organization runs many scheduled updates or scheduled Automation Services jobs, history records can quickly pile up in the database. Spotfire Server automatically purges the history after three days, but you can change how often this occurs by editing the `configuration.xml` file.

Procedure

1. Export and open the Spotfire Server configuration file; for general instructions, see Manually editing the Spotfire Server configuration file.
2. Do one of the following:
   - If you are editing a Spotfire Server 7.5 or later configuration file, change the number "3" (which indicates 3 days) in the following section:

     ```
     <scheduled-jobs>
       <!-- All the job executions older than the specified number of days will be
      automatically deleted. Default: 3 days-->
         <purge-history-older-than>3</purge-history-older-than>
         <!--Intervals in hours, Default: 12 hours -->
         <purge-history-interval>12</purge-history-interval>
       </scheduled-jobs>
     ```

   - If you are updating an existing configuration file from a previous version of Spotfire Server, add the entire `<scheduled-jobs>` section to the file and then change the number of days between history purges.
3. Save the configuration file and import it back to the server; for instructions, see Manually editing the Spotfire Server configuration file.

Common analysis loading errors

The following are the most common error codes and messages that are displayed when an analysis file does not load successfully.

- SPOT-10001 FileCorruptMissingRequiredEntry

  Server was unable to read the uploaded file because it is not a valid DXP file.
- SPOT-10002 IncompatibleVersion

  Unsupported file version.
- SPOT-10003 FileCorrupt

  Server was unable to read this file because it is not a valid DXP file.
- SPOT-10004 IncompatibleDevelopmentVersion

  Server was unable to read the file. The file was saved with a development version of Spotfire and contains features that are not supported by this version.
- SPOT-20000 LoadFileUnknownError

  Server was unable to read the file.
- SPOT-20001 IOException

  An I/O error occurred when the server attempted to open the file.

- SPOT-30001 LoadFileNoPermissions

  Server was denied access to the file.
- SPOT-40001 LoadFileOutOfMemory

  Server was unable to load the file due to insufficient memory.
- SPOT-50001 LibraryFailedLoad

  Server could not load the analysis.
- SPOT-70001 FailedToExecuteDataSource

  Server was unable to execute the data query.
- SPOT-70002 CouldNotCreateDatabaseConnection

  Server was unable to access one or more data sources.
- SPOT-70003 FailedToOpenInformationLink

  Server was unable to load the information link.
- SPOT-100000 UnknownError

  Server was unable to read the file.

Routing rules

A routing rule specifies the *resource pool* on which an analysis opens. You can create routing rules to set a *resource pool* on which to open analyses that are requested by members of a specific group, or by a specific user. You can also set a resource pool for a specific analysis, regardless of who requests it.

You can use routing rules to fine-tune resource management, but their use is optional.

Specific reasons for creating routing rules include the following:

- Define an exclusive resource pool for a critical analysis so that it can be updated and viewed without interference from other analyses and user requests.
- Define a resource pool for management so that they can view and work with analyses without waiting.
- Define a resource pool for users who are trying out a new version of Spotfire.
- Load an analysis on several Spotfire Web Player instances to handle a large number of users.

The default routing rule

The default routing rule indicates the resource pools on which all analyses are opened, unless the analysis itself, or the user who is requesting it, is subject to another routing rule. By default, the default routing rule includes all the services and instances that are available in your Spotfire implementation.

You can edit default routing to include only certain services and instances, but the rule cannot be deleted.

The default routing rule is always displayed at the bottom of the **Rules** list on the **Scheduling & Routing** page.

Creating routing rules

You can create routing rules that apply to user groups, individual users, or specific analysis files.

Prerequisites

- Create the resource pool that you want to specify for the rule; see Creating a resource pool.
- If you are creating a rule for an analysis file, the file must be in the Spotfire library.

For general information, see Routing rules.

Procedure

1. Log in to Spotfire Server and click **Scheduling & Routing**.
2. In the **Rules** pane, click **Create rule**.
   The Create rule dialog opens.
3. Under **Type**, do one of the following and then click **Next**:
   - If you want to set a *resource pool* on which to open analyses that are requested by members of a specific group, select **Group**.
   - If you want to set a resource pool on which to open analyses that are requested by an individual user, select **User**.
   - If you want to set a resource pool on which to open a specific analysis file, select **File**.
4. Enter a name for the rule and then do one of the following:
   - Select the group to which the rule applies.
   - Select the user to which the rule applies.
   - Select the file to which the rule applies.
5. Under **Select resource pool**, select the resource pool on which the analyses that are affected by this rule should open.

   If a scheduled update rule indicates that a file should open on a specific resource pool, that rule overrides any routing rules (for a group or an individual user) that specify a different resource pool for the user who opens the updated file.
6. Optional: Set a priority. This setting comes into effect if two or more rules occur at the same time. **0** is the highest priority.
7. If you want the rule to be disabled initially, select the **Disable rule** check box in the bottom right of the dialog. You can enable the rule later on the Scheduling & Routing page.
8. Click **Save**.

Result

The rule is displayed in the **Rules** list.

Monitoring and diagnostics

Spotfire Server provides a wide range of information to help you manage and troubleshoot your implementation.

Server and node logging levels

To help locate and respond to issues that can arise in your Spotfire implementation, you can easily change the amount and types of server and node logs that Spotfire Server collects, without leaving the administration interface .

Spotfire Server provides four logging templates that correspond to the most common logging requirements. Each server and node in your implementation can be set to one of these logging levels.

Logging level	Description
Standard (default)	This logging level captures information-level data about runtime events. The `log4j2.xml` file controls this logging level.
Debug	This level captures detailed debugging information as well as warnings, errors, and other details in the `server.log`. The `sql.log` captures detailed SQL Server information. If the server is started from a command prompt or shell, the output to the command prompt or shell is included in the server log. The `logging-debug.properties` template controls this logging level.
Minimal	This level captures basic information about errors and warnings. The `logging-minimal.properties` template controls this logging level.
Trace	This level captures more detailed information than the debug level. Because this logging level is very comprehensive, it should be used carefully. The `logging-trace.properties` template controls this logging level.
Custom	This level is used by Spotfire support. It makes it possible to upload customized logging templates.

Administrators are strongly advised to use the included logging templates. Do not modify or delete these templates.

For a list of logging levels for services, see Service log levels.

Changing server and node logging levels

When Spotfire Server alerts you to an issue in your implementation, you can switch to a more complete logging level from within the Monitoring & Diagnostics area of the administration interface .

Prerequisites

You must have administrative credentials for Spotfire Server.

Alternatively, you can change logging levels by using the set-logging command. For information on using the command line, see Executing commands on the command line.

It is a good practice to back up the existing logs and clear the logs folder before capturing the debug logs.

Procedure

1. Log in to Spotfire Server and click **Monitoring & Diagnostics**.
2. On the Overview page, select the server(s) or the node(s) whose logging level you want to change.
3. Click **Set log configuration**, select a different logging level, and click **Set**.

Result

The changes appear in the **Log configuration** column.

When the troubleshooting is completed, you should switch back to a lower logging level. You can return quickly to the **Standard (default)** level by selecting the server or node and clicking **Reset log configuration**.

Changing the logging level for a server or node that is not running

When a server or node is not running, you can increase its logging level to capture more troubleshooting data.

Best practice is to back up the existing logs and clear the logs folder before capturing the debug logs.

Procedure

1. On the computer that is hosting the server or node whose logging level you want to change, navigate to the directory that contains the logging templates.
   - For a Spotfire Server, the default location is *`server installation dir`*`\tomcat\spotfire-config`.
   - For a node manager, the default location is *`node manager installation dir`*`\nm\config\log-config`.
2. If the directory already contains a `logging-levels.properties` file, delete it.
3. Make a copy of the logging template that you want to apply to the server or node, and name the copy "logging-levels.properties". The available templates are `logging-debug.properties`, `logging-minimal.properties`, and `logging-trace.properties`.
4. Open the `logging-levels.properties` file in a text editor or an XML editor.
5. In the `logging-levels.properties` file, add the following line of code to the top of the file, replacing *template name* with the name of the template file that you copied:

   ```
   ActiveConfig=template name
   ```

   Example

   ```
   ActiveConfig=logging-debug.properties
   ```
6. Save and close the `logging-levels.properties` file.
7. Restart the Spotfire Server service or the Spotfire Node Manager service for the changes to take effect.

Result

The server or node captures the logging information that is indicated in the `logging-levels.properties` file.

When the troubleshooting is completed, switch back to a lower logging level.

Switching back to the Standard (default) logging level

After troubleshooting an issue, you can quickly return to the Standard (default) logging level.

Procedure

1. Log in to Spotfire Server and click **Monitoring & Diagnostics**.
2. On the Overview page, select the server(s) or the node(s) whose logging level you want to return to the default.
3. Click **Reset log configuration**.

Result

The changes appear in the Log configuration column.

Accessing Spotfire Server and node logs

You can view and download various types of Spotfire Server and node logs.

For more information about available logs, see the following topics.

- Spotfire Server logs
- Node logs
- Web Player service logs

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. On the **Overview** page, under **Spotfire Servers** or **Nodes**, locate the server or node for which you want to access logs, and click **View logs**.
   The Logs page opens.
3. In the **Select log file to view** drop-down list, select the log file you want to view.
   The selected log file is shown in the "View logs" pane.

   You can export the log file by clicking **Download full log file**.

Spotfire Server logs

The server logs store important diagnostic information about the Spotfire Server. The information can help in troubleshooting and resolving issues.

The Spotfire Server runs by default at a basic logging level. This can be elevated when needed; for instructions, see Changing server and node logging levels.

The most important log is the `server.log`. This log file stores information about all activities on the server and can be very handy in troubleshooting issues.

If you encounter an issue with Spotfire Server, provide the server logs to Spotfire Support when you enter the support request.

The following log files are available.

Log file	Description
`access.log`	Provides information about client access and access attempts to the server and files in the library.
`actionslogs\actionlog.log`	Provides information about user actions.
`catalina.<date>.log`	An Apache Tomcat log file.
`commons-daemon.<date>.log`	An Apache Tomcat procrun log. See https://commons.apache.org/proper/commons-daemon/procrun.html for more information.
`impex.log`	Provides information about Spotfire library imports and exports.
`impex.rules.log`	Provides information about importing, exporting, and copying scheduled updates and routing rules between computers running Spotfire Server.
`isusage.log`	Provides information about Information Services usage.
`library.log`	Provides information about Spotfire Library usage.

Log file	Description
`localhost.<date>.log`	An Apache Tomcat log file.
`performance.monitoring.log`	Spotfire Server performance metrics.
`s3request.log`	Provides information about Amazon S3 storage.
`server-diagnostics.log`	Provides diagnostic information about server properties.
`server.log`	Provides information about all activity on the server except those events recorded in `access.log`.
`sessions.log`	Provides information about new sessions and their originating IP-address and user-agent.
`soap.log`	Provides information about SOAP communication.
`sql.log`	Provides information about executed SQL queries performed when an information link is executed.
`startup.log`	Provides information about `JAR` files loaded on server startup.
`tools.log`	Information about activity in the configuration tool and on the command line. If you run any configuration commands at the command prompt or use the administration console, this is the log that captures that information.
`tss`*`version`*`-stderr.<date>.log`	An Apache Tomcat log file.
`tss`*`version`*`-stdout.<date>.log`	An Apache Tomcat log file.
`usage.log`	Provides information about client access and access attempts to the server.
`user-interface.log`	Provides information about errors generated by the server web client.

For more information about other available logs, see the following topics.

- Node logs
- Web Player service logs

Location of server logs

Find server logs at different locations.

Spotfire Server logs
Spotfire Server logs are located under `<`*`installation dir`*`>\tomcat\logs` folder.

Example: `C:\tibco\tss\<`*`version`*`>\tomcat\logs`

Spotfire Server Upgrade logs
Spotfire Server Upgrade logs are located under `<`*`installation dir`*`>\tools\upgrade\logs` folder.

Example: `C:\tibco\tss\<`*`version`*`>\tools\upgrade\logs`

To change these default locations, see Changing the default location of server logs.

Changing the default location of server logs

You can change the default directory location for logs by changing a property setting in the `log4j2-custom.xml` file, located in `<installation dir>\tomcat\spotfire-config.`

Prerequisites

You must have administrative privileges on the Spotfire Server.

Procedure

1. On the machine where Spotfire Server is installed, locate the file `<installation dir>\tomcat\spotfire-config\log4j2-custom.xml`.
2. Locate and modify the following property:

   ```
   <Property name="log.dir">logs</Property>
   ```

   This property is relative to `<installation dir>/tomcat` unless the given value is an absolute path to an existing folder.

Log4j2 configuration properties

In the `log4j2-custom.xml`, located in `<installation dir>/tomcat/spotfire-config`, there are a number of properties that control log file size and rollover.

None of the following properties control Tomcat logs or the user action log.

Property	Description
`serverLogSizePolicy`	Controls the maximum size of server log files.
`serverLogDefaultRollover`	Controls when server log files are rolled over; that is, this property controls the maximum number of server logs there can be before the log files are overwritten.
`nonServerLogsSizePolicy`	Is similar to the `serverLogSizePolicy` property, but controls the other log files that are listed in Spotfire Server logs on page 264.
`nonServerLogsDefaultRollover`	Is similar to the `serverLogDefaultRollover` property, but controls the other log files that are listed in Spotfire Server logs on page 264.

The following properties control `catalina.<date>.log` and `localhost.<date>.log` .

Property	Description
`tomcatLogsSizePolicy`	Controls the maximum size of the log files in question.
`tomcatLogsDefaultRollover`	Controls how many days the logs should be saved.

Node logs

The node logs store important diagnostic information. The information can help in troubleshooting and resolving issues.

To view node manager logs, see Accessing Spotfire Server and node logs.

The following table is a partial list available from the **Select log file to view** drop-down list found in the node Log files page. These are the most important node manager log files. You can find information for other logs on this list in the following topics:

- Spotfire Server logs
- Service logs

Log file	Description
`jetty.log`	The output from the jetty container that the node manager runs within (similar to catalina.log).
`nm.log`, `nm.log.`*n* (*n* is a number between 1 and the maximum number of logs that is configured to roll through.)	Information about all activity on the node.
`nodemanager.txt`	Generated only when you create a troubleshooting bundle. If you download another troubleshooting bundle at a later time, this log file is overwritten with newer data.
`service-<guid>.log`	`STDOUT` from the service with the specific guid. This is a service instance log, and not an installation log.
`winsw.err.log`	`STDERR` output captured by the Windows service handler.
`winsw.out.log`	`STDOUT` output captured by the Windows service handler.

If you have an issue with the node manager, the `nm.log` generally provides the needed details.

Enabling Kerberos debug logging

You can troubleshoot issues with the Kerberos authentication by enabling Kerberos debug logging.

It is a good practice to back up the existing logs and clear the logs folder before capturing the debug logs.

Procedure

1. Export and open the `configuration.xml` file from `<server installation dir>\tomcat\spotfire-bin` folder in an XML editor or a text editor; for details, see Manually editing the Spotfire Server configuration file.

2. In the `configuration.xml` file, locate the configuration block:

```
<jaas-config>
   <name>SpotfireKerberos</name>
   <entries>
     <entry>
       <login-module-name>com.sun.security.auth.module.Krb5LoginModule</login-module-name>
       <control-flag>required</control-flag>
       <options>
         <option>
           <key>debug</key>
           <value>false</value>
         </option>
         <option>
           <key>useKeyTab</key>
           <value>true</value>
         </option>
         <option>
           <key>principal</key>
           <value>HTTP/spotfiretss@TEST.COM</value>
         </option>
         <option>
           <key>storeKey</key>
           <value>true</value>
         </option>
         <option>
           <key>keyTab</key>
           <value>${java.home}/lib/security/spotfire.keytab</value>
         </option>
       </options>
     </entry>
   </entries>
 </jaas-config>
```

3. Change the value for debug key from `false` to `true`.

```
<jaas-config>
   <name>SpotfireKerberos</name>
   <entries>
     <entry>
       <login-module-name>com.sun.security.auth.module.Krb5LoginModule</login-module-name>
       <control-flag>required</control-flag>
       <options>
         <option>
           <key>debug</key>
           <value>true</value>
         </option>
         <option>
           <key>useKeyTab</key>
           <value>true</value>
         </option>
         <option>
           <key>principal</key>
           <value>HTTP/spotfiretss@TEST.COM</value>
         </option>
         <option>
           <key>storeKey</key>
           <value>true</value>
         </option>
         <option>
           <key>keyTab</key>
           <value>${java.home}/lib/security/spotfire.keytab</value>
         </option>
       </options>
     </entry>
   </entries>
 </jaas-config>
```

4. Save and close the file.
5. Import the configuration using the import-config command. For example: `config import-config --comment="Enabled Kerberos Debug Logging"`
6. On the computer that is hosting the server, navigate to the `nm\config` directory. The default location is `<installation dir>\nm\config`.

7. Do one of the following:
   - If the `logging.properties` file is present in the directory:
     a. Open the `logging.properties` file in an XML editor or text editor.
     b. Replace the current contents of the file with the contents of the `logging-debug.properties-template`.
     c. Save and close the `logging.properties` file.
   - If the `logging.properties` file is *not* present in the directory:
     a. Make a copy of the `logging-debug.properties-template`.
     b. Rename the copy "logging.properties".
8. Restart the Spotfire Server service for the changes to take effect.

When the troubleshooting is completed, it is recommended to switch back to a lower logging level. You can quickly return to the **Standard (default)** level; for instructions, see Switching back to the Standard (default) logging level.

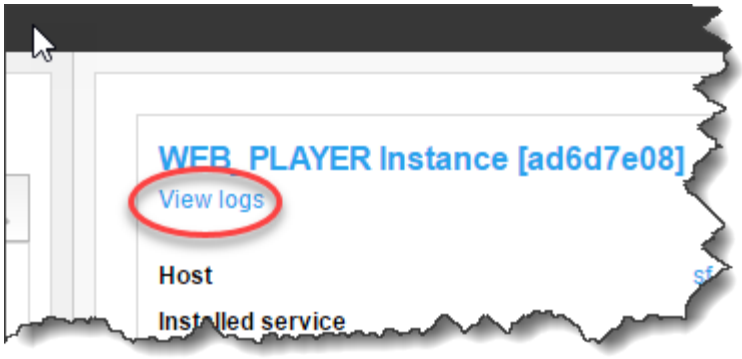
Accessing services logs

Spotfire Server provides easy access to logs for each service. You can select from a list of log files, and you can download the full log file for troubleshooting and working with Spotfire Support.

For more information about the logs created for the Web Player, see Service logs. For information about customizing these logs, see Customizing the service logging configuration.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
4. In the area displaying the selected instance, click **View logs**.



The Logs page is displayed in a new browser window.

5. In the **Select log file to view** drop-down list, select the type of log you want to view.
The selected log file is shown in the View logs area.

Service logs

The service logs listed in this topic are available for both Web Player services and Automation Services. You can configure the log files listed here in the file `log4net.config`.

To track the resource usage for services, you can enable logging and monitoring of the services by configuring log files in the `log4net.config` file. See Customizing the service logging configuration for information about exporting and editing this file.

The log4net tool is part of the Apache product family. For more information, see https://logging.apache.org/log4net/.

- In the configuration, specify writing all information to a log file by using the default format `%message`.
- For most log files, you can specify the logging level to write, and which properties to write.

For the TERR service and Spotfire Service for Python, JMX is the only way to capture logs. For more information, see the following.

- "Monitoring the TERR service using JMX" in TIBCO® Enterprise Runtime for R - Server Edition.
- "Monitoring Spotfire Service for Python using JMX" in TIBCO® Spotfire Service for Python.

General logging properties are written into each of the log files listed below. For more detailed information about the additional properties that can be written to each log file, see its linked reference topic.

Log file	Default level	Description
AuditLog.<ID>.txt	`INFO`	• At the `INFO` (default) level, for example, user login and logout, and analysis open and close are logged. • At the `DEBUG` level, state changes (apply and save) are also logged.
DateTimesLog.<ID>.txt	`OFF`	Time points from the services logs are collected in this file to simplify joins between tables. If logging is set to the `DEBUG` level, this file can get very large, so time points are not written at the microsecond level.
DocumentCacheStatisticsLog.<ID>.txt	`OFF`	The cached analyses are sampled regularly.
MemoryStatisticsLog.<ID>.txt	`OFF`	Writes resource usage per document. Logs the amount of memory used by tables and views, the number of internal document nodes, and the execution time. • At the `INFO` level, the total values per document are logged/ • At the `DEBUG` level, detailed information per table is recorded. , and
MonitoringEventsLog.<ID>.txt	`INFO`	• At the `INFO` level, the start up and shut down of the service are logged. • At the `DEBUG` level, session create and remove, analyses open and close, and cached analyses add and remove are also logged.

Log file	Default level	Description
OpenFilesStatisticsLog.<ID>.txt	`OFF`	The open analyses sampled regularly.
PerformanceCounterLog.<ID>.txt	`INFO`	Standard and custom performance counters logged regularly.
QueryLog.<ID>.txt	`OFF`	Logs all queries for data that are generated and sent to external data sources, when working with data connectors.
Spotfire.Dxp.Worker.Host.Debug.<ID>.log and Spotfire.Dxp.Worker.Host.<ID>.log	n/a	The general purpose log files. • `Spotfire.Dxp.Worker.Host.Debug.`*`<ID>`*`.log` writes all logging levels. • `Spotfire.Dxp.Worker.Host.`*`<ID>`*`.log` writes logging levels down to `INFO`
TimingLog.<ID>.txt	`INFO`	Logs similar information as the `AuditLog`, except all events also log a start time, an end time, and a duration.
UserSessionStatisticsLog.<ID>.txt	`OFF`	The existing sessions are sampled regularly.

For more information about other logs, see the following topics.

- Spotfire Server logs
- Node logs

General logging properties

The properties listed here are logged for all service log files.

Property	Description
`hostName`	The node name.
`timeStamp`	The local timestamp of the event.
`timeStampUtc`	The Coordinated Universal Time of the event.
`instanceId`	The unique ID of the running instance.
`serviceId`	The unique ID of the running service.

Auditlog

The Auditlog properties listed in this topic are written to the log file named `AuditLog.`*`<ID>`*`.txt`. By default, the logging level is set to `INFO`.

Property	Description
`sessionId`	The internal Spotfire session ID.
`ipAddress`	The IP address of the web client.

Property	Description
`userName`	The name of the logged on user.
`operation`	The audit operation, for example "Login".
`analysisId`	The document id of the currently open document.
`argument`	An argument for the operation, for example the path of the analysis.
`status`	Failure or Success.

DateTimesLog

DateTimesLog supports writing to the log file using only the `%message` format. The default level for DateTimesLog properties is `OFF`.

This log file compiles all time points from all service logs to simplify joins between tables. If you set this log to the `DEBUG` level, the resulting log file can be very large, so DateTimesLog does not compile time points at the microsecond level.

DocumentCacheStatisticsLog

The DocumentCacheStatisticsLog properties listed in this topic are written to the log file named `DocumentCacheStatisticsLog.<ID>.txt`. By default, the logging level is set to `OFF`.

Property	Description
`path`	The path of the currently open document.
`modifiedOn`	The date the document was modified.
`referenceCount`	The count of concurrent open references to the current document.

MemoryStatisticsLog

The MemoryStatisticsLog properties listed in this topic are written to the log file named `MemoryStatisticsLog.<ID>.txt`. By default, the logging level is set to `OFF`.

Property	Description
`sessionId`	The internal Spotfire session ID.
`userName`	The name of the logged on user.
`analysisId`	The unique ID for the analysis.
`tableId`	The unique ID for the table. This is empty if the value is a total.
`analysisPath`	The library path for the analysis.
`title`	The title of the analysis.

Property	Description
`type`	The type of information. Can be one of the following. • `SharedApproximateTotalTableSize` • `SharedApproximateTotalViewSize` • `DocumentNodeCount` • `SharedDocumentNodeCount` • `ApproximateExecutionTime`
`value`	The number of bytes, nodes, or milliseconds depending on type.

The table and view sizes (`SharedApproximateTotalTableSize` and `SharedApproximateTotalViewSize`) begin with the word "Shared" because these items can be shared among analysis instances. If two instances have the same `tableId` value, the memory chunks are shared. Also note that the values are approximate because calculating the exact size of a table or view size consumes extra resources.

MonitoringEventsLog

The MonitoringEventsLog properties listed in this topic are written to the log file named `MonitoringEventsLog.<ID>.txt`. By default, the logging level is set to `INFO`.

Property	Description
`eventType`	The type of event.
`argument`	Arguments related to the event.
`information`	Information related to the event.

OpenFilesStatisticsLog

The OpenFilesStatisticsLog properties listed in this topic are written to the log file named `AuditLOpenFilesStatisticsLogog.<ID>.txt`. By default, the logging level is set to `OFF`.

Property	Description
`sessionId`	The internal Spotfire session ID.
`filePath`	The path of the currently open document.
`modifiedOn`	The date the document was modified.
`fileId`	The file ID.
`elapsedTime`	The time since opened.
`inactiveTime`	The inactivity time.

PerformanceCounterLog

The PerformanceCounterLog properties listed in this topic are written to the log file named `PerformanceCounterLog.<ID>.txt`. By default, the logging level is set to `INFO`.

Property	Description
`counterCategory`	The category of the performance counter.
`counterName`	The name of the performance counter.
`counterInstance`	The instance of the performance counter.
`counterValue`	The value the performance counter returns.

QueryLog

The QueryLog properties listed in this topic are written to the log file named `QueryLog.<ID>.txt`. Each row in the log represents a query, which was generated from a data connector running on the Spotfire instance and sent to an external data source. By default, the logging level is set to `OFF`.

Property	Description
`Level`	The logging level
`HostName`	The name of the computer running the Spotfire service.
`TimeStamp`	The date and time, in the local time of the computer running the service, when the query was generated in Spotfire.
`UTCTimeStamp`	The date and time, in UTC, when the query was generated in Spotfire.
`QueryId`	The unique identifier of the query, as assigned by Spotfire.
`UserName`	The Spotfire username of the logged in user.
`Status`	Specifies whether the query succeeded, failed, or was canceled by the user.
`DurationMs`	The amount of time, in milliseconds, that the query took to execute in the external data source.
`RowCount`	The number of rows in the query result.
`ColumnCount`	The number of columns in the query result.
`DataSourceType`	The type of Spotfire connector that was used in the connection.
`DatabaseServer`	The URL or IP address of the server of the external data source.
`Database`	The name of the database in the external data source.
`DatabaseUser`	The database user that was used to log in to the external data source.

Property	Description
`Analysis`	The name of the Spotfire analysis file.
`Visualization`	The name of the visualization in the analysis that generated the query.
`Operation`	The type of operation that generated the query.
`DataSourceInfo`	Connector type specific information regarding the data connection.
`Parameters`	Any parameters in the query.
`QueryString`	The full query string sent from Spotfire to the external data source.

Spotfire.Dxp.Worker.Host and Spotfire.Dxp.Worker.Host.Debug

The properties for Spotfire.Dxp.Worker.Host and Spotfire.Dxp.Worker.Host.Debug are written to the log files `Spotfire.Dxp.Worker.Host.`*`ID`*`.log` and `Spotfire.Dxp.Worker.Host.Debug.`*`ID`*`.log`. These are general purpose log files for all logging levels, and for logging levels down to `INFO`, respectively. For the properties listed in this topic, you cannot use the standard Apache log4net pattern strings.

Property	Description
`pid`	The Process ID.
`user`	The name of the logged on user.
`windowsUser`	The Windows user.
`sessionId`	The internal Spotfire session ID.

TimingLog

The TimingLog properties listed in this topic are written to the log file named `TimingLog.<`*`ID`*`>.txt`. By default, the logging level is set to `INFO`.

Property	Description
`endTime`	The time the event ends.
`duration`	The duration of the event.
`sessionId`	The internal Spotfire session ID.
`ipAddress`	The IP address of the web client.
`userName`	The name of the logged on user.
`operation`	The audit operation, for example "Login".
`analysisId`	The document id of the currently open document.

Property	Description
`argument`	An argument for the operation, for example, the path of the analysis.
`status`	Failure or Success.

UserSessionStatisticsLog

The UserSessionStatisticsLog properties listed in this topic are written to the log file named `UserSessionStatisticsLog.<ID>.txt`. By default, the logging level is set to `OFF`.

Property	Description
`sessionID`	The internal Spotfire session ID.
`ipAddress`	The IP address of the web client.
`userName`	The name of the logged on user.
`browserType`	The name and (major) version number of the browser.
`cookies`	Returns `true` if cookies are enabled.
`loggedInDuration`	The duration of time the user has been logged in.
`maxOpenFilesCount`	The maximum number of open files.
`openFileCount`	The number of currently open files.

Action logs and system monitoring

Action logs collect user actions. System monitoring collects information about the performance of the Spotfire Server and the services. Information from action logs and from system monitoring is written to the same files or database; therefore, you can use the data you collect to correlate the usage with the system performance.

Action logging and system monitoring are disabled by default.

- To log such information from Spotfire Server, you must enable writing to files, to a database, or to both files and database.
- To also log information from non-server nodes, such as information from Spotfire Analyst, Automation Services and Spotfire Web Player (from Business Author and Consumer users), then you must configure Spotfire Server to accept incoming log events through web service calls.
- If you write the logs to a Microsoft SQL Server or Oracle database, you can choose to import the enclosed library import file, which contains information links for logging categories, as well as a sample Spotfire analysis file to your library. The information model and analysis file can easily be configured to read the logs from the database and it is a good starting point to analyze the logs.

Action logging and system monitoring	Comments
Writing to files.	Log files are not pruned. By default, a new log file is created every day; although you can change the action log interval, you must ensure that there is free space in the file system.

Action logging and system monitoring	Comments
Writing to a database.	You can set an option to remove entries that are older than a certain number of hours. Spotfire provides an information model and an analysis file that you can use to start analyzing usage patterns.  If you are logging to a database then it might be a good idea to involve your database administrator to regularly monitor the usage and see if indices should be rebuilt or dropped. If no pruning is turned on, then manual pruning or partitioning must eventually be performed on the database.
Capturing service logs.	You can specify the service or services for which to capture logging information. If you do not configure any web services, only actions performed on the server are logged.

You can enable and configure Spotfire Server for action logging and system monitoring either from the command line or from the configuration tool.

- To enable and configure action logging and system monitoring from the command line, follow the steps under Configure action logging using the command line on page 277.
- To enable and configure action logging and system monitoring using the configuration tool, follow the steps under Configure action logging using the configuration tool on page 283.

Note that the log points represent what is happening on the system. This means that there might be cases where what is shown in the log is not directly connected to a user action. For example, when using NTLM, you might see more login actions than expected. If you follow what is happening on the network during a normal session, you will see that several logins will take place during the session. Another case is when a session is automatically closed. There is a maximum life span for a session and, in the logs, you will see an event even if the user has not actively closed the session. You can also see that there might not be a session when these events are logged, because the session has already ended.

Configure action logging using the command line

By default, user action logging with system monitoring is not enabled or configured. You can enable and configure it from the server command line.

When configuring action logging to a database, you will have help from the included sample scripts. They are found in the installation kit under `<Spotfire Server installation kit>/scripts/{database type}_install/actionlog`.

The database scripts perform the following tasks:

- Create a user, schema and/or database.
- Create the ACTIONLOG table.
- Optionally, create index tables to help with searches on the ACTIONLOG table.
- Create some views for categories and actions with informative column names, and with the same information as that described in Action log entries on page 290. The views are needed only if you use them for analysis. During an upgrade, these views are the only things that must be updated in the database to enable new functionality.

Follow the guidance in this section to enable and configure action logging from the command line.

Enabling action logging and system monitoring using the command line

By default, action logging and system monitoring is not enabled. You can enable it from the server command line.

From the command line, running the **`enable-action-logging`** command is the first step.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to the Spotfire Server, and from the **Start** menu, open a command-line window as administrator.
2. Browse to `<installation dir>\tomcat\spotfire-bin`.
3. At the command prompt, type the command **`config config-action-logger`**, passing in the arguments specifying where to record the logs.

   - To write the action logs to a file, type the following.

     ```
     config config-action-logger --file-logging-enabled=true --database-logging-
     enabled=false
     ```

     Log files are not removed automatically. If you enable action logging to write to a file, remember to manage space needs for the resulting log files. By default, new log files are created on a daily basis, but the configuration can be changed. See Setting the action log interval for more information.

   - To write the action logs to a database, type the following.

     ```
     config config-action-logger --file-logging-enabled=false --database-logging-
     enabled=true
     ```

   - To write the action logs to both a file and a database, type the following.

     ```
     config config-action-logger --file-logging-enabled=true --database-logging-
     enabled=true
     ```

   In these examples, other command-line defaults are accepted. For example, the default configuration enables all categories for logging (**`categories="all"`**). To limit the enabled categories, provide a comma-separated list. See Action log categories for a complete list.

   For information about all available options for this command, see config-action-logger.

What to do next

- If you specify the database option, configure the action log to write to the database you use.
- To specify which services are allowed for logging on the server, see configure the action log web service.

Configuring logging to a Microsoft SQL Server database using the command line

You can configure action logging to write to a Microsoft SQL Server database by running additional scripts needed for logging to a database. Sample scripts are included in the installation kit for Spotfire Server.

This topic describes the steps required to run the configuration scripts from the command line. Alternatively, you can enable and configure the action logging and system monitoring using the configuration tool.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You must have enabled logging to a database.

Procedure

1. Log in to Spotfire Server, and in the file system, browse to the directory containing the installation kit files that you downloaded from the TIBCO eDelivery site.
2. In the installation kit directory, browse to the directory containing the scripts to create a new database and schema.

   For SQL Server, this directory is `/scripts/mssql_install/actionlog`.
3. Using a text editor, open the script file `create_actionlog_db.bat` (or, for Linux, `create_actionlog_db.sh`).
4. In the script file, edit the section containing the database name `spotfire_actionlog`, setting the variables to reflect your database environment.

   You must provide the database password in this script. If you do not have the password, consult your database administrator for assistance.

   If you want to use the information layer for analyzing action logs, you should not change the `ACTIONDB_NAME` from the default `spotfire_actionlog`, or else, you must use the **Redirect dependent elements** functionality in Information Designer (in Spotfire Analyst) to fix the mapping. See the Spotfire Analyst help topic Redirecting the Information Model for more information on this functionality.
5. Optional: If your database is running on Amazon RDS, also edit the script file `create_actionlog_db_rds.bat` (or, for Linux, `create_actionlog_db_rds.sh`), specifying the same information.
6. Run the script to create the database.

   Information and error logs are written to a file named `actionlog.txt` in the directory from where you run the script. If the script takes a very long time, or if it fails, check this text file for more information.

   The database is created on the server.
7. On Spotfire Server, from the **Start** menu, open a command-line window as administrator.
8. Browse to `<installation dir>\tomcat\spotfire-bin`.
9. Export the configuration: At the command prompt, type the command **`config export-config`**.

   ```
   config export-config --force
   ```

   When prompted, supply the tools password. See export-config for more information.
10. At the command prompt, type the command **`config config-action-log-database-logger`**, passing in the arguments specifying the details of the database.
    For example, to specify the Microsoft SQL Server database URL, driver class, user name, and password, provide the following.

    ```
    config config-action-log-database-logger --database-url="jdbc:sqlserver://
    [mycompany]:1433;DatabaseName=[Mydatabase]"
    --driver-class="com.microsoft.sqlserver.jdbc.SQLServerDriver" --
    username="spotfire_actionlog"
    ```

    When prompted, supply the tools password. See config-action-log-database-logger for more information.
11. At the command prompt, type the command **`config import-config`**.

    ```
    config import-config --comment="adding database configuration for action logging."
    ```

    When prompted, supply the tools password. See import-config for more information.
12. Restart Spotfire Server.

Result

The database is configured.

What to do next

If desired, follow the instructions in Importing a library for analyzing action logs in Spotfire Analyst on page 286, to get a quick start with your action log analysis.

Configuring logging to an Oracle database using the command line

You can configure action logging to write to an Oracle database by running additional scripts needed for logging to a database. Sample scripts are included in the installation kit for Spotfire Server.

This topic describes the steps required to run the configuration scripts from the command line. Alternatively, you can enable and configure the action logging and system monitoring using the configuration tool.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You must have enabled logging to a database.

Procedure

1. Log in to Spotfire Server, and in the file system, browse to the directory containing the installation kit files that you downloaded from the TIBCO eDelivery site.
2. In the installation kit directory, browse to the directory containing the scripts to create a new database and schema.

   For Oracle, this directory is `/scripts/oracle_install/actionlog`.
3. Using a text editor, open the script file `create_actionlog_db.bat` (or, for Linux, `create_actionlog_db.sh`).
4. In the script file, edit the section containing the database name `spotfire_actionlog`, setting the variables to reflect your database environment.

   You must provide the database password in this script. If you do not have the password, consult your database administrator for assistance.

   If you want to use the information layer for analyzing action logs, you should not change the `ACTIONDB_USER` from the default `spotfire_actionlog`, or else, you must use the **Redirect dependent elements** functionality in Information Designer (in Spotfire Analyst) to fix the mapping. See the Spotfire Analyst help topic Redirecting the Information Model for more information on this functionality.
5. Optional: If your database is running on Amazon RDS, also edit the script file `create_actionlog_db_rds.bat` (or, for Linux, `create_actionlog_db_rds.sh`), specifying the same information.
6. Run the script to create the database.

   Information and error logs are written to a file named `actionlog.txt` in the directory from where you run the script. If the script takes a very long time, or if it fails, check this text file for more information.

   The database is created on the server.
7. On Spotfire Server, from the **Start** menu, open a command-line window as administrator.
8. Browse to `<installation dir>\tomcat\spotfire-bin`.

9. Export the configuration: At the command prompt, type the command **`config export-config`**.

   ```
   config export-config --force
   ```

   When prompted, supply the tools password. See export-config for more information.

10. At the command prompt, type the command **`config config-action-log-database-logger`**, passing in the arguments specifying the details of the database.
    For example, to specify the Oracle database URL, driver class, user name, and password. The following example demonstrates the information you must provide.

    ```
    config config-action-log-database-logger
    --database-url="jdbc:tibcosoftwareinc:oracle://
    some.oraserver.com:1521;ServiceName=pdborcl.example.com"
    --driver-class="tibcosoftwareinc.jdbc.oracle.OracleDriver" --
    username="spotfire_actionlog"
    ```

    When prompted, supply the tools password. See config-action-log-database-logger for more information.

11. At the command prompt, type the command **`config import-config`**.

    ```
    config import-config --comment="adding database configuration for action logging."
    ```

    When prompted, supply the tools password. See import-config for more information.

12. Restart Spotfire Server.

Result

The database is configured for use.

What to do next

If desired, follow the instructions in Importing a library for analyzing action logs in Spotfire Analyst on page 286, to get a quick start with your action log analysis.

Configuring logging to a PostgreSQL database using the command line

You can configure action logging to write to a PostgreSQL database by running additional scripts needed for logging to a database. Sample scripts are included in the installation kit for Spotfire Server.

This topic describes the steps required to run the configuration scripts from the command line. Alternatively, you can enable and configure the action logging and system monitoring using the configuration tool.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You must have enabled logging to a database.

Procedure

1. Log in to Spotfire Server, and in the file system, browse to the directory containing the installation kit files that you downloaded from the TIBCO eDelivery site.
2. In the installation kit directory, browse to the directory containing the scripts to create a new database and schema.
   For PostgreSQL, this directory is `/scripts/postgres_install/actionlog`.
3. Using a text editor, open the script file `create_actionlog_db.bat` (or, for Linux, `create_actionlog_db.sh`).

4. In the script file, edit the section containing the database name `spotfire_actionlog`, setting the variables to reflect your database environment.

   You must provide the database password in this script. If you do not have the password, consult your database administrator for assistance.

5. Run the script to create the database.

   Information and error logs are written to a file named `actionlog.txt` in the directory from where you run the script. If the script takes a very long time, or if it fails, check this text file for more information.

   The database is created on the server.

6. On Spotfire Server, from the **Start** menu, open a command-line window as administrator.
7. Browse to `<installation dir>\tomcat\spotfire-bin`.
8. Export the configuration: At the command prompt, type the command **`config export-config`**.

   ```
   config export-config --force
   ```

   When prompted, supply the tools password. See export-config for more information.

9. At the command prompt, type the command **`config config-action-log-database-logger`**, passing in the arguments specifying the details of the database.
   For example, to specify the PostgreSQL database URL, driver class, user name, and password, provide the following.

   ```
   config config-action-log-database-logger --database-url="jdbc:postgres://
   [mycompany]:5432/[Mydatabase]"
   --driver-class="org.postgresql.Driver" --username="spotfire_actionlog"
   ```

   When prompted, supply the tools password. See config-action-log-database-logger for more information.

10. At the command prompt, type the command **`config import-config`**.

    ```
    config import-config --comment="adding database configuration for action logging."
    ```

    When prompted, supply the tools password. See import-config for more information.

11. Restart Spotfire Server.

Result

The database is configured.

Configuring the action log web service using the command line

To collect logging from the Spotfire Server and specified services (Spotfire Analyst, Web Player, and Automation Services), first enable and configure writing to a file or to a database, and then enable and configure the action log web service.

If you do not configure the action log web service, then only actions performed on Spotfire Server are logged.

This task describes configuring the action log web service using the command line. Alternatively, you can enable and configure the action log web service using the configuration tool.

Prerequisites

You must have administrative credentials for Spotfire Server.

You must have completed the following tasks.

- Enabling action logging and system monitoring using the command line on page 277.

- Configure action logging to write to either a file or to a database.

Procedure

1. Log in to the Spotfire Server, and from the **Start** menu, open a command-line window as administrator.
2. Browse to `<installation dir>\tomcat\spotfire-bin`.
3. At the command prompt, type the command **`config config-action-log-web-service`**, passing in the arguments specifying the services for which to collect logs.
   For example, to enable all categories from all hosts, type the following command.

   ```
   config config-action-log-web-service --allowedHosts=".*" --categories="all"
   ```

   By default, all hosts are allowed and all categories are logged. If you want to reduce the traffic passing between services and the server, replace the default argument values.

   - Specify from which host the server should accept logging requests.
   - Specify which individual services are allowed for logging. Provide a comma-separated list.

     At startup, all configured services check the server for allowed categories. See Action log categories for a complete list.

Configure action logging using the configuration tool

By default, user action logging with system monitoring is not enabled or configured. You can enable and configure it from the Spotfire Server configuration tool.

Follow the guidance in this section to enable action logging using the configuration tool.

Setting action logging to write to a file using the configuration tool

If you need to capture action logs, you can configure the Spotfire Server to write the action logs to a file, a database, or both. This topic discusses writing an action log to a file.

Alternatively, you can also configure the action logs by using command-line commands.

Log files are not removed automatically. If you enable action logging to write to a file, remember to manage space needs for the resulting log files. By default, new log files are created on a daily basis, but the configuration can be changed in the `log4j2.xml` configuration file. See Setting the action log interval for more information.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You have performed initial configuration as described in Configuration using the configuration tool on page 46.

Procedure

1. On the computer running Spotfire Server, click **Start**, and locate and click **Configure TIBCO Spotfire Server**.
2. On the Configuration page, click **User Action log**.
   User Action Log configuration options are displayed.
3. For **Enable file logger**, select **Yes**.

4. Optionally, configure the web service to log actions from the configuration tool.
   If you do not configure the web service, only actions that occur on the Spotfire Server are logged.
5. At the bottom of the page, click **Save configuration**.
   The Save configuration wizard is displayed. **Database (recommended)** is the pre-selected option, used to immediately apply the new configuration.
6. Click **Next**.
7. Enter a comment about the changes done to the configuration, and then click **Finish**.

What to do next

Restart all services and Spotfire Server for your changes to take effect. The action logs are written to the `<server installation dir>\tomcat\logs\actionlogs` folder.

Setting action logging to write to a database using the configuration tool

If you need to capture action logs, you can configure the Spotfire Server to write the action logs to a file, a database, or both. This topic discusses writing an action log to a database using the configuration tool.

Alternatively, you can also configure the action logs by using command-line commands.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You have performed initial configuration as described in Configuration using the configuration tool on page 46.
- You must have a database established to collect the logs.

Procedure

1. On the computer running Spotfire Server, click **Start**, and locate and click **Configure TIBCO Spotfire Server**.
   Log in if needed.
2. On the Configuration page, click **User Action log**.
   User Action Log configuration options are displayed.
3. For **Enable database logger**, click **Yes**.
4. To set specific categories to log, for **Enable categories**, click **Some Categories**, and from the list, select the categories to log.
   Only those categories you select are added to the database logger queue. By default, all categories are logged.
5. To ensure certain categories are added to the database logger queue, select the Prioritized check box.
   See Database logging for more information.
6. Complete the **Database logger configuration** section, specifying the required database connection information.
7. Click **Test connection** to make sure the configuration works.
8. Optionally, change the default configuration settings.
9. Optionally, configure the web service to log actions from the configuration tool.
   If you do not configure the web service, only actions that occur on the Spotfire Server are logged.

10. At the bottom of the page, click **Save configuration**.
    The Save configuration wizard is displayed. **Database (recommended)** is the pre-selected option, used to immediately apply the new configuration.
11. Click **Next**.
12. Enter a comment about the changes done to the configuration, and then click **Finish**.

What to do next

Restart all services and Spotfire Server for your changes to take effect. The action logs are written to the specified database.

Configuring the action log web service using the configuration tool

To collect logging from the Spotfire Server and specified services (Spotfire Analyst, Web Player and Automation Services), first enable and configure writing to a file, to a database, or both, and then enable and configure the action log web service. This task describes configuring the action log web service from the configuration tool.

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You must have configured the server to write action logs to a file and/or to a database.

Procedure

1. On the computer running Spotfire Server, click **Start**, and locate and click **Configure TIBCO Spotfire Server**.
   Log in if needed.
2. On the Configuration page, click **User Action log**.
   User Action Log configuration options are displayed.
3. Set **Enable web service** to **Yes**.
   For this option to be enabled, you must have completed the prerequisite to write to a file or database to collect logs.
   The Web service configuration section is available.
4. Specify the settings for the web service configuration.
   - Specify the allowed host as a regular expression, if different from the default `.*`. For example, `192\.168\.[0-9]{1,3}\.[0-9]{ 1,3}$`
   - Specify which categories to allow to communicate with the server. The default is **All**. If you set this option to Some Categories, then you can select from the resulting list box the service categories to allow. See Action log categories for a complete list.

   At startup, a service reads the list and sends to the Spotfire Server user action logger only the user action information for those services that are allowed. If a service is not allowed, then at startup, it has no communication with the Spotfire Server action logger. This setting is useful if you want to remove high-volume services from filling the log files.

   If you set the property to enable a service, but you do not set the property to allow it, remember that no communication is sent from the service to the logger.
5. At the bottom of the page, click **Save configuration**.
   The Save configuration wizard is displayed. **Database (recommended)** is the pre-selected option, used to immediately apply the new configuration.
6. Click **Next**.

7. Enter a comment about the changes done to the configuration, and then click **Finish**.
8. Restart all servers and services

Importing a library for analyzing action logs in Spotfire Analyst

The installation kit includes a library import file containing information links and a sample analysis file so that you can easily analyze your user action logs (only available for Microsoft SQL Server and Oracle databases).

Prerequisites

- You must have administrative credentials for Spotfire Server.
- You must have enabled action logging, configured action logging to a database (MSSQL or Oracle), and configured the web service to specify which services to log.

Procedure

1. In the installation kit that you downloaded from the TIBCO eDelivery site, browse to the directory containing the scripts to create a new database and schema.

   This directory is found under `/scripts/{database_type}_install/actionlog`.
2. In the installation kit directory, find the `logged_user_actions_{database_type}.part0.zip` file.
3. On Spotfire Server, open a command line as administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
4. On the command line, type the command **`config import-library-content`**, specifying the options needed to import the `.zip` file.

   Example:

   ```
   config import-library-content --tool-password=<password> --file-path=/scripts/
   oracle_install/actionlog/logged_user_actions_ora.part0.zip --conflict-resolution-
   mode=KEEP_BOTH --user=jdoe
   ```

   See import-library-content for more information.
5. Open Spotfire Analyst.
6. From the menu, click **Data** > **Information designer**.
7. In Information designer, click the Data Sources tab, locate the data source `logged_user_actions_datasource`, right-click on it and select **Edit**.
8. Provide information to connect to the data source, and then save the changes.

   You must provide the **Type**, the **Connection URL**, the **Username**, and the **Password** to the database.

Result

The analysis is now ready to use, and will show you an overview of your system monitoring and user action log data from your logging database.

Setting the action log interval

If you configure the Spotfire Server to write an action log to a database or a file, then the log is updated and events are logged when an action is performed. If you write the action log to a file, by default, a new file is created on a daily basis, but you can change this setting by editing the `log4j2.xml` configuration file.

In contrast to other log files, action log files are not removed automatically. Instead, a new file is created on a daily basis. Ensure that there is room in the file system by clearing old files on a regular basis.

Prerequisites

You must have administrative credentials for Spotfire Server.

Before editing the `log4j2.xml` configuration file, make a backup copy.

Procedure

1. On the computer running Spotfire Server, open the following file in a text editor or an XML editor: `<server installation dir>/tomcat/spotfire-config/log4j2.xml`.
2. Find the appender section specifying `<RollingFile name="actionlog"...>`
3. Edit the `filePattern` entry to specify a different interval.
   For detailed information about `filePattern`, see https://logging.apache.org/log4j/2.x/manual/appenders.html#RollingFileAppender.
4. Save and close the file.
5. Restart the server service.

Result

Any new action log files are created at the new interval.

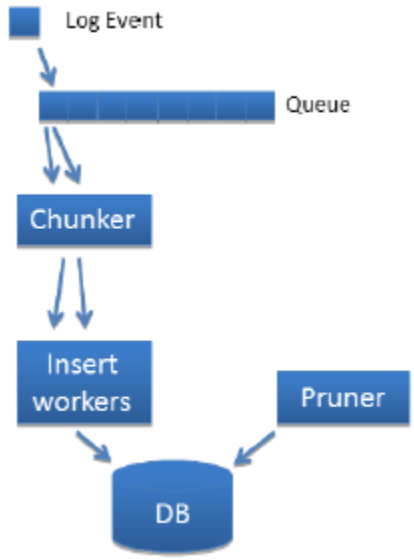
Database logging

When you configure Spotfire Server to log user actions to a database, you create a dependent and integrated system that you can tune to your logging needs. You can monitor its health with a JMX-compatible application such as JConsole.

If you enable database logging, then the server depends on being able to connect successfully to the database. During startup, the database logger attempts to connect to the database. If the database logger fails to connect, it attempts to reconnect at increasing intervals. If the database logger is not successful after the startup attempts, the server does not run.

Times are logged as GMT by default. To change the logging times value to local time, in the Spotfire Server configuration tool, go to the User Action Log page and set **Log in local time** to **Yes**.

Because several configuration options are available for the database logging, you can tailor the action logging system for your needs. To learn more about how database logging works, follow the steps for event logging.



1. Spotfire Server registers an event and checks if action logging is enabled.
2. If yes, then Spotfire Server checks if the category where the event occurred is enabled for logging.
3. If yes, then the event information is sent to one or two of the loggers.
   - If file logging is enabled, the event is written to the file.
4. Spotfire Server checks if database logging is enabled.
5. If yes, the database logger adds the event to a fixed-size queue. (The queue size is fixed at runtime.)

   You can configure the Spotfire Server logging queue to handle the following conditions. See config-action-log-database-logger for more information.
   - Control the maximum number of log events in the queue.
   - If the queue is more than half full, prioritize events so that only certain events are added to the queue.
   - If the queue is full, wait until there is room in the queue.
   - If the queue is full, wait for a given period of time.
6. The chunk worker waits until the configured number of events are available, or until the configured amount of time has passed.
7. The chunk worker starts an insert worker.

   You can configure the number of simultaneous insert workers. If the limit of simultaneous workers is reached, the chunk worker waits for an insert worker to finish. See config-action-log-database-logger for more information.
8. The insert worker runs a batch insert into the database.

To manage the size and performance of the database, consider the following additional configurations to the action log database logger.

Action	Configuration option in `config-action-log-database-logger`
If everything must be logged, set the database logger to block for a place in the queue.	`--block-on-full-queue=true`

Action	Configuration option in `config-action-log-database-logger`
Prioritize desired categories. If the queue is more than half full, the database logger adds to the queue only events in the prioritized categories. Other events are discarded.	`--prioritized-categories=<value>`
To ensure that important elements are not discarded, set the queue to wait if it is full.	`--wait-on-full-queue-time=<value>`
If the load is high, set multiple simultaneous insert workers. Otherwise, if you want to sample the system, and you do not want to load a database instance, set the number of insert workers to a low number.	`--workers=<value>`
By default, the database pruner checks every hour for events older than the set number of hours (by default 48 hours). The events that are older are deleted. If you set the number of hours to 0, no pruning takes place, and your database administrator must manage the growth through some other means (for example, by manually pruning, or by partitioning the table).	`--pruning-period=<value>`
Set a grace period, in seconds, to move events that are in the queue to the database when Spotfire Server is shutting down. Spotfire Server attempts to write these remaining events during this grace period.	`--grace-period=<value>`

The database administrator should monitor the usage regularly to determine if index tables should be rebuilt or dropped.

When you initially configure the action logger to send user action logs to a database, you must run database scripts. These scripts create a new schema and database for the action logs to make it easier to partition the data table. (See Configure action logging using the command line on page 277 and subtopics for more information about creating a database and schema with these scripts.)

- Events for enabled categories are logged to the table ACTIONLOG, and index tables are created. If you run database searches, you can omit these index tables. (See Update action logs and system monitoring for more information.) If you include the index tables, and you also set the option for pruning, then your database administrator should consider rebuilding the index tables periodically. See your database administrator for more information.
- Views are created for categories and actions. These views help to interpret the generic columns. If you do not use the views, then you can omit them from the database creation script.

By specifying these options from the command-line command `config config-action-log-database-logger`, you can tune the system for your particular environment and load. Additionally, you can use JMX to tune the system. See Monitoring for more information about using JMX with Spotfire Server.

In JConsole, under `com.spotfire.server`, you can examine the attributes for `action-log-db-worker`, of type `ActionDBLogger`, to answer the following questions.

Question	JMX Attribute
How many more insert workers can be started?	`CurrentNumberOfSpareWorkers`

Question	JMX Attribute
How many events are in the queue?	`CurrentQueueSize`
What is the minimum number of spare insert workers since the server was started? (0 indicates that all possible workers were started at some point.)	`MinimumFreeWorkers`
How many events have not been put in the database?	`NumberOfFailedLogs`
How many events have tried to be logged?	`NumberOfLogged`
How many items have been pruned from the database?	`NumberOfPrunedEntries`
How many SQL Exceptions have been encountered?	`NumberOfSQLExceptions`
How many more events can be queued?	`RemainingQueueCapacity`

For some database types, the installation kit also includes a library import file, which you can use to gain insight into the usage of the system. See Importing a library for analyzing action logs in Spotfire Analyst on page 286 for instructions on how to get access to the Information Services model and analysis file.

Action log reference

Spotfire Server action logs capture usage data, such as when a user logs in, opens a file from the library, adds bookmarks, pages through analyses, and so on. Action logs capture events from Spotfire Server and can also be configured to capture events from Automation Services, Spotfire Analyst, and Spotfire Business Author.

You can use the action logs to log users' actions in Spotfire, but you cannot use it to log the user state. For example, you can log when a user changes licenses or access permissions for another user (user actions), but you cannot log which actions a user is allowed to perform (user state).

Actions are collected in the logs and stored in files or in a database. Actions that do not originate from the server are sent to Spotfire Server through a web service.

- You must enable and configure the web service if you want actions that do not originate from the server to be logged.
- When you enable action logging, you must restart all service instances. If you do not restart all service instances, your logging changes will not take effect.

Action log entries

When you analyze an action log, you can organize the data into categories, actions, and properties (identifiers and arguments). You can map these to database columns, which you can display in a Spotfire visualization.

Different levels and types of information are logged. The log entries include the following information:

- The time of the action.
- The time the server logged the action.
- The addresses for the server and the computer where the action was performed.

- The user account that performed the logged action.
- The category of the action, specifying whether the action originated on the Spotfire Server (such as an admin action) or from a service (such as Automation Services).
- The properties (identifying properties or arguments) specific to the action performed. For example, when a user changes a password, the property `uName` is logged to indicate the user name.
- Whether the action was completed successfully.
- The session and service instance unique identifiers.

How the data is presented

Every logged event is placed on a new row in the log, and semicolons separate the entries within the logged event.

The logged information appears in the following fields, in the order given.

```
LOGGED_TIME; MACHINE; USER_NAME; ORIGINAL_TIME; ORIGINAL_IP; LOG_CATEGORY; LOG_ACTION;
 SUCCESS; SESSION_ID; ID1; ID2; ARG1; ARG2; ARG3; ARG4; ARG5; ARG6
```

The properties and arguments correspond to the fields `id1`, `id2`, and `arg1` through `arg6`.

For categories with the suffix "_wp" (Web Player), `ARG5` corresponds to `SERVICE_INSTANCE_ID`.

Example

When a user changes a password, the text log entry resembles the following.

```
2019-03-18T09:36:00.381+0100;10.100.32.129;jdoe;2019-03-18T09:36:00,381+0100;10.98.45.189;
admin;change_passwd;true;7583cdc4-a6b8-40d4-88e6-90f5d499ff79
```

For logs that are written to a database, the fields correspond to column names. The logged data is mapped to the appropriate columns to create a database view.

When the log entry is written to the database, the data appears in a specific view. For an Oracle database, the view is defined as the following.

```
CREATE OR REPLACE VIEW ADMIN_CHANGE_PASSWD AS SELECT
LOGGED_TIME;
MACHINE;
USER_NAME;
ORIGINAL_TIME;
ORIGINAL_IP;
SUCCESS;
SESSION_ID;
ID1 AS UNAME FROM ACTIONLOG WHERE LOG_CATEGORY = 'admin' AND LOG_ACTION = 'change_passwd'
```

To view the table of action log entries, go to the "Action log entries" topic in the Spotfire Server Installation and Administration help. The table is too wide for this PDF document.

For more information, see Action log generic entries and Action log categories. For an example of a typical set of user actions logged to the action logs, see Sample action log output.

You can configure action logging so that only certain categories are logged. See Configuring the action log web service from the configuration tool for more information.

Action log generic entries

Each log entry contains the generic information described in this topic.

Log entry	Description	Example
logged_time	The time the event was logged, in the format *YYYY-MM-DDTHH:MM:SS:mic+rosc*.	`2019-03-18T09:36:12.739+0100`

Log entry	Description	Example
machine	The IP address of the computer that performed the logging.	`10.100.21.230`
user_name	The name of the authenticated user that performed the logged action.	`JDOE`
original_time	The time the logged event was originally created, in the format `YYYY-MM-DDTHH:MM:SS:mic+rosc`. This time might differ from the logged time, because it can take time for the log event to be written.	`2019-03-18T09:36:12,733+0100`
original_ip	The IP address from where the call originates. It can be a proxy.	`10.98.25.189`
log_category	The category of the event. See Action log categories for a complete list.	`analysis_wp`
log_action	The action performed. For example, change_passwd.	`set_page`
success	Reports whether the operation succeeded.	`true`
session_id	A unique ID for the session.	`1b15369d63bbed3a64b576b29d0a34a26f2871b8`
service_instance_id	A unique ID for the service instance. This value applies only for the categories with the suffix "_wp" (Web Player). It is listed as `arg5`.	`cdd2eec7-15cd-4e2f-b426-b88d04a41e22`

Action log categories

When you enable action logging, you can enable any of the following categories. When you configure the web service, you can specify from which services to accept requests. When you read the action logs, you can look at these categories for information about where user actions are being logged from. You can specify some or all categories from the command line or from the configuration tool.

category	Description
admin	An administrator request on the server.
analysis_as	A Spotfire analysis sent to the server by Automation Services.
analysis_pro	A Spotfire analysis sent to the server by Spotfire Analyst.
analysis_wp	A Spotfire analysis sent to the server by the Web Player (Spotfire Business Author or Spotfire Cloud.)
auth	An authorization request on the server.
auth_as	An authorization request sent from Automation Services.
auth_pro	An authorization request sent from the Spotfire Analyst.
auth_wp	An authorization request sent from the Web Player.
automation_job_as	An automation job sent from Automation Services.

category	Description
automation_task_as	An automation task sent from Automation Services.
data_connector_pro	A data connection request sent from Spotfire Analyst.
data_connector_wp	A data connection request sent from Web Player.
datafunction_pro	A data function sent from Spotfire Analyst.
datafunction_wp	A data function sent from Web Player.
datasource_pro	A data source request sent from Spotfire Analyst.
datasource_wp	A data source request sent from Web Player.
dblogging	Action logs written only if you log to a database.
ems	A server request for establishing a TIBCO Enterprise Message Service (EMS) connection.
file_pro	A file sent from Spotfire Analyst.
find_pro	A find request sent from Spotfire Analyst.
find_wp	A find request sent from Web Player.
info_link	An information link request on the server.
library	A library request on the server.
library_as	A library request sent from Automation Services.
library_pro	A library request sent from Spotfire Analyst.
library_wp	A library request sent from Web Player.
monitoring	A server monitoring measure on the server.
monitoring_wp	A server monitoring request from the Web Player.
routing_rules	A server request related to routing rules.
scheduled_updates	A server request related to scheduled updates.

admin actions logged on Spotfire Server

Spotfire Server can log actions that an administrator takes to manage users, groups, licenses, preferences, and so on. These actions are logged under the admin category.

The following administration actions are logged on the Spotfire Server. For a list of the specific properties that are logged for each action, see Action log entries.

For more information on administrator actions, see Administration.

Action logged	Description
change_passwd	Changed the password for the specified user.

Action logged	Description
create_group	Created the group with the specified name, display name, and email alias.
create_user	Created the user with the specified user name, display name, and email alias.
exclude_license	Removed the license feature from the specified group.
group_add_member	Added the specified user name to the specified group name, provide a sorting order, and a grouping ID.
group_remove_member	Removed the specified user name from the specified group, providing a sorting order and a grouping ID.
remove_license	Removed the license from the specified group.
remove_principal	Removed the principal with the specified name from the groupingId and sorts the results.
rename_principal	Renamed the principal, replacing the old name with the new name and re-sorts the results.
set_license	Set the license with the specified name to the specified group name.
set_preference	Set the preference with the specified name to the specified type, category, and ID.

auth actions logged from Spotfire Server

Spotfire Server can log user actions for authentication, such as logging in and logging out. Spotfire Server can also log authentication with impersonation credentials. These actions are logged under the category auth.

These authentication actions are logged on the Spotfire Server. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about authentication, see User authentication.

Logged action	Description
impersonate	The authentication for the specified user name is from an impersonation.
login	The specified user (email argument and display name) logged in to the specified client type and version.
logout	The specified user logged out.

dblogging actions logged from the database

If you configure your action logs to log to a database, you have an additional category: dblogging. This category has three actions.

Logged action	Description
pruned	Entries are deleted as a result of a pruning action.
startup	The server is started and logging begins.

Logged action	Description
shutdown	The server is shut down and logging ends.  There is a risk that this is action is not logged if the grace period is too short; however, normally it should be logged.

ems action logged from Spotfire Server

Spotfire Server logs connection requests that are sent from TIBCO Enterprise Message Service (EMS).

This EMS action is logged on Spotfire Server from EMS. For a list of the specific properties that are logged for this action, see Action log entries.

For more information about EMS, see the help at https://docs.tibco.com/products/tibco-enterprise-message-service.

Logged action	Description
create_connection	Created an EMS connection.

info_link actions logged from Spotfire Server

Spotfire Server can log actions that a user takes when using information links. These actions include creating, loading, getting data, and updating the information link. These actions are logged under the category info_link.

These information link actions are logged on Spotfire Server from Spotfire Analyst. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about these actions, see the help topics for information links in the Spotfire Analyst User's Guide.

Logged action	Description
create_il	Created an information link in the specified library, with the specified path.
get_data	Retrieved the data for the information link in the specified library, with the specified path.
load_il	Loaded the infromation link in the specified library, with the specified path.
update_il	updated the information link in the specified library, with the specified path.

library actions logged from Spotfire Server

Actions that a user takes that correspond to categories on Spotfire Server or Spotfire Analyst include managing library permissions, and creating, importing, exporting, moving, copying, and loading content. These actions are logged under the category library on Spotfire Server.

These library actions are logged on Spotfire Server. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about these actions, see the help topics on library administration in the Spotfire Analyst User's Guide.

When content is imported to the library, all the related `user_name` fields in the action log contain the text "NULL" except for the first log entry, which is made when the server receives the import request.

Action logged	Description
clear_perm	Cleared permissions for a folder. Can be recursive.
copy	Copied library content.
create	Created a library.
delete	Deleted an item from the library.
export	Exported an item in the library to the specified path.
import	Imported library content to the specified path.
load_content	Loaded the specified item from the library.
move	Moved an item in the library to the specified path.
remove_perm	Removed permissions for the specified name.
save_content	Saved content to the library.
set_group_perm	Set the group permissions.
set_user_perm	Set the user permissions.

routing_rules actions logged from Spotfire Server

Spotfire Server can log actions that control routing rules.

The following routing rules actions are logged on Spotfire Server. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about routing rules, see Routing Rules.

Logged action	Description
create	Created a routing rule.
create_schedule	Created a schedule for a routing rule.
delete	Deleted the routing rule.
disable	Disabled the routing rule.
enable	Enabled the routing rule
update	Updated the routing rule.

scheduled_updates actions logged from Spotfire Server

Spotfire Server can log actions that occur as a result of establishing and managing scheduled updates.

The following scheduled update actions are logged on Spotfire Server. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about scheduled updates, see Scheduled updates to analyses.

Logged action	Description
adjust_ratio	Anaysis load distribution logging.
analysis_update	A server request to update analysis.
cancel_update	A server request to cancel loading analysis.
evaluation	A server request to evaluate scheduled updates.
external_update	An external update request for an analysis.
job_cancel_load	A scheduled update request to cancel loading analysis.
job_execution	A scheduled update job task execution.
job_load	A scheduled update request to load analysis.
job_unload	A scheduled update request to unload analysis.
load	A server request to load analysis.
no_retry	No retry request will be sent.
no_update	No update request wiil be sent.
reload	A user request to manually load an analysis.
reschedule	A request to reschedule the rule.
retry	A scheduled update request to retry analysis update.
retry_exhausted	A scheduled update request to retry on exhuasted services.
routing	Route created by scheduled updates.
rule_schedule	A request to schedule a rule.
schedule_change	A server request to change the schedule.
su_evaluation	A server request to evaluate scheduled updates.
su_execution	Scheduled update execution.
su_request	A scheduled update request to process.
task_execution	A scheduled update request to execute a task.
unload	A server request to unload analysis.
update	A server request to update the rule.

Automation Services actions logged from the web service

Spotfire Server logs actions that are performed by Automation Services. They include starting and finishing tasks or jobs, logging in and out, loading content from the library, and applying bookmarks.

For information about each category, see Action log categories. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about Automation Services, see the Automation Services User's Guide.

Category: analysis_as

Logged action	Description
apply_bookmark	A bookmark with the specified name was applied to the specified library item in the specified path.

Category: auth_as

Logged action	Description
login	The specified user logged in to Spotfire Server.
logout	The specified user logged out from Spotfire Server.

Category: automation_job_as

Logged action	Description
job_finished	The specified Automation Services job finished.
job_started	The specified Automation Services job started.

Category: automation_task_as

Logged action	Description
task_finished	The Automation Services task finished.
task_started	The Automation Services task started.

Category: library_as

Logged action	Description
load	Loaded content from the library specified in Automation Services.

Spotfire Analyst actions logged from the web service

Spotfire Server logs actions that are performed by the user in Spotfire Analyst. Actions are logged according to the category.

For information about each category, see Action Log categories. For a list of the specific properties that are logged for each action, see Action log entries.

For more information about the actions, see the help topics in Spotfire Analyst.

Category: analysis_pro

Logged action	Description
apply_bookmark	Applied a bookmark to the specified analysis.
arrange_visualizations	Used the rearrange visualizations feature.
canvas_size	Specified the canvas size.
change_column_or_aggregation	Changed column or aggregation on an axis.
create_annotation	Added an annotation to the active visualization.
create_comment	Added a comment.
create_details_visualization	Created a details visualization.
create_page	Created a new page in the analysis.
create_visualization	Created a new visualization.
create_visualization_recommendations	Created a visualization from recommendations presented when analyzing marked data.
delete_page	Deleted a page from the analysis.
duplicate_page	Duplicated a page.
duplicate_visualization	Duplicated a visualization.
exclude_column_from_recommendations	Excluded column from relationships recommendations.
export	Exported the analysis content.
hide_page	Made a page hidden for consumers.
initiate_analyze_for_marking	Initiated calculations to find relationships between marked and unmarked data.
modify_filter	Modified a filter.
rename_page	Renamed a page.
reset_all_filters	Reset all the filters.
reset_all_visible_filters	Reset the filters that are currently visible.
reset_filter	Reset a filter.
set_custom_expression	Applied a custom expression on an axis.
set_page	Set a page in the specified library item.
show_page	Switched a hidden page to a shown page.
switch_visualization	Changed a visualization to another visualization type.

Logged action	Description
visualization_area_layout_mode	Specified responsive layout.
y_axis_number_of_scales	Specified the scale option used; Single, Dual, PerColor, or PerTrellisPanel.

Category: auth_pro

Logged action	Description
login	The specified user logged in to Spotfire Analyst.
logout	The specified user logged out from Spotfire Analyst.

Category: data_connection_pro

Logged action	Description
create_connection	Connected to a data source.
create_source	Created a data source.
get_data	Retrieved data from a data source.
load_connection	Loaded the data connection.
load_source	Loaded the source.
synch_connection	Synchronized the connection.
update_connection	Updated a connection.
update_source	Updated the source.

Category: datafunction_pro

Logged action	Description
execute	Ran a data function in Spotfire Analyst.

Category: datasource_pro

Logged action	Description
execute	Ran an analysis using a data source using the specified parameters.

Category: file_pro

Logged action	Description
load	Loaded a file in Spotfire Analyst.

Category: find_pro

Logged action	Description
search	Used the find feature.

Category: library_pro

Logged action	Description
close	Closed an analysis in Spotfire Analyst.
load	Loaded an analysis in Spotfire Analyst.

Web Player actions logged from the web service

Spotfire Server logs actions that are performed by users of Spotfire Business Author for all categories.

For information about each category, see Action Log categories. For a list of the specific properties that are logged for each action, see Action log entries.

For information about Spotfire Business Author, see the Business Author and Consumer User's Guide.

Category: analysis_wp

Logged action	Description
apply_bookmark	Applied a bookmark to the specified analysis.
arrange_visualizations	Used the rearrange visualizations feature.
change_column_or_aggregation	Changed column or aggregation on an axis.
create_annotation	Added an annotation to the active visualization.
create_comment	Added a comment.
create_details_visualization	Created a details visualization.
create_page	Created a new page in the analysis.
create_visualization	Created a new visualization.
delete_page	Deleted a page from the analysis.
duplicate_page	Duplicated a page.
duplicate_visualization	Duplicated a visualization.
export	Exported the analysis content.
hide_page	Made a page hidden for consumers.
modify_filter	Modified a filter.
rename_page	Renamed a page.

Logged action	Description
reset_all_filters	Reset all the filters.
reset_all_visible_filters	Reset the filters that are currently visible.
reset_filter	Reset a filter.
set_custom_expression	Applied a custom expression on an axis.
set_page	Set a page in the specified library item.
show_page	Switched a hidden page to a shown page.
switch_visualization	Changed a visualization to another visualization type.
visualization_area_layout_mode	Specified responsive layout.
y_axis_number_of_scales	Specified the scale option used; Single, Dual, PerColor, or PerTrellisPanel.

Category: auth_wp

Logged action	Description
login	Logged in to Spotfire Business Author.
logout	Logged out from Spotfire Business Author.

Category: data_connection_wp

Logged action	Description
create_connection	Created a data connection in Spotfire Business Author.
create_source	Created a data source in Spotfire Business Author.
get_data	Retrieved data from a data source.
load_connection	Loaded a connection.
load_source	Loaded the source.
synch_connection	Synchronized the connection.
update_connection	Updated the connection.
update_source	Updated the source.

Category: datafunction_wp

Logged action	Description
execute	Executed a data function.

Category: datasource_wp

Logged action	Description
execute	Executed a call to a data source.

Category: file_wp

Logged action	Description
load	Loaded a file.

Category: find_wp

Logged action	Description
search	Used the find feature.

Category: library_wp

Logged action	Description
clone	Cloned a library entry.
close	Closed a library entry.
load_start	Started loading a library entry.
load	Loaded a library entry.
update_start	Began updating a library entry.
update	Updated a library entry.

Action log properties

Each action log entry contains generic information, the category of the action, the action logged, and identifying information (id1 and id2), as well as arguments providing more detail about the action. The identifying information and arguments are the properties described in this reference.

For more information about how these properties are reported in a log entry, see Action log entries. For an example of a typical set of user actions and a sample log written as a result, see Sample action log output.

Property	Description	Categories that use this property
`analysisId`	A unique identifier for the instance of the analysis.	analysis_as analysis_wp library_wp scheduled_updates
`analysisPath`	The path to the analysis.	scheduled_updates

Property	Description	Categories that use this property
`arguments`	Any arguments passed to the server from the EMS.	ems
`captureState`	Indicates whether the state of the document is captured.	analysis_pro analysis_wp
`category`	Specifies the category of the preference.	admin
`clientType`	The type of client (for example, Spotfire Analyst).	auth
`clientVer`	The version of the client that is connecting.	auth
`conversationId`	The Id of the thread of comments.	analysis_pro analysis_wp
`dataSourceInformation`	Connector-specific information about the data source. Typically the location of the database.	data_connection_pro data_connection_wp
`dataSourceLibraryId`	The unique library identifier of the connected data source, if applicable.	data_connection_pro data_connection_wp
`dataSourceType`	The type of external data source.	data_connection_pro data_connection_wp
`destLibraryId`	The destination library unique identifier.	library
`destPath`	The destination library path.	library
`destination`	The Spotfire Web Player instance URL.	scheduled_updates
`destinationList`	A list of service URLs. This list is created in the application, based on the scheduled update.	scheduled_updates
`destinationName`	The name specifying the destination URL.	scheduled_updates
`displayName`	The display name for a user (for example, John Smith).	admin auth

Property	Description	Categories that use this property
`duration`	The amount of time the operation or operations took (in ms).	data_connection_pro data_connection_wp datafunction_pro datafunction_wp datasource_pro datasource_wp
`email`	The e-mail address.	admin auth
`excludingFunction`	This is a subfunction within a license that is not enabled.	admin
`exportFormat`	The format that the content is exported to.	analysis_pro analysis_wp
`exportPages`	The pages that are exported.	analysis_pro analysis_wp
`expression`	The search expression.	analysis_pro analysis_wp
`externalQuery`	The external query, as generated by the adapter.	data_connection_pro data_connection_wp
`filterName`	The column name of the specified filter.	analysis_pro analysis_wp
`filterType`	The type of the filter.	analysis_pro analysis_wp
`gName`	The group name.	admin library
`groupingId`	Operations related to the same operation can share a common grouping identifier. For some operations, this identifier is the same as the job identifier seen in the other logs.	admin info_link library
`id`	The name of the preference.	admin
`internalQuery`	The Spotfire query.	data_connection_pro data_connection_wp

Property	Description	Categories that use this property
`jobTaskId`	The unique identifier for the job task.	scheduled_updates
`jobid`	The unique identifier of the job.	automation_job_as automation_task_as scheduled_updates
`layoutName`	The layout name of the canvas size.	analysis_pro analysis_wp
`layoutSize`	The width and height of the canvas size.	analysis_pro analysis_wp
`libraryId`	The unique identifier for the library item.	analysis_as analysis_pro analysis_wp automation_job_as automation_task_as data_connection_pro data_connection_wp info_link library library_as library_pro library_wp scheduled_updates
`libraryPath`	The library path.	analysis_pro automation_job_as automation_task_as data_connection_pro data_connection_wp library_wp
`libraryType`	The type of library. For example, dxp. query.	library
`licenseName`	The license name.	admin
`message`	An informational message provided by the rule or the scheduled update.	scheduled_updates routing_rules
`name`	The name of the entity.	library

Property	Description	Categories that use this property
`newExpression`	The updated expression.	analysis_pro analysis_wp
`newName`	The new name.	admin
`numberOfScales`	The scale option that is selected on an axis.	analysis_pro analysis_wp
`numRows`	The number of rows returned.	data_connection_pro data_connection_wp datasource_pro datasource_wp
`oldExpression`	The original expression.	analysis_pro analysis_wp
`oldName`	The old name.	admin
`origin`	The access point of the action.	analysis_pro analysis_wp
`originVisualizationType`	The original visualization type.	analysis_pro analysis_wp
`origin userInput`	The access point and type of user interaction.	analysis_pro analysis_wp
`pageName`	The name of the page.	analysis_pro analysis_wp
`params`	For some operations we do not have the exact functionality, but this information can help identify the action.	datafunction_pro datafunction_wp datasource_pro datasource_wp

Property	Description	Categories that use this property
`path`	The path.	analysis_as analysis_wp datafunction_pro datafunction_wp datasource_pro datasource_wp file_wp file_pro info_link library library_as library_pro library_wp
`payLoad`	An object or a map containing information related to the specific action.	scheduled_updates
`permission`	The permission.	library
`postSize`	The size afterwards (in bytes).	library
`predictorsType`	The column categorization and data type.	analysis_pro
`preSize`	The size before (in bytes).	library
`prefType`	The type of the preference.	admin
`processType`	The type of the scheduled update process, such as `load`, `unload`, or `stop_loading`.	scheduled_updates
`recursive`	Indicates whether the performed action was recursive.	library
`resourcePool`	The resource pool used in the specific scheduled update.	scheduled_updates
`ruleName`	The name of the rule.	scheduled_updates routing_rules
`ruleId`	The unique identifier of the rule.	routing_rules
`scheduleId`	The unique identifier for the scheduled update.	routing_rules
`scheduleName`	The friendly name of the schedule update entry.	routing_rules

Property	Description	Categories that use this property
`serviceUrl`	The link to the Spotfire web service. (The web service is the Spotfire Web Player instance on which the scheduled update is running. This can be the same as `destination`.	scheduled_updates

`serviceStatus`	That status for the scheduled update service. Can be one of the following.
	`Failed`
	`Installing`
	`Restart`
	`Running`
	`Starting`
	`Stopped`
	`Stopping`
	`Unreacheable`
		scheduled_updates

`sort`	The type (a user or a group).	admin library
`taskId`	The unique identifier of the task.	scheduled_updates
`tileMode`	The arrangement of the visualizations.	analysis_pro analysis_wp
`title`	The document title.	datasource_pro datasource_wp
`uName`	The user name.	admin auth auth_as auth_pro auth_wp library
`unused`	This property is not used.	automation_task_as datafunction_pro datafunction_wp datasource_pro datasource_wp ems file_pro file_wp routing_rules

Property	Description	Categories that use this property
`visualizationAreaMode`	The page visualization mode.	analysis_pro analysis_wp
`visualizationTitle`	The title of the visualization.	analysis_pro analysis_wp
`visualizationType`	The type of the visualization.	analysis_pro analysis_wp

Sample action log output

Reading the output from an action log file can be challenging. The sample shown below demonstrates a series of user actions and the resulting log entry that the system provides.

User action	System output
The user jdoe logs in to Spotfire Server.	`2019-03-18T09:36:00.381+0100;10.100.32.118;jdoe;2019-03-18T09:36:00,381+0100; 10.98.45.199;auth;login;true;7583cdc4-a6b8-40d4-88e6-90f5d499ff79;;;jdoe;;;;;`
jdoe logs in to Spotfire Business Author. Note the session ID is `1b153....`	`2019-03-18T09:36:12.152+0100;10.100.32.130;jdoe;2019-03-18T09:36:12,140+0100; 10.98.45.199;auth_wp;login;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655;jdoe;;;; 1b15369d63bbed3a64b576b29d0a34a26f2871b8;;;`
jdoe loads from the library the DXP contents for the analysis `/drafts/MyAnalysis - first version`.	`2019-03-18T09:36:12.268+0100;10.100.32.118;jdoe;2019-03-18T09:36:12,267+0100; 10.100.32.130;library;load_content;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655; 79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;dxp; 0000000036;0001145557;;;`
The analysis is loaded into Spotfire Business Author. Note that the session ID matches the value above (`1b153...`), and the analysis ID for the analysis instance is `bwHPZ....`	`2019-03-18T09:36:12.722+0100;10.100.32.130;jdoe;2019-03-18T09:36:12,717+0100; 10.98.45.199;library_wp;load;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655;79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;AnalysisDxp;; 1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;`
jdoe flips through the pages. Note that the session ID and analysis ID match the values above.	`2019-03-18T09:36:12.739+0100;10.100.32.130;jdoe;2019-03-18T09:36:12,733+0100; 10.98.45.199;analysis_wp;set_page;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655; 79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;Intro;; 1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;`  `2019-03-18T09:36:16.408+0100;10.100.32.130;jdoe;2019-03-18T09:36:16,399+0100; 10.98.45.199;analysis_wp;set_page;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655; 79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;Algebra;; 1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;`  `2019-03-18T09:36:22.044+0100;10.100.32.130;jdoe;2019-03-18T09:36:22,031+0100; 10.98.45.199;analysis_wp;set_page;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655; 79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;Intro;; 1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;`

User action	System output
jdoe applies a bookmark	2019-03-18T09:36:22.528+0100;10.100.32.130;jdoe;2019-03-18T09:36:22,514+0100;10.98.45.199;analysis_wp;apply_bookmark;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655;79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;geometrics;;1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;
As jdoe closes the analysis, its state is saved to the library.	2019-03-18T09:36:27.279+0100;10.100.32.118;jdoe;2019-03-18T09:36:27,279+0100;10.100.32.130;library;create;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655;dbfc821b-0e02-494c-8360-cf8c9c3e07fe;/RelatedItems/AnalysisStates/092a7424-fa68-4179-b762-7f16a5c11e18;analysisstate;0000000000;0000028364;;;
jdoe closes the analysis.	2019-03-18T09:36:27.288+0100;10.100.32.130;jdoe;2019-03-18T09:36:27,288+0100;10.98.45.199;library_wp;close;true;21dc38aa-3ec7-4938-8b7e-1dfe218f8655;79c727c3-d70d-43f5-b681-360cee89a821;/drafts/MyAnalysis - first version;AnalysisDxp;;1b15369d63bbed3a64b576b29d0a34a26f2871b8;bwHPZisVZUeE_Nxj5ybYn-0414411f61_jf2;;
jdoe logs out from Spotfire Server and Spotfire Business Author.	2019-03-18T09:36:30.884+0100;10.100.32.118;;2019-03-18T09:36:30,884+0100;10.98.45.199;auth;logout;true;7583cdc4-a6b8-40d4-88e6-90f5d499ff79;jdoe;;;;;;; 2019-03-18T09:36:30.897+0100;10.100.32.130;jdoe;2019-03-18T09:36:30,892+0100;10.100.32.112;auth_wp;logout;true;15966a47-aafd-460e-a649-a80c020a9ca2;jdoe;;;;1b15369d63bbed3a64b576b29d0a34a26f2871b8;;;

System monitoring reference

System monitoring saves information about the performance of Spotfire Server and the services in the same database or files as the action logs.

System monitoring collects information at regular intervals.

- If you log to a database, to reduce the number of measurements in the database over time, measurements older than a specified amount of time are replaced with average, minimum, and maximum values for a given period of time. The general pruning for the database also affects the monitoring values.
- If you log to file, a file is created every day (the default), so no pruning or averaging is done, and you must manage the space needs of the files.

System monitoring entries

This reference lists all of the entries. When you analyze an action log, you can organize the semi-colon separated data into categories, actions, and properties (identifiers, and arguments). You can map these to database columns, which you can display in a Spotfire Analyst visualization.

Category	Action	id1	id2	arg1	arg2	arg3	arg4
monitoring	average	measure	unused	mean	min	max	
monitoring	measurement	measure	unused	value			
monitoring_wp	average	measure	unused	mean	min	max	
monitoring_wp	counter	measure	wp_id*	value	countercategory	countername	counterinstance
monitoring_wp	start_instance						
monitoring_wp	stop_instance						

*wp_id is a unique id that identifies the currently-running instance of the Web Player service.

System monitoring properties

Spotfire Server and the Web Player service instance log different properties. The properties are described in this reference.

The tables lists the different properties (id1, id2, arg1, arg2, arg3, arg4):

Spotfire Server

Measure	Description
cpu	Average CPU load, in percent.
mem	Heap memory used, in megabytes.
sessions	The number of authenticated HTTP sessions.

Spotfire Web Player service instance (_wp)

Measure	Description
available bytes	The available number of bytes.
cached docs	The number of cached documents.
cpu	Average CPU load, in percent.
disk queue	The length of the disk queue.
mem	The number of bytes used.
network	The total number of bytes transferred per second.
open docs	The number of open documents.
scheduled updates docs	The number of documents controlled by the scheduled updates feature.
uptime	The time in seconds since the service instance was started.

Update action logs and system monitoring

If you have been running action logging with a previous release of Spotfire Server, then logging should continue to run, out of the box, after a successful upgrade. However, you might not be able to make full use of any new functionality unless you perform some manual changes. If you upgrade by installing new computers, you must ensure that the new hardware can connect to the database.

Things that might be added in a newer release are new categories, actions, extra properties, measures, etc. Occasionally, there will be updates to database views for more important additions. For Microsoft and Oracle databases there is an example analysis file and information services layer available, which might be updated.

Depending on which categories you enabled earlier, you should review the list of categories (including the web services settings). If you configure user action logs with the configuration tool, selecting categories is easy: you can review and select categories using the check boxes. If you previously selected **all**, any new categories will automatically be enabled.

- If you are logging to a file only, then no additional changes are necessary.
- If you are logging to database, there are some things to note. As before, all measures are logged to one single table called "ACTIONLOG". New categories, actions, properties, and measures will be logged to this table and you should not lose any log points. This table is the only thing required to run the logging to database.

No SQL related to the action logging functionality is executed automatically during an upgrade. This design gives full control to you and your database administrator, so the upgrade will not interfere if you have chosen to do some specific adaption, e.g., partitioned the "ACTIONLOG" table.

The database scripts used to configure database logging can also be used to update the views. They include the following functionality:

- Create user, schema and/or database. After an update, you can continue to log to the same location, so you do not need to recreate these.
- Create the ACTIONLOG table. This table is still used, and the structure is not altered.
- Optionally, create index to help with searches on the ACTIONLOG table. If you configured your earlier installation to omit indexes, then you do not need to change this configuration. With pruning enabled, the ACTIONLOG table has rows both added and deleted, so indexes benefit from being rebuilt regularly.
- Create views for the most important categories and actions, with more informative column names, using the same information as described in Action log entries on page 290. The views are needed only if you use them for analysis. There might be more views in a later release. The SQL for creating the views is available in the `create_actionlog_db.sql` scripts, which are found in the installation kit in the following directory:

  `./scripts/{database type}_install/actionlog`

See the specific topic below for details on how to update the database in your environment.

Updating the Oracle database

When you update your Spotfire Server to a newer version, any user action database logging you have configured earlier will most likely continue to work. If there are new views available, you might want to add them.

See also Update action logs and system monitoring.

If you are familiar with SQL utilities, it is probably the fastest to log in to the schema `spotfire_actionlog` and run the SQL found in `create_actionlog_db.sql`. The SQL is written so that it can be run many times.

Otherwise, you can tweak the existing scripts to run the SQL.

Prerequisites

You must have credentials to the action log database.

Procedure

1. In the new installation kit directory, browse to `\scripts\oracle_install\actionlog`.
2. Using a text editor, open the script file `create_actionlog_db.bat` (on Windows) or `.sh` (on Linux).
3. In the file, remove the section that creates the tablespace and user. Then, enter the information for `CONNECTIDENTIFIER`, `ACTIONDB_USER`, and `ACTIONDB_PASSWORD`, and run the edited script the same way as when you enabled the database logging.

What to do next

The installation kit also includes a library import file, which contains information links for logging categories, as well as a sample Spotfire analysis file, which you can use to gain insight about your system. This import file might be updated with a new release and, if so, you can import it again. See Importing a library for analyzing action logs in Spotfire Analyst on page 286 for more information.

Updating the Microsoft SQL Server database

When you update your Spotfire Server to a newer version, any user action database logging you have configured earlier will most likely continue to work. If there are new views available, you might want to add them.

See also Update action logs and system monitoring.

- Using a text editor, open the file `create_actionlog_db.sql`.
- Remove the lines above `use $(ACTIONDB_NAME)` and change this line to `use spotfire_actionlog`. The script will only create the views that do not already exist.

If you are familiar with SQL tools, it is probably the fastest to log in to the database `spotfire_actionlog` and run the SQL in your edited `create_actionlog_db.sql`.

Otherwise, you can tweak the existing scripts to run the SQL.

Prerequisites

You must have credentials to the action log database.

Procedure

1. In the new installation kit directory, browse to `\scripts\mssql_install\actionlog`.
2. Using a text editor, open the script file `create_actionlog_db.sql` and edit it as described above.
3. Using a text editor, open the script file `create_actionlog_db.bat` (on Windows) or `.sh` (on Linux).
4. In the file, remove the section "Create the Spotfire Action log database user". Then, enter the information for `CONNECTIDENTIFIER`, `ACTIONDB_USER`, and `ACTIONDB_PASSWORD`.
5. Run the edited script the same way as when you enabled the database logging.

What to do next

The installation kit also includes a library import file, which contains information links for logging categories, as well as a sample Spotfire analysis file, which you can use to gain insight about your system. This import file might be updated with a new release and, if so, you can import it again. See Importing a library for analyzing action logs in Spotfire Analyst on page 286 for more information.

Updating the PostgreSQL database

When you update your Spotfire Server to a newer version, any user action database logging you have configured earlier will most likely continue to work. If there are new views available, you might want to add them.

See also Update action logs and system monitoring.

If you are a familiar with SQL tools, it is probably the fastest to log in to the database `spotfire_actionlog` and run the SQL in `create_actionlog_db.sql`.

Otherwise, you can tweak the existing scripts to run the SQL.

Prerequisites

You must have credentials to the action log database.

Procedure

1. In the new installation kit directory, browse to `\scripts\postgres_install\actionlog`.
2. Using a text editor, open the script file `create_actionlog_db.bat` (on Windows) or `.sh` (on Linux).
3. In the file, remove the section "Creating Spotfire Action log database user". Then, enter the information for `ACTIONDB_USER` and `ACTIONDB_PASSWORD`.
4. Run the edited script the same way as when you enabled the database logging.
   The script will only create the views that do not already exist.

Server monitoring using JMX

You can monitor the Spotfire Server to detect problems with the server itself, with external systems, or with the network. You can also detect misconfigured clients or (in some cases) malicious behavior.

Spotfire Server runs within the Tomcat application server. Tomcat provides the basic functionality needed, the server (Agent level), and a Java Remote Method Invocation (Java RMI) connector (Remote Management level).

Tomcat provides a rich instrumentation set for monitoring and managing the application server. For example, it monitors Tomcat configuration parameters and basic usage statistics. The Java runtime environment that ships with Spotfire Server is also heavily instrumented using JMX. This toolset provides information about CPU and memory usage, garbage collection, and thread pools. In addition, JMX is the only way to capture logs for the TERR service or for Spotfire Service for Python. For more information, see the following.

- "Monitoring the TERR service using JMX" in TIBCO® Enterprise Runtime for R - Server Edition.
- "Monitoring Spotfire Service for Python using JMX" in TIBCO® Spotfire Service for Python.

- To monitor the server itself, view and manage logs, and troubleshoot the server, log in to the Spotfire Server administration interface and see the Overview page of Monitoring and Diagnostics.
- To monitor user actions and system events, such as those from Spotfire, Spotfire Web Player, and Spotfire Automation Services, see Action logs and system monitoring.
- To monitor other aspects of the server, use available tools such as TIBCO Hawk®, JConsole (which is included in the Java JDK), or any other Java Management Extensions (JMX)-compliant monitoring tool.

This section provides information on the architecture of the JMX system, types of information captured by JMX, and how to configure and work with JMX-compliant tools to monitor Spotfire Server.

Spotfire Server instrumentation

Spotfire Server components are instrumented to capture detailed information. The following table provides details on the information that the administrator can monitor through instrumentation.

Spotfire Server component	Instrumented information
Server	• Server address (IP). • Server hostname. • Server version. • Date and time the server was started. • Uptime time since the server was started, both as a formatted string and in milliseconds since January 1, 1970, 00:00:00 GMT.
Logging	• Current log configuration file (configurable). • Available log configuration files (read only). – Lists all log configuration files in `<installation dir>\tomcat\webapps\spotfire\WEB-INF`. • The number of logging events for the levels set to `warn`, `error`, and `fatal`.
Logger	Information captured depends on the log configuration. It can be set to capture no logs, a single log, or several logs. • Log appender name. • Notifications. (Outputs all log statements from a configured log4j appender as JMX notifications.)
Server metrics	• Number of attachments on the server. • Number of running Information Services jobs. • Number of authenticated HTTP sessions.
HTTP status codes	The number of HTTP response codes representing client or server errors. Includes the 4xx and 5xx ranges returned from the server.  Responses in these series can be common, even in a system that works well.
Data source	Records one entry for each currently-running data source on the server, including the server's own data source, as follows. • Name. • URL. • Configured minimum number of connections. • Configured maximum number of connections. • Current number of active connections. • Current number of idle connections. • The maximum number of concurrently active connections seen.

JMX configuration security features

Sensitive information can be exposed through JMX and Java. Tomcat and Spotfire Server provide management capabilities to restrict access through authentication, authorization, and encryption security features. Also, as a security measure, the JMX RMI connector is disabled by default, so the administrator must enable it.

Security feature	Description	Default setting
Authentication	Spotfire Server applies the existing database authentication mechanism using a separate database table. Passwords are obscured with hash marks. you can use the same principal names across an entire Spotfire Server cluster.	Enabled.
Authorization	You can configure authorization to specify the level of user permissions. • If a user has only read permissions, the user can only read attribute values. • If a user has read-and-write permissions, the user can read and modify any writable attributes. JMX accounts and credentials are separated from Spotfire accounts and credentials. The JMX accounts are used only for monitoring.	Enabled.  Authorization works only with the default authentication implementation.
Encryption	You can configure the Remote Method Invocation (RMI) connector to encrypt the traffic using Transport Layer Security (TLS). This configuration is recommended; otherwise, user names and passwords are transmitted in plain text.	Not enabled.  Encryption configuration requires a certificate.
Firewall	You can configure a firewall to allow traffic to the desired ports.	The RMI registry and the RMI connector share a common port (1099) to simplify firewall configuration.

JMX configuration commands

Use these commands to configure and administrate JMX access to the monitoring component.

JMX configuration command	Description
`config-jmx`	Configures the JMX RMI connector.
`create-jmx-user`	Creates a new JMX user account.
`delete-jmx-user`	Deletes a JMX user.
`list-jmx-users`	Lists all JMX users.

Except for the `config-jmx` command, which works on the `configuration.xml` file, all monitoring commands connect directly to the database. You must first import the `configuration.xml` file using the `import-config` command for any changes to take effect. See Setting up JMX monitoring for JConsole and import-config for more information.

 Click the links in the table for detailed reference for these configuration commands.

JMX levels

A Java Management Extensions (JMX)-compliant monitoring tool, such as TIBCO Hawk® or JConsole, provides three administration levels to Spotfire Server administrators.

The three JMX administration levels are as follows.

JMX administation level	Description
Remote Management level	This level contains connectors and adaptors that provide access to the Agent level.
Agent level	This level is a server that provides applications access to the Instrumentation level.
Instrumentation level	This level provides monitoring information and management operations.

Enabling the JMX logging appender

To monitor the server by using TIBCO Hawk or another Java Management Extensions (JMX)-compliant monitoring tool, you can enable an extra log appender so that the server outputs log events as JMX notifications.

Prerequisites

You must have write access to the server where Spotfire Server is installed.

Perform this task on the computer where Spotfire Server is installed.

Procedure

1. Open the following file in a text editor or an XML editor: `<server installation dir>/tomcat/spotfire-config/log4j2.xml`.
2. Add a new appender definition to the `<Appenders>` section of the `log4j2.xml` file.
   For example:

   ```
   <Jmx name="Jmx" description="description of the log">
       <PatternLayout pattern="[%X{thread.info}] %c{3}: %m%n" />
     </Jmx>
   ```

   where the values of the `name`, `description`, and `pattern` attributes can be changed, and the `description` attribute is optional.
3. In the `<Loggers>` section of the file, locate the loggers for which you want to enable JMX functionality and then add a reference to the JMX appender. See the fourth line of the following example code.
   For example:

   ```
    <Logger name="com.spotfire" level="DEBUG" additivity="false">
       <AppenderRef ref="serverlog"/>
       <AppenderRef ref="Console"/>
       <AppenderRef ref="Jmx"/>
     </Logger>
   ```

   You can configure multiple JMX appenders, but each one must have a different value for the `name` attribute.
4. Save and close the file.
5. Restart the server service.

Setting up JMX monitoring for JConsole

This task walks you through setting up JMX monitoring for using JConsole. It does not use Transport Layer Security (TLS).

Prerequisites

- You must have administrative credentials for Spotfire Server. If you are running these commands in Windows, run the command-line interface as administrator.
- You must have access to JConsole.

Perform this task at a command-line prompt on the server, from the directory where the file `config.bat` (on Windows) or `config.sh` (on Linux) is installed. By default, this location is `<server installation dir>/tomcat/spotfire-bin`.

Procedure

1. Log in to the Spotfire Server, and from the **Start** menu, open a command-line window as administrator.
2. At the command line, run the command **`config export-config`**.
   Provide the tools password when prompted.
   The configuration is successfully exported and is ready to change.
3. At the command line, run the command **`config config-jmx --enabled=true`**.
   Provide the tools password when prompted.
4. Import the configuration by running the command **`config import-config --comment="Enabling JMX" configuration.xml`**.
   Provide the tools password when prompted.
5. Create a JMX user by running the command **`config create-jmx-user --username=MyJMXUser`**.
6. Provide a password for the user *MyJMXUser*.
   Provide the tools password when prompted.
7. Restart Spotfire Server.
8. Browse to the JDK directory containing the JConsole executable.
   The JConsole executable is in the `bin` directory of the JDK installation, such as `<JAVA_HOME>/bin`.
9. Launch the JConsole application.
10. In the JConsole New Connection dialog, select **Remote Process**, enter the `<hostname>:1099`, and then provide the JMX user name and password.

To view the Spotfire information, click the **MBeans** tab, and then select the **com.spotfire.server** domain.

Services monitoring

You can collect and review information on the services running under Spotfire Server using a variety of tools and resources that are provided with your Spotfire Server installation.

Accessing performance data

If your users report to you that the system is slower than they expect, you can begin investigating the problems by examining the performance tracking tools found in Monitoring & Diagnostics.

Prerequisites

You must have administrative privileges on the Spotfire Server.

You can find the performance data for either Automation Services instances or Web Player instances.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
   - If you select an Automation Services instance, by default, the Diagnostics area shows **Automation Services Diagnostics** in the drop-down list box and a list of the performance counters.
   - If you select a Web Player instance, by default, the Diagnostics area shows **Analyses and Diagnostics** in the drop-down list box, an Information area containing individual data table instances, and a list of the performance counters.
4. Review the potential problems and troubleshooting suggestions described in Performance troubleshooting.

   The performance counters and information list are diagnostic tools to help you determine if the problems are with the CPU, the RAM, or the .NET memory allocation.

Under Performance Counters, the numbers within parentheses in the Value column (as shown below) indicate the change in value since the previous reading.

Total processor %	(-2.1) 0.7
Total average processor %	(-0.6) 0.4
Web Player current processor %	(-1.1) 0.2
Web Player average processor %	(-0.1) 0.2

Web Player analyses information - Overview

You can review information about open Web Player instances in **Monitoring & Diagnostics** > **Instances**. Select the Web Player instance to monitor, and then review the **Overview** tab in the **Information** area.

To access the table, see Accessing performance data.

Click **Refresh** in the **Diagnostics** section to update the list of open analysis.

Overview analysis information

Overview Column head	Description
Title	The title of the analysis. The path of the analysis file is shown in the tooltip.
Instances	The number of open instances of the analysis file.
Average load time	The average time it takes the analysis to load, in seconds.
Execution time	The time spent executing request for the analysis, in seconds. This value is a measure of the CPU load the selected analysis puts on the server.
Total data table size	The total memory size of the data tables in the analysis.
Total data table cells	The total number of cells in the data tables.

Overview Column head	Description
Total data view size	This column is displayed only when **Show document nodes and view sizes** is selected. The total data view size is a measure of the memory required for generating the visualizations of the analysis. The memory required varies depending on the complexity of the data needed for the visualization.
Total document node count	This column is displayed only when **Show document nodes and view sizes** is selected. The total number of document nodes. The document node count is a measure of the complexity of the analysis. More visualizations, pages, columns, filtering schemes, markings, and so on, lead to a higher value. If .NET memory is a problem, it is likely that the analyses that use many more document nodes than the others are an issue.
Idle time	The time elapsed, in seconds, since the last user interaction in the analysis.  For analyses with streaming data, idle time is the number of seconds that the analysis has been open but in a non-visible browser tab.
Scheduled	Displays Yes if the analysis is scheduled for automatic updates.
Running jobs	The total number of currently running internal analysis jobs.

Web Player analysis information - Details

You can review information about open Web Player instances in **Monitoring & Diagnostics** > **Instances**. Select the Web Player instance and then review the **Details** tab in the **Information** area.

To access the table, see Accessing performance data.

 Click **Refresh** in the **Diagnostics** section to update the list of open analysis.

Details column head	Description
Title	The title of the analysis. The path of the analysis file is shown in the tooltip.
User name	The name of the user that uses the analysis
Load time	The loading time (in seconds) for the analysis.
Execution time	The execution time (in seconds) measures the time spent executing request for the analysis. It is a measure of the CPU load the selected analysis puts on the server.
Shared data table size	The memory size of data tables that are shared between instances of the analysis.
Shared data table cells	The number of data table cells shared between instances of the analysis
Private data table size	The memory size of the data tables that are not shared between instances.
Private data table cells	The number of data table cells that are not shared between instances.

Details column head	Description
Shared data view size	The memory size of the data views that are shared between instances of the analysis. Data view size is a measure of the memory required for generating the visualizations of the analysis. The memory required varies depending on the complexity of the data needed for the visualization. This column is displayed only when **Show document nodes and view sizes** is selected.
Private data view size	The memory size of the data views that are not shared between instances. This column is displayed only when **Show document nodes and view sizes** is selected.
Shared document node count	The number of document nodes that are shared between instances of the analysis. The document node count is a measure of the complexity of the analysis. More visualizations, pages, columns, filtering schemes, markings, and so on, lead to a higher value. If .NET memory is a problem, it is likely that the analyses that use many more document nodes than the others are an issue. This column is displayed only when **Show document nodes and view sizes** is selected.
Private document node count	The number of document nodes that are not shared between instances. This column is displayed only when **Show document nodes and view sizes** is selected.
Idle time	The time elapsed, in seconds, since the last user interaction in the analysis. For analyses with streaming data, idle time is the number of seconds that the analysis has been open but in a non-visible browser tab.
Scheduled	Yes if the analysis is scheduled for automatic updates.
Running jobs	The total number of currently running internal analysis jobs.

Web Player service performance counters

When you monitor the instance of a Web Player service, you can review the detailed information provided in the **Performance Counters** area to assess the performance measures of open analyses. All memory values are shown in MB.

To access the table, follow the instructions in Accessing performance data.

- To reset the number of cached queries to external data sources, click **Clear cache for all data connections**.
- To run a full garbage collection twice (to clear memory no in use), click **Run a full GC(2)**. Remember that a full garbage collection may take time and the service will be unresponsive during the running.

For information about using performance counters, see Performance troubleshooting.

Performance Counter	Description
# **.NET Induced GC**	The number of times that an induced GC has been performed. This is .NET Common Language Runtime (CLR) Memory.

Performance Counter	Description
% Time In GC	The percentage of processor time spent in GC, this is .NET CLR Memory.
Active threading jobs	The number of active jobs in graphical tables.
Active threads in thread pool	The number of active threads in thread pool (in .NET).
Available memory	The total MBytes available, based on standard performance counter in the category Memory. If this value is low compared to **Web Player total working memory**, then you might have performance problems related to RAM. See Performance troubleshooting for more information.
Available memory %	Available memory for the Node Manager, as a percentage of total memory.
Avg. disk queue length	The length of the queue for disk input-output. This number should be low.
Current time	The time (in UTC) when the page was updated last time.
Data engine active queries	The number of active data engine queries. The number of active data engine queries. This value should not be far above 0 for very long. Normally, data engine queries do not take very long.
Data engine cache memory	The amount of memory used by the data engine cache. This value can be very high without causing problems because it can be paged out to disk if necessary.
Data engine memory	The amount of memory used by the data engine. This includes all data views and data tables. • If this value is a large portion of **Web Player total working memory**, then you might have performance problems related to RAM. • If this value is only a small portion of the **Webplayer total working memory**, then you might have performance problems related to .NET memory. See Performance troubleshooting for more information.
Data engine paged in memory	The accumulated amount of paged in memory. This value must be much smaller than **Data engine paged out memory**
Data engine paged out memory	The accumulated amount of paged out memory. This value can be high, as long as **Data engine paged in memory** is much smaller.
Data engine queries finished	The number of finished low level data engine queries.
Data engine query cache memory	The amount of memory used for cached calculations in the data engine.
Idle threads in thread pool	The number of idle threads in thread pool (.NET) that are ready to be used.
May be recycled	Depending on settings for `recoverMemory` and the current system status, the service instance may send an event to the server that it may recycle the service instance. For more information on `recoverMemory`, see its entry in Spotfire.Dxp.Worker.Web.config.

Performance Counter	Description
Memory health status	According to configured memory limits, this value is displayed as one of the following: • `0:OK`. Indicates that the instance is under no pressure. • `5:Strained`. Indicates that the instance is under pressure but is not a problem. • `8:Exhausted`. Indicates that the instance is under a higher load, so avoid routing new users to this instance, but current users can keep working in this instance. Users of analyses in scheduled updates can be routed to a service instance with a status of `8: Exhausted`. If you discover that service instances that are used for scheduled updates are often in this state, you should consider adding more service instances to the resource pool. This status is sent to the server to be used for routing decisions. For example, you want to avoid sending many users to service instances that are under a higher load. The limits that determine the health status are configurable for both CPU and memory.
Memory in all .NET heaps	The total MBytes in all .NET heaps, based on .NET CLR Memory.
Network kBytes/sec	The current rate of the network traffic, as measured in kilobytes per second.
Number of shared document nodes	The total number of document nodes that can be shared.
Processor health status	The same as **Memory health status** above, but for CPU load.
Thread pool queue length	The queue length for the thread pool (in .NET).
Total average processor %	The average recent CPU % for the node manager, calculated over 120 seconds by default. (For information about tracking the average percentage of CPU usage for a service, see Monitoring CPU usage by services.)
Total processor %	The total processor usage (not just the web client). (For information about tracking the percentage of CPU usage for a service, see Monitoring CPU usage by services.)
Total thread pool requests finished	The total number of thread pool jobs finished (.NET thread pool).
Web Player analyses under scheduled updates control	The number of analyses added to scheduled updates.
Web Player available temp disk space	The amount of free temporary disk space. This value should never approach 0. If the system runs out of temp disk space, all processing halts and any users accessing the server will get no responses. If the value approaches 0, you must add more temp disk space as soon as possible.
Web Player average processor %	The average processor usage recently. Set the time period in `cpuAverageTimeSpan`, under `performanceCounterLogging`. See Spotfire.Dxp.Worker.Web.config for more information.
Web Player cached documents	The number of cached analyses.
Web Player current processor %	The processor usage for the web client process.
Web Player image render executions	The number of image-render executions. Typically one image corresponds to one visualization.
Web Player number of users	The number of logged in users.

Performance Counter	Description
Web Player open documents	The number of open document instances. (If many users have the same document opened, each copy is counted here.)
Web Player total working memory	The amount of memory used by the web client process. If this value is high compared to **Available memory**, you might have performance problems related to RAM. See Performance troubleshooting for more information.
Web Player accumulated processor time	The total number of CPU seconds consumed by the web client. If this number is consistently high, you might have performance problems related to CPU consumption. See Performance troubleshooting for more information.
Web Player uptime	The number of seconds since the service instance was started.

Automation Services instance performance counters

When you monitor the instance of Automation Services, you can review the detailed information provided in the **Performance Counters** area to assess the performance measures of the service instance. All memory values are shown in MB.

To access the table, see Accessing performance data.

- To reset the number of cached queries to external data sources, click **Clear cache for all data connections**.
- To run a full garbage collection twice (to clear memory no in use), click **Run a full GC(2)**. Remember that a full garbage collection may take time and the service will be unresponsive during the running.

For information about using performance counters, see Performance troubleshooting.

Performance Counter	Description
# .NET Induced GC	The number of times that an induced GC has been performed. This is .NET Common Language Runtime (CLR) Memory.
% Time In GC	The percentage of processor time spent in GC. This is .NET CLR Memory.
Accumulated processor time	The accumulated number of CPU seconds since the service start. If this number is consistently high, you might have performance problems related to CPU consumption. See Performance troubleshooting for more information.
Active threading jobs	The number of active jobs in graphical tables.
Active threads in thread pool	The number of active threads in thread pool (in .NET).
Available memory	The total MBytes available, based on standard performance counter in the category Memory. If this value is low compared to **Total working memory**, then you might have performance problems related to RAM. See Performance troubleshooting for more information.
Available memory %	The memory that is still available, as a percentage of the total.
Available temp disk space	The amount of available disk space allocated as temporary.
Average processor %	The average recent CPU percentage for this service instance, calculated over 120 seconds by default. Set the time period in `cpuAverageTimeSpan`, under `performanceCounterLogging`. See Spotfire.Dxp.Worker.Web.config for more information. (For information about tracking the average percentage of CPU usage for a service, see Monitoring CPU usage by services.)

Performance Counter	Description
Avg. disk queue length	The length of the queue for disk input-output. This number should be low.
Current processor %	The processor usage for the web client process. (For information about tracking the percentage of CPU usage for a service, see Monitoring CPU usage by services.)
Current time	The time (in UTC) when the page was last updated.
Data engine active queries	The number of active data engine queries. This value should not be far above 0 for very long. Normally, data engine queries do not take very long.
Data engine memory	The amount of memory used by the data engine. This includes all data views and data tables. • If this value is a large part of **Total working memory**, then you might have performance problems related to RAM. • If this value is only a small portion of the **Total working memory**, then you might have performance problems related to .NET memory. See Performance troubleshooting for more information.
Data engine paged in memory	The accumulated amount of paged in memory. This value must be much smaller than **Data engine paged out memory**
Data engine paged out memory	The accumulated amount of paged out memory. This value can be high, as long as **Data engine paged in memory** is much smaller.
Data engine queries finished	The number of finished low level data engine queries.
Data engine query cache memory	The amount of memory used by the data engine cache. This value can be very high without causing problems because it can be paged out to disk if necessary.
Idle threads in thread pool	The number of idle threads in thread pool (.NET) that are ready to be used.
Image render executions	The number of image render executions. Typically one image corresponds to one visualization.
May be recycled	Depending on settings for `recoverMemory` and the current system status, the service instance may send an event to the server that it may recycle the service instance. For more information on `recoverMemory`, see its entry in Spotfire.Dxp.Worker.Web.config.
Memory health status	According to configured memory limits, this value is displayed as one of the following: • `0:OK`. Indicates that the instance is under no pressure. • `5:Strained`. Indicates that the instance is under pressure but is not a problem. • `8:Exhausted`. Indicates that the instance is under a higher load, so avoid routing new users to this instance, but current users can keep working in this instance. Users of analyses in scheduled updates can be routed to a service instance with a status of `8: Exhausted`. If you discover that service instances that are used for scheduled updates are often in this state, you should consider adding more service instances to the resource pool. The health status is sent to the server to be used for routing decisions. For example, you want to avoid sending many users to service instances that are under a higher load. The limits that determine the health status are configurable for both CPU and memory.
Memory in all .NET heaps	The total MBytes in all .NET heaps, based on .NET CLR Memory.
Network kBytes/sec	The current rate of the network traffic, as measured in kilobytes per second.

Performance Counter	Description
Number of users	The number of logged in users.
Processor health status	The same as **Memory health status** above, but for the CPU load.
Thread pool queue length	The queue length for the thread pool (in .NET).
Total average processor %	The average recent CPU percentage for the node manager, calculated over 120 seconds by default. (For information about tracking the average percentage of CPU usage for a service, see Monitoring CPU usage by services.)
Total processor %	The total processor usage. (For information about tracking the percentage of CPU usage for a service, see Monitoring CPU usage by services.)
Total thread pool requests finished	The total number of thread pool jobs finished (.NET thread pool).
Total working memory	The amount of memory used by the web client process. If this value is high compared to **Available memory**, you might have performance problems related to RAM. See Performance troubleshooting for more information.
Uptime	The number of seconds since the service instance was started.

Performance troubleshooting

Your users might report that the system is much slower than they expect. System slowdowns can result from one or multiple problems, including system resources and memory. The tools found in Monitoring & Diagnostics can help you track down these types of problems.

By analyzing the problems described in this topic, you can collect information about which analyses are consuming system resources and memory.

Not all performance problems can be traced to the performance issues reported in Monitoring & Diagnostics. If you do not discover the source of the performance problem through the performance counters, you might need to investigate problems with connectivity, network speed, or other external problems.

To find the performance counters and analysis statistical information, follow the instructions in Accessing performance data.

- If a Web Player instance indicates high consumption of resources, you can review the Analysis information to determine if you have problem analyses causing these issues.
- If an Automation Services instance indicates a high consumption of resources, then review the running analysis for usage information.

You can get additional statistics for a single analysis in the desktop client. You can discover which of its pages or visualizations use most of the resources. See Examining the statistics of an individual analysis for more information.

- For information about troubleshooting the TERR service, see TIBCO® Enterprise Runtime for R - Server Edition.
- For information about troubleshooting Spotfire Service for Python, see TIBCO® Spotfire Service for Python.

1. In the list of Performance counters, find the entry for **Accumulated processor %** (or **Web Player average processor %** for a Web Player instance). Monitor it for a few minutes.

   - If the entry for **Accumulated processor %** (or **Web Player accumulated processor %**) is consistently high, then you have problems with CPU consumption. For a Web Player instance, in the Information area, click the **Overview** tab and review the **Average load time** and **Execution Time** columns. The analyses with the highest values are consuming the most CPU.
   - If the entry for **Web Player accumulated processor** is not consistently high and varies, and your performance problems continue, then check for problems with RAM and .NET memory.

2. To check for problems with RAM or .NET memory, in the list of performance counters, review the following values for the following conditions.

   - The value for **Total working memory** (or **Web Player total working memory** for a Web Player instance) is high, and the value for **Available memory** is low.
   - The value for **Data Engine memory** is a large portion of the value for **Total working memory** (or **Webplayer total working memory**).

   If these conditions exist, then memory consumption is the issue. For a Web Player instance, in the Information area, click the **Overview** tab and examine the list of data table instances. The values listed for the columns **Total data table size** and **Total data view size** indicate which analyses are holding the most data table and view memory.

   - If an analysis has a large value for **Total data table size**, then the amount of raw data can cause problems. Check the analysis to see if it includes any tables or columns that are not used. If all tables and columns are used, then you need to install more RAM in the Spotfire Server computer.
   - If an analysis has a high value for **Total data view size**, or if it appears that the number of document nodes is high, the analysis might be too complicated.

   Unused tables, columns, pages, and visualizations generate more document nodes and use data engine memory. However, unused data engine memory can be paged out to disk when available memory becomes low.

3. To check for additional problems with .NET memory, in the list of performance counters, review the entry for **Memory in all .NET heaps**. Click **Run a full GC(2)** twice. This action gives the system a chance to reclaim memory that is released.

   For a Web Player instance, in the Information area, click the **Overview** tab and review the **Document Node Count** column. Document nodes are more complicated because they can be different sizes. Analyses that use many more document nodes than the others can cause problems with .NET memory.

   Try to perform this action when the server is not very busy, because the system can be unresponsive while running the GC action.

Examining the statistics of an individual analysis

If you have problems with performance of a server, and you suspect one or more analyses of causing the problems on the server, you can examine the suspicious analyses individually using Spotfire Analyst.

See Performance troubleshooting for advice for identifying any analyses run from the Web Player or through Automation Services that might be causing problems with resource consumption.

Prerequisites

You must have log in credentials for the Spotfire Server for which you want to load the analysis and examine its performance data.

Procedure

1. Log in to Spotfire Analyst and load the analysis to examine.
2. On the menu, click **Help** > **Support Diagnostics and Logging**.
3. Click the **Diagnostics Information** tab.
   Detailed usage information for the analysis is displayed on this tab.

What to do next

Temporarily removing pages, plots, or tables, and then re-examining the resource usage data can provide more insight for troubleshooting, including whether to increase system resources or recommend changes to the analysis.

Logging and exporting monitoring diagnostics

Monitoring diagnostics can be logged, and the logged results can be exported as a Spotfire analysis file that displays the information in the log files.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. On the **Instances** page, under **Network diagnostics**, click the instance for which you want to log and export monitoring diagnostics.
3. Under **Diagnostics**, in the left drop-down list, select the default diagnostics option.
   - For Web Player instances, this option is **Analyses and Diagnostics**.
   - For Automation Services instances, this option is **Automation Services Diagnostics**.
4. In the **Logging** drop-down list to the right, select one of the following options.

Option	Description
Enable Monitoring Logging	Start logging to the logs needed for the monitoring analysis on debug level.
Enable Full Monitoring Logging	Start logging, with enabled performance diagnostics, to the logs needed for the monitoring analysis on debug level. This monitoring level is extremely verbose, so do not set this option unless asked to do so by Spotfire Support. After collecting the necessary information from this level, reset logging by selecting Restore Monitoring Logging or by restarting the service instance.
Restore Monitoring Logging	Restore logging levels to what is specified in the `log4net.config` file.
Export Monitoring Logs and Analysis	Export a snapshot of the log files together with the Spotfire analysis file used to analyze them.  In Spotfire, the Missing File dialog may open. Do the following: a. Select the **Apply to all missing files in the analysis** check box. b. Click **OK**. c. In the Match Columns dialog that opens, click **OK**.
Export Monitoring Analysis	Export the monitoring analysis file without the logs. Use this if the logs have been copied in another way.
Export Information	Export diagnostics information to a text file.

Result

Any specified monitoring logs are written to the directory `<installation directory>/nm/logs`.

Viewing node information

You can gather information about a specific node to analyze its available resources and check version details. This information is useful for troubleshooting and working with Spotfire Support.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
4. Under Diagnostics, in the left drop-down list box, select **Node**.

Result

The information about the selected node is displayed.

What to do next

For the node's CPU usage detail, see Monitoring CPU usage by nodes.

Viewing service configuration information

You can gather information about the service configuration for a node. This information is useful for troubleshooting configuration problems.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
4. Under Diagnostics, in the left drop-down list box, select **Service Configuration**.

Result

The configurations and settings that are specified in the `Spotfire.Dxp.Worker.Web.config` file of the service are listed.

Monitoring CPU usage by nodes

If you have performance problems, and you expect the CPU usage is an issue, you can monitor the usage for nodes and instances. The Spotfire Server Monitoring & Diagnostics tools provide this information. This topic covers monitoring for nodes.

- To monitor CPU usage by nodes, examine the Node area in Monitoring & Diagnostics.
- To monitor CPU usage by instances, examine the performance counters for Web Player instances or Automation Services instances.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. From Monitoring & Diagnostics, click the **Overview** tab.
   The information page for Spotfire Server, Nodes, and Web Player instances is displayed.
3. In the Nodes area, look for the column **CPU usage(Avg)**.
   The value outside of the parenthesis indicates the percentage of the CPU the node for that row is using. The value inside the parenthesis specifies the average CPU usage for that node.

What to do next

For general diagnostic information about an instance's node, see Viewing node information.

Monitoring CPU usage by instances

If you have performance problems, and you expect the CPU usage is an issue, you can monitor the usage for instances. The Spotfire Server Monitoring & Diagnostics tools provide this information.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. From Monitoring & Diagnostics, click the **Overview** tab.
   The information page for Spotfire Server, Nodes, and service instances is displayed.
3. In the service instances area, look for the column **CPU usage (Avg)**.
   The value outside of the parenthesis indicates the percentage of the CPU that the service instance identified in that row is using. The value inside the parenthesis specifies the average CPU usage for that service instance.

What to do next

For general diagnostic information about an instance's node, see Accessing performance data and Viewing service configuration information.

Viewing assemblies information

You can gather information about the assemblies that are loaded by a specific service. This information is useful for troubleshooting and working with Spotfire Support.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.

2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
4. Under Diagnostics, in the left drop-down list box, select **Loaded Assemblies**.

Result

The complete list of assemblies for the service is displayed.

Viewing website information

You can gather information about current activity on the website for a specific service.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the **Instances** tab.
3. Under Network Diagnostics, select the Automation Services instance or Web Player instance to review.
4. Under Diagnostics, in the left drop-down list box, select **Website**.

Result

The Website diagnostics are displayed.

Website diagnostics

The Website diagnostics provide you with details about the current activity of the selected service instance.

Name	Description
Uptime	How long the Web Player service has been running.
Concurrent users	The number of currently logged in users. The number in parentheses indicate the maximum number of concurrent users that is being measured during this uptime.
Number of cached queries for data connections	The number of cached queries to external data sources. This can be reset by clicking **Clear cache for all data connections**, see Web Player Service Performance Counters.
Cached analyses	The number of currently cached analyses. The number in parentheses indicate the maximum number of analyses that is being measured during this uptime.
Open analyses	The number of currently open analyses.

Name	Description
Current sessions	Lists the currently-active sessions. **Current sessions** includes the following information. • User name(s). • The number of open analyses. The number in parentheses indicate the maximum number of analyses that is being measured during this uptime. • The session ID. • The IP number of the client. • The browser used. • The time the session started.
Current analyses	Lists the currently-open analyses and which users are accessing them. **Current analyses** includes the following information. • The path to the efile. • The time the file was opened. • The analysis ID. • Any pending HTTP requests. • The time since the last ping. • The idle time of the analysis.

Viewing routing

You can get overviews of the routing, such as which instances are used for the different resource pools. You can get this information from both analyses and instances perspectives.

Prerequisites

You must have administrative privileges on the Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Click the routing tab to display the information you need.

Option	Description
Routing: Analyses	Displays a list of analyses that are currently active on the server, including the analysis path, the number of users, and the number of instances. • To view more information about analysis routing, including the instance and the resource pool click the arrow next to the analysis name.
Routing: Instances	Displays a list of Web Player instances currently active on the server, including the resource pools and number of users. • To view more information about the routing, click the arrow next to the instance name. • To view more information about the instance, click its name.

Clicking an instance name from either the Analysis area or the Instance area displays the instance information in Nodes & Services.

Enabling automatic dump capture from non-responsive Web Players

To capture diagnostic information from Spotfire Web Players that stop responding, set up the automatic dump capture.

Procedure

1. On each computer that is running a node manager with the Spotfire Web Player service, download and install the Microsoft Debugging Tools for Windows (WinDbg). This toolkit is available from the following website: https://developer.microsoft.com/en-us/windows/hardware/windows-driver-kit.
2. On the server computer, export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
3. Using the set-config-prop command, set the `nodemanager.memorydump-after-failures` property to an integer greater than 0. This sets the interval after which the memory dump will occur.

   Values for the Web Player auto-dump feature

Value	Description
-1	The Spotfire Web Player automatic dump feature is turned off.
1	The memory dump occurs one interval after the Spotfire Server determines that a service is unreachable. The server performs ten verification steps, so this would cause the dump to occur after 11 failures to communicate with the service.
2	The memory dump occurs two intervals after the Spotfire Server determines that a service is unreachable. This would cause the dump to occur after 12 failures to communicate with the service.

   The values continue to increase in the same way.

   For a large system, you may want to set a high value because the process may be unresponsive for some time due to blocking garbage collection.

   Example:

   ```
   config set-config-prop --name nodemanager.memorydump-after-failures --value 5
   ```

4. Import the configuration back into the database by using the import-config command.
5. Do the following on the server computer that you accessed in step 2 above:

   a. Export and open the `Spotfire.Dxp.Worker.Web.config` file for editing; for instructions, see Manually editing the service configuration files.

   b. In `Spotfire.Dxp.Worker.Web.config`, locate the following section:

   ```
   <errorReporting
     emailAddress="" maxMailLength="1000"
     includeDetailedErrorInformation="false"
     enabledMiniDumpCreationOnError="true"
     miniDumpPath=""
     miniDumpSizeLarge="false"
     dumpToolPath ="C:\Program Files (x86)\Windows Kits\10\Debuggers\x64\cdb.exe"
     dumpToolFlagsSmall="-c &amp;quot;.dump /mhtpFidcu {0};q&amp;quot; -p {1}"
     dumpToolFlagsLarge="-c &amp;quot;.dump /ma {0};q&amp;quot; -p {1}"/>
   ```

   c. Set the `dumpToolPath` to match the location of the `cdb.exe` file that you installed in step 1.

   d. (Optional) To configure flags, see the descriptions of the following settings in Spotfire.Dxp.Worker.Web.config: `dumpToolFlagsSmall`, `dumpToolFlagsLarge`, and `miniDumpSizeLarge`.

   e. Save the file and then import it back to the server by using the import-service-config command.

f. Assign the updated configuration to the services by using the set-service-config command.

6. Restart the server.

Result

If a Spotfire Web Player becomes non-responsive, a dump file with the name hanging_process_dump_*ServiceInstanceID*_pid*XX*.dmp will be created in the `C:\tibco\tsnm\`*`version number`*`\nm\logs` directory of the node manager computer.

Basic troubleshooting

Spotfire Server provides tools to troubleshoot if you encounter problems in your installation and configuration.

Troubleshooting Spotfire Server

Before diving deeply into logs or contacting support, you can perform some basic steps to check where problems might exist.

Prerequisites

You must have administrative access to Spotfire Server.

Procedure

1. Make sure that Spotfire Server has network connectivity.
2. Make sure that the Spotfire Server service is up and running.
   If a custom user account is used to run the Spotfire Server service, ensure that the account credentials are valid and not locked.
3. Verify that there are no port conflicts with the Spotfire Server ports.
4. Verify that the Spotfire Server administration interface can be accessed outside of the Spotfire Server computer.
   If it works correctly on the server machine but is not accessible outside the server, make sure that there is no firewall or proxy blocking the server access.

What to do next

If none of these steps solve the problem with Spotfire Server, review all of the logs and consider creating a troubleshooting bundle for Spotfire Support to analyze. See also Contacting support.

Spotfire Server fails to start

If the Spotfire Server fails to start, check the log for the error described in this topic.

```
Error initializing the Spotfire web application. Please
contact the server administrator.
```

The following errors are captured in the server logs.

```
SEVERE: Catalina.start
LifecycleException: service.getName(): "Spotfire"; Protocol
 handler start failed: java.net.BindException: Address already in
 use: JVM_Bind <null>:
```

Cause

This is an indication of a port conflict.

Resolution

You can check if any of the Spotfire Server ports are blocked by other processes on the Spotfire Server machine. Either stop those services so that Spotfire Server can grab these ports or assign a different port by modifying the `server.xml` file located under `\tomcat\conf` folder.

Spotfire Server runs out of JVM memory

If the Spotfire Server runs out of JVM memory, Spotfire Server can fail or stop responding. This failure can make new connections impossible, and opening any files can fail.

The following errors are captured in the server logs.

```
Caused by: java.lang.OutOfMemoryError: GC overhead limit exceeded
......
SEVERE: Exception invoking periodic operation:
java.lang.OutOfMemoryError: Java heap space
```

Cause

This exception indicates that you are out of memory. It is thrown by the garbage collector in the underlying Java and is not specific to Spotfire.

Resolution

You must add more memory. See Virtual memory modification for more information.

Users cannot log in

Two conditions can cause users to not be able to log into Spotfire Server. The causes and resolutions for these problems are described in this topic.

In both conditions, users are not able to log in to Spotfire Analyst or Spotfire Business Author. Administrators can fail to log into the Spotfire Server administration interface. Both of these conditions result in LDAP errors being generated. You can find the error codes in the server logs.

LDAP error codes	Cause	Resolutions
`javax.naming.AuthenticationException: [LDAP: error code 49 - 80090308: LdapErr: DSID-0C090334, comment: AcceptSecurityContext error, data 52e, vece ]`	This LDAP error code indicates that the log in credentials used for LDAP binding are invalid. This can happen if the password of the LDAP Service Account is expired.	Modify the LDAP configuration with the updated credentials.
`javax.naming.AuthenticationException: [LDAP: error code 49 - 80090308: LdapErr: DSID-0C0903A9, comment: AcceptSecurityContext error, data 533, v1db1 ]`	This LDAP error code indicates that the Service Account that is used for LDAP binding can be locked out or disabled.	Enable the Service Account and then try again.

Troubleshooting the Spotfire database

Before diving deeply into logs or contacting support, you can perform some basic steps to check where problems might exist.

Prerequisites

You must have administrative access to Spotfire Server and the Spotfire database.

Procedure

1. Make sure that the Spotfire database is up and running.
2. Validate the database credentials specified in the `bootstrap.xml` file.

   Ensure that the database user has access to all the required Spotfire database tables and procedures. That is, if you log in to the Spotfire Server database with those credentials, you should be able to browse and access all the contents of the Spotfire database.
3. Make sure there is communication between the Spotfire Server computer and the Spotfire database server.

   For example, ping the database server from Spotfire Server.

What to do next

If none of these steps solve the problem with the Spotfire database, review the Spotfire Server logs or see Contacting support.

Creating a thread dump

Creating thread dumps can be useful. For example, you can use a thread dump to examine problems with servers that appear to be unresponsive, or to investigate why the server is taking an unusual amount of time to respond.

To help troubleshoot such cases, Spotfire Support can examine a dump of thread activity to help determine what is happening. When the Spotfire Server is running as a Windows service, it can be complicated to create this thread dump. This topic describes a simple way to create a thread dump.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Select the Spotfire Server for which you want to download the dump.
3. Click the **More** button (...), and from the resulting drop-down list, click **Download thread dump**. The thread dump is written to a text file and downloaded to the computer.
4. In the server file system where the Spotfire Server is installed, browse to the directory where the text file was written.

   The text file name follows the convention `threadDump-<guid>.txt`.

Result

You can open the text file and review the results, and you can share the thread dump with Spotfire Support.

Memory exhaustion

An exhausted memory usually shows an out-of-memory exception in the log. If you are using Microsoft SQL Server, it can manifest itself as a deadlock.

First, try to increase the amount of memory available to the server. For more information, see Virtual memory modification.

If increasing the memory for the server does not solve the problem, you can contact Spotfire Support. Spotfire Support might want to get a dump of the memory to investigate memory leaks. See Creating a memory dump for instructions.

If your organization handles sensitive information that should not be exposed in a memory dump, you might need to disable this feature. For more information, see Disabling the memory dump feature.

Creating a memory dump

You can create a memory dump to examine problems with exhausted memory.

Perform this task from the Administration interface, and from the file system of the server where Spotfire Server is installed.

When a memory dump is created, the Java Virtual Machine halts for a short period.

Prerequisites

- You must be a member of the Administrator group. It is not sufficient to be only a member of the Diagnostics Administrator group.
- You must have write access to the server's file system where Spotfire Server is installed.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. Select the server for which to create the memory dump.
3. On the right end of the row, click the **More** button (...), and then select **Create memory dump**.

   Because memory dumps contain the entire state of the running server, they can contain sensitive information. Therefore, you must prove that you have access to the server.

   You are prompted to create a "proof file" in a specific location and with a specific name, and then to return to the Administration interface to resubmit your request.
4. In the server file system where the Spotfire Server is installed, create the specified proof file.

   The file does not need to contain content; it merely demonstrates that you are an Administrator with write access to the file system on the server. The memory dump cannot proceed until the file exists.

   A new name is generated every time the server is restarted or when a memory dump is made.
5. After you create the proof file as instructed, return to the Administration interface.

   The name of the proof file should appear on the page.
6. Click Refresh, and then repeat Step 3.

   - A memory dump file is created. This process can take some time.
   - Any previous dump file is overwritten.
   - When it is completed, the path to the file in the server's file system is displayed.

7. Return to the server file system to retrieve the file.
   There is no download functionality on the page.
8. After you have analyzed and solved the memory problem, delete the dump file.
   The dump file can contain sensitive information.

On normal termination of the server, the generated heap dump file is deleted automatically.

Disabling the memory dump feature

Because a memory dump can contain sensitive information, you might need to configure the Spotfire Server to never create this artifact.

Perform this task in the file `configuration.xml`, exported from the Spotfire Server.

Prerequisites

- You must have credentials to export, edit, and import the configuration file for Spotfire Server.
- You must export the file `configuration.xml` for editing. See Manually editing the Spotfire Server configuration file for more information.

Procedure

1. In the file `configuration.xml`, create a new node as follows.

```
<tools>
           <enable-memory-dump>
              <enabled>false</enabled>
           </enable-memory-dump>
        </tools>
```

2. Save and close the file.
3. Follow the steps for importing the file to the server and then restarting the service.
   See Manually editing the Spotfire Server configuration file.

Result

The new imported configuration becomes the active configuration for that server or cluster.

Creating a troubleshooting bundle

You can create a zip archive of different types of logging information. This information can help Spotfire Support assist you with troubleshooting Spotfire Server.

Prerequisites

You must have administrative credentials for Spotfire Server.

Procedure

1. Log in to Spotfire Server and then click **Monitoring & Diagnostics**.
2. On the Overview page, select the Spotfire Server for which you want to create the troubleshooting bundle.
3. Click **Download troubleshooting bundle**.
   A warning dialog is shown advising you that this process can take several minutes.
4. In the Download server troubleshooting bundle dialog, click **OK** to continue.
   The troubleshooting bundle is written to a zip archive and downloaded to the server file system.

5. In the server file system where the Spotfire Server is installed, browse to the directory where the zip archive was written.

   The zip archive can contain some or all of the following information.

   - The entire logs directory.
   - A thread dump.
   - The results of diagnostics.
   - The full configuration history (but not the actual configurations).
   - A list of all server startup and shutdown events.
   - A list of all nodes in the collective.
   - A list of all certificates issued by the internal CA.

What to do next

Contact Spotfire Support for instructions on sharing the troubleshooting bundle.

Automation Services job scheduling

In the Automation Services area of the Spotfire administration interface, you can schedule Automation Services jobs to run periodically. Then you can monitor and troubleshoot the resulting job executions.

Automation Services jobs are automated procedures that carry out multi-step tasks. For example, an Automation Services job could be used to open and update a sales analysis, and then email specific sections of the analysis to specific managers. Automation Services jobs are created in Spotfire Analyst by using the Job Builder, and are saved as XML files. For details, see the Automation Services User's Guide.

After an Automation Services job file is created and then saved in the Spotfire library, any member of the Automation Services Users group who has the necessary library permissions can schedule the job. The Automation Services user selects the Automation Services job file and specifies the day or days of the week, and the time, that the job should run. This creates the "scheduled job". Scheduled jobs can contain multiple schedules, but each Automation Services job file can be associated with only one scheduled job.

Alternatively, you can use the **`create-scheduled-jobs`** command to create schedules from a JSON file. For details, see create-scheduled-jobs.

By default a new scheduled job is enabled and will run at the next scheduled time. You can edit the scheduled days and times, and switch the schedules to a different Automation Services job. You can also disable the scheduled job to temporarily stop its activity.

Deleting the scheduled job will permanently remove the schedule from Spotfire. The Automation Services job that was associated with that schedule is then available for new scheduling.

Moving an Automation Services job file from one folder in the Spotfire library to another does not affect the scheduled job, which continues to run as scheduled. Deleting an Automation Services job from the library disables (but does not delete) its schedule, which you can then attach to a different Automation Services job, or delete altogether.

Existing scheduled jobs that were scheduled using the Client Job Sender in Spotfire Analyst, or the API, will continue to work. These job executions will be visible on the Activity pages along with the jobs that were scheduled in the administration interface. For more information, see Automation Services API.

Scheduling Automation Services jobs

After an Automation Services job is created in Spotfire Analyst, you can schedule the job in the Spotfire administration interface.

Prerequisites

- You must belong to the Automation Services Users group.
- The Automation Services job that you want to schedule must be saved in the Spotfire library.
- You must have Browse + Access permission for the library folder that contains the Automation Services job.

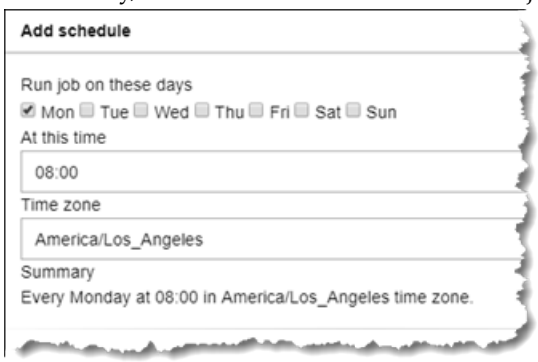
For general information on this feature, see Automation Services job scheduling.

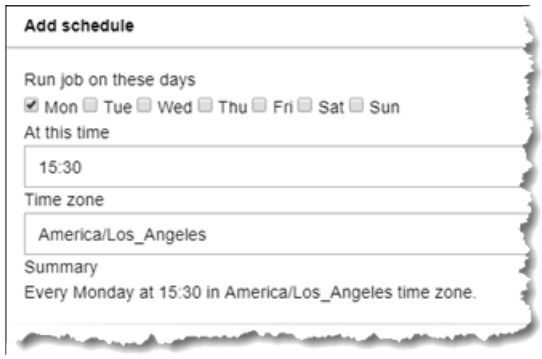
Procedure

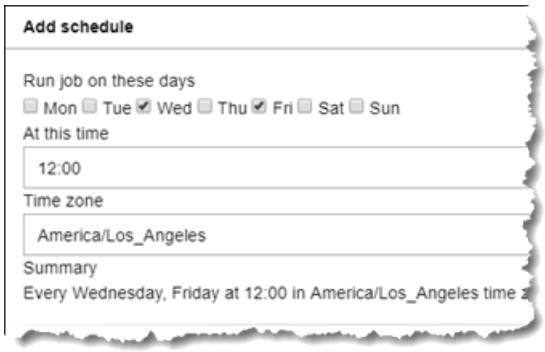
1. Do one of the following:
   - Follow these steps:
     a. Log in to Spotfire Server and click **Automation Services**.
     b. On the Scheduled jobs page, click **Schedule a job**.
     c. On the Schedule a job page, next to the **Job path** field, click **Browse** and select the Automation Services job that you want to schedule.
   - In the Automation Services Job Builder, open the job you want to schedule, and then click **Tools** > **Manage Job Schedule**.
2. Optional: On the Schedule a job page, in the **Scheduled job name** field, you can edit the name of the scheduled job. By default the scheduled job name is the same as the Automation Services job name.
3. Click **Add schedule** and then select the day(s) of the week and the time that you want the job to run. If necessary, change the time zone.

   You can enter only one time in the Add schedule dialog. If you want to schedule the job to run twice (or more) on the same day, add additional schedules for the job.

   Example If you want a job to run at 8:00 AM and 3:30 PM on Monday, and at noon on Wednesday and Friday, add these three schedules to the job:







4. When you have finished adding schedules, click **Save**.

Result

The new scheduled job appears in the **Scheduled jobs** list, and will run on the scheduled day at the scheduled time.

The Next run information for the job appears in the **Scheduled jobs** list after you visit a different page of the interface.

Automation Services activity

The activity pages of the Automation Services area present an overview of all job activity, as well as links to other pages and logs that provide detailed information on job executions.

Library job activity

The "Library job activity" page displays the latest execution for Automation Services jobs that are saved in the Spotfire library.

These job executions may have been initiated in any of the following ways:

- By a scheduled job that was created in the Spotfire administration interface.
- By a user running the scheduled job, outside of its schedule, from the "Scheduled jobs" page.
- By a user running the job from the Automation Services Job Builder or Client Job Sender.
- By using the `create-scheduled jobs` command.
- By a schedule that was created in Windows Task Scheduler for running the job on the Client Job Sender.
- By using the public API.

For information about the Job Builder or Client Job Sender, see the Automation Services User's Guide.

To view previous executions of a scheduled job, click in the job's row. This displays the job's History page.

By default job activity older than a week is deleted. If your organization runs many Automation Services jobs, you may want to reduce the default time period for deleting job history items from the database; for instructions, see Changing how often the scheduled job history is cleared. You may want to consider other options for tracking history, such as action logging; for more information, see Action logs and system monitoring. You can also clear individual job activities or large numbers of them by using the **Actions** list near the top of the page.

You can use the **Initiated by me** quick filter to view only activities that you scheduled or initiated directly. In the case of a scheduled job that was created in the Spotfire administration interface, an activity will appear in this filtered view if one of the following is true:

- You scheduled the activity in the Spotfire administration interface and no other user has been the last to edit the scheduled job.
- You were the last person to edit a scheduled job that was created in the Spotfire administration interface (by any user).

You can apply additional quick filters, such as **Failed jobs**, to the job activity list by clicking the filter names.

Local job activity

The "Local job activity" page displays the latest execution for Automation Services jobs that are not saved in the Spotfire library.

Because jobs that are not in the Spotfire library cannot be scheduled in the Spotfire administration interface, the job activity that appears on this page was initiated in one of the following ways:

- By a user running the job from the Automation Services Job Builder or Client Job Sender.
- By a schedule that was created in Windows Task Scheduler for running the job on the Client Job Sender.
- By using the public API.

To reduce the number of activities that currently appear in the list, you can use the **Actions** list near the top of the page.

You can use the **Initiated by me** quick filter to view only those activities that you scheduled or initiated directly.

You can apply additional quick filters, such as **Failed jobs**, to the job activity list by clicking the filter names.

Available information for troubleshooting job activity

The Automation Services area of Spotfire Server provides a variety of information and links to help you understand any job execution issues.

To access the following information options, click the more info icon () that appears when you hover to the right of a job activity row.

- To view the Properties dialog of an activity, click **Properties**. The left column of the dialog provides information about the scheduled job and the right column provides information about the Automation Services job file.

  The **Scheduled job name** field does not apply to activities that are listed on the "Local job activity" page.
- To view related logs, click **View logs**. The "Log files" page identifies the service instance and node on which the activity occurred. Clicking the instance or node name opens the Nodes & Services page with that component highlighted.
- To view the activity history for a job, click **History**.

  Activity history is not available for Automation Services jobs that are not in the Spotfire library.

Detailed information about the performance of Automation Services instances is available in Monitoring & Diagnostics on the Overview and Instances pages.

Editing scheduled Automation Services jobs

You can edit the name and schedule of scheduled Automation Services jobs, or apply the current schedules to a different Automation Services job.

For general information on this feature, see Automation Services job scheduling.

Procedure

1. Do one of the following:
   - Follow these steps:
     a. Log in to Spotfire Server and click **Automation Services**.
     b. On the Scheduled jobs page, select the check box to the left of the scheduled job that you want to edit.
     c. Near the top of the page, click **Actions** > **Edit**.
   - In the Automation Services Job Builder (in Spotfire Analyst), open the job you want to schedule, and then click **Tools** > **Manage Job Schedule**. This will take you to the server web interface. Log in if needed.
2. On the 'Edit scheduled job' page, perform any of the following tasks:
   - To edit a schedule, hover with the mouse pointer over the schedule line and then click the **Edit** icon to the right.
   - To delete a schedule line, hover with the mouse pointer over the line and then click the **Trash** icon to the right.
   - To add a schedule, click **Add schedule**.
   - To change the name of the scheduled job, edit the **Scheduled job name** field.
   - To switch the Automation Services job file to which the schedules apply, next to the **Job path** field, click **Browse** and select the job file to use.
3. When you have finished editing the scheduled job, click **Save**.

Running a scheduled Automation Services job outside of its schedule

You can initiate a non-scheduled execution of a scheduled Automation Services job from the administration interface.

You can also initiate a non-scheduled job execution from the Automation Services Job Builder and the Client Job Sender (both in Spotfire Analyst), and by using the public API.

For information about running jobs in Spotfire Analyst, see the Automation Services User's Guide.

For general information on scheduling jobs in the administration interface, see Automation Services job scheduling.

Prerequisites

The scheduled job must be enabled.

Procedure

1. Log in to Spotfire Server and click **Automation Services**.
2. On the Scheduled jobs page, select the check box to the left of the scheduled job that you want to run.
3. Near the top of the page, click **Actions** > **Run**.
   Then you can go to the Library job activity page where you can see the result of your action.

Disabling or deleting scheduled Automation Services jobs

You can disable a scheduled job to pause its activity, or delete the scheduled job to permanently stop the activity and remove its schedule from Spotfire.

For general information on this feature, see Automation Services job scheduling.

Procedure

1. Log in to Spotfire Server and click **Automation Services**.
2. On the Scheduled jobs page, select the check box to the left of the scheduled job that you want to disable or delete.
3. Near the top of the page, click **Actions** > **Disable** or **Actions** > **Delete**.

Result

- To resume the activity of a disabled scheduled job, re-enable the scheduled job.
- To resume scheduled activity for an Automation Services job whose schedule was deleted, you must create a new scheduled job; see Scheduling Automation Services jobs.

Command-based library administration tasks

Most library administration tasks are performed in Spotfire Analyst. These include structuring the library and its contents, and setting permissions for library folders. The tasks listed here either can be performed only in Spotfire Server, or can be performed in the server (as well as in Spotfire Analyst) for administrators who prefer using the command line.

For information about library administration in Spotfire Analyst, see the Spotfire Analyst User's Guide.

Importing library content by using the command line

Instead of using the Library Administration tool in Spotfire Analyst, you can import content to the library by using the command line.

Prerequisites

You must have administrative credentials for Spotfire Server.

For general information about library administration, see the Spotfire Analyst User's Guide.

Procedure

1. Open a command line as an administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
2. On the command line, enter the **`import-library-content`** command, specifying the options needed to import the `.zip` file.

   Example:

   ```
   config import-library-content --tool-password=password --file-path=/path/to/folder/
   content.part0.zip --conflict-resolutionmode=KEEP_BOTH --user=jdoe --library-path=/
   ```

   For more information, see import-library-content.

Result

The progress of the import is displayed on the command line.

Exporting library content by using the command line

Instead of using the Library Administration tool in Spotfire Analyst, you can export content from the library by using the command line.

Prerequisites

You must have administrative credentials for Spotfire Server.

For general information about library administration, see the Spotfire Analyst User's Guide.

Procedure

1. Open a command line as an administrator and go to the `<server installation dir>/tomcat/spotfire-bin` directory.
2. On the command line, enter the **`export-library-content`** command, specifying the options needed to import the ZIP file.

   Example:

   ```
   config export-library-content --tool-password=password --file-path=C:/YearEndAnalyses --
   user=jdoe --item-type=analysis_files --library-path=/Finals/Europe
   ```

   For more information, see export-library-content.

Result

The progress of the export is displayed on the command line.

The exported folder and its contents are saved as a ZIP file. The exported items are not removed from the library.

Library content storage outside of the Spotfire database

To minimize the size of your Spotfire database, you can store your organization's Spotfire library content (analyses and analysis data) in the cloud using Amazon Web Services S3 (AWS), or in a file system elsewhere.

In a typical Spotfire installation, the largest part of database storage consists of the library content. When you move the library content to external storage, only the metadata about the library files remains in the database. The other items in database storage (system configuration data, permissions, licenses, and so on) remain where they are.

In this scenario, *all* library content is stored externally; it isn't possible to split storage between the server database and the external site.

Currently there are three main drawbacks to this option:

- Referential integrity is not guaranteed; there is the possibility that content referenced in the Spotfire database will not exist in external storage, and vice versa.
- Your system may run more slowly, such as when loading files.
- A database backup will not back up the library content.

Configuring external library storage in AWS

You can configure external library storage in the cloud using Amazon Web Services S3 (AWS).

Prerequisites

- You must have administrative credentials for Spotfire Server.

- You must have an Amazon S3 account.
- You must have a bucket name. Every server database (or database cluster) should have its own bucket. (Items stored in S3 are identified by their GUIDs. If different servers use the same bucket, importing files to Cluster B—when the files already exist in Cluster A—will overwrite the files in Cluster A.)

Procedure

1. Back up the database.
2. On the command line, export the library using the export-library-content command. (For details on using the Spotfire command line, see Executing commands on the command line.)
3. Remove the content from the library.

   Do not use the `truncate` command in the database because there are hidden folders that should not be removed.
4. To enable external storage and select the type of external storage, use the command config-library-external-data-storage.
5. To configure AWS storage, use the command config-library-external-s3-storage.

   You can set the following options when using this command:
   - Which AWS regional datacenter the server should connect to.
   - Whether large files should be uploaded in chunks, and the details of this behavior.
6. Import the library using the import-library-content command.

   The external library storage system uses the Spotfire library globally unique identifiers (GUIDs) to identify files.

   For information on monitoring the external system, see Monitoring external library storage and fixing inconsistencies.

Configuring external library storage in a file system

You can configure external library storage in a file system by using the command line.

Prerequisites

You must have administrative credentials for Spotfire Server.

In addition to the information in Library content storage outside of the Spotfire database on page 346, you should be aware that the file system operations have no transactions, which might lead to corruption of files. File system storage is primarily intended for test systems.

Procedure

1. Back up the database.
2. On the command line, export the library using the export-library-content command. (For details on using the Spotfire command line, see Executing commands on the command line.)
3. Remove the content from the library.

   Do not use the **`truncate`** command in the database because there are hidden folders that should not be removed.
4. To enable external storage and to select the type of external storage, use the command config-library-external-data-storage.

5. To specify the path to the storage root, use the command config-library-external-file-storage. Subdirectories for the content files are created under this root.
6. Import the library.

The external library storage system uses the Spotfire library globally unique identifiers (GUIDs) to identify files.

For information on monitoring the external system, see Monitoring external library storage and fixing inconsistencies.

Monitoring external library storage and fixing inconsistencies

Because there is no guarantee of referential integrity when using external library storage, the administrator should regularly check for inconsistencies between the metadata in the Spotfire database and the files in external storage.

Procedure

1. On the command line, enter the command check-external-library to check for discrepancies. (For details on using the Spotfire command line, see Executing commands on the command line.) A discrepancy report is generated, including where discrepancies occur and any available information to help identify the "orphan" files. This is an excerpt from a report:

```
check-external-library
Connecting to the library...OK
Retrieving items from database...OK
Retrieving items from external storage...OK
Comparing database to external storage...OK
Found 43 orphaned items.
Retrieving meta data...
Items in external storage but not in database:
=====================================================
ID: 14b82e58-6298-4c4e-8605-f745812629e0
-----------------------------------------------------
     lastModified:   Tue Feb 25 09:50:02 CET 2015
     path:           /Sales/2nd quarter
     uploadedby:     laosh@APOTFIRE
     type:           dxp
     contentLength:  403002
=====================================================
```

2. If a file is found in external storage that is not referenced in the Spotfire database, you can download the file. If it is an analysis file, you can then manually save it to the Spotfire library. If metadata is found for a file that does not exist, you can delete the metadata.

If you want to	Do this
Retrieve an orphan file from Amazon Web Services S3 (AWS)	Download it using the command s3-download.

If you want to	Do this
Retrieve an orphan file from an external file system	Manually copy it from the file system.
Delete files from AWS	Use the command delete-library-content.
Delete files from an external file system	Manually delete the files.
Delete metadata from Spotfire Server	Use the command delete-library-content.

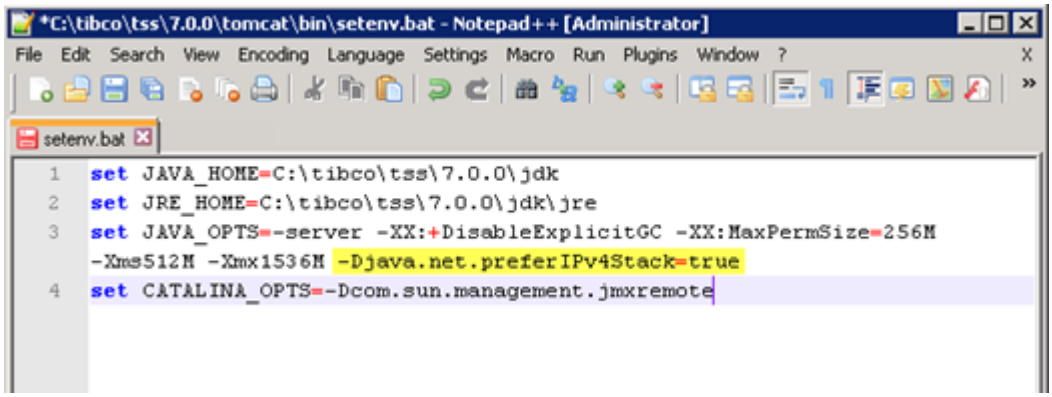
Forcing Java to use Internet Protocol version 4

If your library files are stored on Amazon Web Services S3 (AWS) and you discover instances of the following event in the server logs, you should force Java to use Internet Protocol version 4 (IPv4): `java.net.UnknownHostException: <your bucket name>.s3.amazonaws.com at java.net.Inet6AddressImpl.lookupAllHostAddr(Native Method)`

This parameter is set manually in a Spotfire Server configuration file.

Procedure

1. Open the appropriate file in a text editor:
   - If you are running Spotfire Server as a Windows service, open the `<installation dir>/tomcat/bin/service.bat` file.
   - If you are *not* running Spotfire Server as a Windows service, open the `<installation dir>/tomcat/bin/setenv.bat` file.
2. Locate the variable named `JAVA_OPTS`.
3. Enter the following parameter in the `JAVA_OPTS` section: `-Djava.net.preferIPv4Stack=true`
   The file will look similar to this (the new parameter is highlighted in yellow):



4. Save and close the file.
5. Restart Spotfire Server.

Upgrading Spotfire

Follow these steps to upgrade your Spotfire environment to the new version of Spotfire.

The Spotfire Server and node manager upgrade tools copy all relevant settings, including configurations and node manager trust, to your new Spotfire environment.

As of Spotfire Server version 10.3.0, server hotfixes can be applied only on the specific service pack version that they were created for. Example: If you currently have version 10.3.1, you can only apply server hotfixes for the 10.3.1 version, such as 10.3.1 HF-001, 10.3.1 HF-002, and so on. If you want a hotfix of a different service pack level, such as 10.3.2 HF-001, you must first make sure to upgrade to that service pack (10.3.2) before applying the hotfix. Client hotfixes have not changed.

Prerequisites

- Before upgrading, create a working backup of your Spotfire database.
- Download the required software from the TIBCO eDelivery website and the TIBCO Support website; for details, see Downloading installation software and Downloading hotfixes.

Spotfire 10.3 introduced the Data Function Trust security feature. This means that, if you are upgrading from an earlier version than 10.3, you must take additional steps during the upgrade for data functions to continue working as expected. Read about the steps you have to take in this Community article: Script and Data function trust in Spotfire 10.3 and later. Also, other important changes were made in Spotfire 10.3. Read about them in the "New Features" and "Changes in Functionality" sections of the Spotfire Server 10.3 release notes.

Procedure

1. Stop your Spotfire Servers and node managers. For information on how to stop them, see Start or stop Spotfire Server and Starting or stopping a node manager.
2. Set the **Startup type** to **Manual** for your existing Spotfire Servers and node managers to prevent the old installation from starting automatically and causing a port conflict with the new installation. For instructions, see Preventing Spotfire Servers and node managers from starting automatically.
3. Install the new version of Spotfire Server. For instructions and details related to the upgrade, see Installation of Spotfire Server during upgrade.
4. Upgrade the server by running the server upgrade tool on each server. For more information, see Run the Spotfire Server upgrade tool.

   If your servers are clustered, run the upgrade tool on all servers in the cluster.
5. Apply to all the Spotfire Servers any available server hotfix that has the same version number as the new server. For more information, see Applying hotfixes to the server during upgrade.

   Do not apply any hotfixes whose three-digit version number is different from the new server's three-digit version number. Apply only the latest hotfix for the version number.
6. Start the new Spotfire Servers. For information on how to start the Spotfire Server, see Start or stop Spotfire Server.
7. Deploy the Spotfire client packages (`Spotfire.Dxp.sdn`) and node manager packages (`Spotfire.Dxp.NodeManagerWindows.sdn`) to the new Spotfire Server. For more information on how to deploy packages to Spotfire Server, see Deploying client packages to Spotfire Server.

   After the deployment, make sure that the Administrator group has all licenses, including new ones, assigned to it. For a description of the licenses, see the License feature reference.

8. Upgrade the nodes by installing the new node manager and running the node manager upgrade tool on each node. For more information, see Upgrading nodes.

When installing the new node managers, specify the same ports that were used by the old node managers.

9. Start the node managers. For information on how to start the node managers, see Starting or stopping a node manager (as a Windows service).
10. Update all services on all nodes in your environment. For information on how to update the services, see Updating services.
11. Optional: Verify or edit changes to service configuration files. Your existing configurations will work in the new version of Spotfire, but some settings have been added or changed and must be updated manually if you do not want to use the default values. For more information, see Upgrading service configurations.

Installation of Spotfire Server during upgrade

When you install Spotfire Server, the upgrade tool is installed as well.

Before installing the new version of Spotfire Server, note the following:

- Configure the new server to use the same ports as the previous installation. This will not cause a port conflict if you have followed steps 1 and 2 in Upgrading Spotfire.
- Do not start or configure the newly installed server before running the upgrade tool.
- If you intend to copy information from the old version, do not uninstall it until the new version of Spotfire Server is in place.
- Some Spotfire releases are designated long-term support (LTS) versions; for information about how LTS versions differ from mainstream (non-LTS) versions, see https://docs.tibco.com/pub/spotfire/general/LTS/spotfire_LTS_releases.htm.

For general instructions on how to install Spotfire Server, see Installation.

Preventing Spotfire Servers and node managers from starting automatically

When upgrading Spotfire Servers and node managers to the next version, you must prevent the old version of these components from starting automatically when Windows starts. Because the old and new versions use the same communication ports, starting both versions results in a port conflict.

These instructions apply to servers that are running as a Windows service.

Procedure

1. Log in to the Spotfire Server or node manager computer as an administrator.
2. Go to **Control Panel** > **Administrative Tools** > **Services** and then, in the Services dialog, locate and select the previous version of the service called **TIBCO Spotfire Server** or **TIBCO Spotfire Node Manager**.
3. Right-click the service and then click **Properties**.
4. In the center of the Properties dialog, next to **Startup type**, select **Manual** and then click **OK**.

Result

When you restart Windows, the server or node manager will not start automatically.

Upgrading a cluster of Spotfire Servers

Clustering is implemented using Apache Ignite, and clustering is enabled by default. When you upgrade your Spotfire installation from version 7.5 or higher, the clusters will be configured to use Apache Ignite, even if you previously used ActiveSpaces or Hazelcast.

For information on upgrading, see Upgrading Spotfire. For general information on clustering, see Clustered server deployments.

If you have a load balancer that routes based on the jvmRoute part of the session id, note that the default value has changed from uppercase to lowercase. If needed, update the load balancer configuration accordingly.

These are the basic steps for upgrading a clustered implementation of Spotfire:

Procedure

1. Download the required software; see Downloading installation software and Downloading hotfixes.
2. Install the Spotfire Servers in your cluster; see Installation of Spotfire Server during upgrade.
3. Apply the latest hotfix for your new version of Spotfire Server (if one is available) to all of the servers; see Applying hotfixes to the server during upgrade.
4. Run the upgrade tool on each server in the cluster; see Run the upgrade tool.
5. On one of the servers in the cluster, set your clustering parameters; see Setting up a cluster of Spotfire Servers.
6. Start the same server; see Start or stop Spotfire Server.
7. Start the other servers in the cluster.

Applying hotfixes to the server during upgrade

Before running the upgrade tool, you must install any available hotfix for the new version of the server on all servers.

Prerequisites

As of Spotfire Server version 10.3.0, server hotfixes can be applied only on the specific service pack version that they were created for. Example: If you currently have version 10.3.1, you can only apply server hotfixes for the 10.3.1 version, such as 10.3.1 HF-001, 10.3.1 HF-002, and so on. If you want a hotfix of a different service pack level, such as 10.3.2 HF-001, you must first make sure to upgrade to that service pack (10.3.2) before applying the hotfix. Client hotfixes have not changed.

- You have installed Spotfire Server.
- You have downloaded the latest hotfix for your new version of Spotfire Server; for instructions, see Downloading hotfixes.

Procedure

- Follow the instructions in the `Installation_Instructions.htm` file that was included in the hotfix package that you downloaded.

  For more information, see Overview of hotfixes for TIBCO Spotfire in the TIBCO Community.

Run the Spotfire Server upgrade tool

The server upgrade tool updates the database. You can run the upgrade tool interactively, or silently by using the command-line interface.

If you have not already done so, make a working backup of your Spotfire database.

For information on how to run the upgrade tool, see Running the upgrade tool interactively or Running the upgrade tool silently.

Running the Spotfire Server upgrade tool interactively

When you run the Spotfire Server upgrade tool interactively, you are prompted for information about both your older installation and your new installation.

If Spotfire Server is set up to authenticate with the Spotfire database using Windows Integrated Authentication, you must run the upgrade tool as the same user that Spotfire Server authenticates as. Otherwise, the upgrade tool cannot authenticate with the database.

Prerequisites

- You have installed the new version of Spotfire Server.
- You have a working backup of your Spotfire database.
- If you are using LDAPS, and if the CA certificate is not included in the cacert file by default, you must import the CA certificate used to issue the LDAP server's certificate before running the upgrade tool. See Configuring LDAP.

Procedure

1. If the server upgrade tool is not open, go to the following directory and double-click `upgradetool.bat` (Windows) or `upgradetool.sh` (Linux): `<new version Spotfire Server install dir>/tools/upgrade`.
   By default, the server installation directory is located here: `C:/tibco/tss/version number`.
   The Spotfire Server Upgrade panel is displayed. It provides a reminder to back up or clone the Spotfire database.
2. Click **Next**.
   The File Locations panel is displayed. It provides new information and the choice to copy, or not to copy, an existing configuration.
3. Specify whether to copy an existing configuration.
   - If you have file access to copy an existing installation, select **Previous server installation**, enter the path to its installation directory, and then click **Next**.
     For example: `C:/tibco/tss/version number` or `/opt/tss/version number`.
   - If you do not have file access to an old installation, click **Next**.

   The Database Type and Driver panel is displayed.
4. Specify the database and database driver you are using, and then click **Next**.

   If you select a database driver type that is not installed in the old installation directory, the message `The selected driver must be installed manually` is displayed. Install the driver manually by placing it in the `<new version Spotfire Server install dir>/tomcat/custom-ext` directory, and then restart the upgrade tool.

   The Database Connection Information panel is displayed.
5. Provide the Spotfire database **Connection string**, **Username** and **Password**. If your database server uses integrated login, like Windows authentication, select the **Integrated login** check box, to disable the **Username** and **Password** fields.

6. Click **Next**.
   - If you copied an existing installation, then the Summary panel is displayed. It shows the new installation directory, the directory from which the configuration is copied, the database connection string, and the database username. It also provides options to **copy logs** and **move library exports**. Specify whether to copy the logs and/or move the library exports, as needed.
   - If you did not copy an existing installation, then the Additional Information panel is displayed. Specify the configuration tool password, the encryption password, and the server name to use when configuring the Spotfire Server, and then click **Next**.
7. Click **Upgrade**.
   The Upgrade panel is displayed. If there were problems with the upgrade, click **Next** to get information on where the issues have been logged.
8. When the upgrade has been successfully completed (the text `The upgrade completed successfully` is displayed in the panel), click **Finish**.

What to do next

If there are changes needed after the upgrade (for example, port configuration or the location of TLS certificate), then manually edit the `server.xml` file, located in the directory `<Spotfire Server install dir>/tomcat/conf`.

See Start Spotfire Server

Running the Spotfire Server upgrade tool silently

As an alternative to running the upgrade tool interactively, you can run it silently using the command line.

If Spotfire Server is set up to authenticate with the Spotfire database using Windows Integrated Authentication, it is important that you run the upgrade tool as the same user that Spotfire Server authenticates as. Otherwise, the upgrade tool will not be able to authenticate with the database.

Prerequisites

- You have installed the new version of Spotfire Server.
- You have a working backup of your Spotfire database.
- If you are using LDAPS, and if the CA certificate is not included in the cacert file by default, you must import the CA certificate used to issue the LDAP server's certificate *before* running the upgrade tool. See Configuring LDAP.

Procedure

1. Go to the following directory: `<new version Spotfire Server install dir>/tools/upgrade`.
2. Open the `silent.properties` file in a text editor.
3. Follow the instructions in the file and specify the values of the parameters.
   The `from` parameter is the only parameter that you are required to specify.
4. Save the `silent.properties` file.
5. Open a command line.

6. To see the parameters that the upgrade tool will use, do one of the following:
   - On Windows, type `upgradetool.bat --h`.
   - On Linux, type `upgradetool.sh --h`.

   The parameters are listed on the command line. Review the list of parameters and specify any that are applicable for your server.
7. To run the upgrade tool silently, do one of the following:
   - On Windows, type `upgradetool.bat --silent silent.properties`.
   - On Linux, type `upgradetool.sh --silent silent.properties`.
8. Press **Enter**.
   The upgrade tool runs silently.

What to do next

Start Spotfire Server

Start Spotfire Server

After applying the server hotfixes, start the Spotfire Server.

Prerequisites

Procedure

1. For information on how to start the Spotfire Server, see Starting Spotfire Server.
2. To verify that Spotfire Server has been installed and started, launch a browser and go to the Spotfire Server start page: `http://<hostname>:<port>/spotfire`.

What to do next

Upgrading nodes

Upgrading nodes

To upgrade the nodes, install the new node managers on the same computers as the old node managers. Then run the node manager upgrade tool on each new node manager.

Set the **Startup type** to **Manual** for your existing node managers to prevent the old installation from starting automatically and causing a port conflict with the new installation. For instructions, see Preventing Spotfire Servers and node managers from starting automatically.

1. Stop all node manager services.
2. Install node manager (as described under Node manager installation on page 128).
3. Run the node manager upgrade tool on all node managers.

Install node manager

The first step in upgrading the node manager is to install new node managers on the same computers as the old node managers.

You can install a node manager either interactively with a graphical interface or silently by using the command line.

- For the interactive installation on Windows, see Installing a node manager interactively during upgrade.
- For the silent installation, see Installing a node manager silently. Then see Running the node manager upgrade tool silently
- For installation on Linux, see Installing a node manager (RPM Linux) on page 132 or Installing a node manager (tarball Linux) on page 134.

  You must run the `/configure` command after installing a node manager on Linux.

Configure the node managers to use the same ports as the previous installation. This will not cause a port conflict if you have followed steps 1 and 2 in Upgrading Spotfire.

Do not start the newly installed node manager before running the upgrade tool.

Once the new node managers are installed, you can continue to Run the node manager upgrade tool on page 357.

Installing a node manager interactively during upgrade

Install the new node manager on the same computer as the old node manager. You must run the node manager installer with administrative permissions.

Prerequisites

- Spotfire Server is installed and running.

Procedure

1. In the installation kit, right-click `nm-setup.exe` and then click **Run as administrator**.
2. On the installation wizard Welcome page, click **Next**.
3. On the License page, read the agreement, select **I accept**, and then click **Next**.
4. On the Destination Folder page you can change the location if you want to, and then click **Next**.
   The Node Manager Ports page opens.
5. On the Node Manager Ports page, specify the same ports that were used by the old node manager.
6. Click **Next**.
   The Spotfire Server page opens.
7. On the Spotfire Server page, enter the following information, and then click **Next**.

   These values must match the values you used when installing the Spotfire Server files.

   - **Server name**–The hostname of Spotfire Server.

     Valid hostnames may contain only alphabetic characters, numeric characters, hyphens, and periods.
   - **Server backend registration port**–The registration port that you specified during Spotfire Server installation.
   - **Server backend communication port (TLS)**–The back-end communication port that you specified during Spotfire Server installation.
8. On the Network Names page, select the computer names that can be used by back-end trust. In general you can leave all the listed names as they are.

9. On the Ready to Install page, click **Install**.

   Do not start the newly installed node manager before running the upgrade tool.

10. On the Install Wizard Completed page, select **Launch the upgrade tool** and click **Finish**.

What to do next

Running the node manager upgrade tool interactively

Run the node manager upgrade tool

You can run the node manager upgrade tool interactively, or silently by using the command-line interface.

For information on how to run the node manager upgrade tool, see Running the node manager upgrade tool interactively or Running the node manager upgrade tool silently.

Running the node manager upgrade tool interactively

When you run the node manager upgrade tool interactively, you are prompted for the installation directory of both your old node manager installation and your new installation.

Prerequisites

You have installed the new node manager.

Procedure

1. If the node manager upgrade tool is not already open, go to the directory `<new node manager installation directory>/nm/upgrade` and double-click `upgradetool.bat`. By default, the node manager installation directory is located here: `C:/tibco/tsnm/<version number>`.
   The node manager upgrade tool opens.
2. In the **Upgrade to path** field, specify the location of your new node manager installation directory.
3. In the **Upgrade from** field, specify the location of your old node manager installation directory.
4. Indicate whether you want the upgrade tool to start the node manager Windows service after upgrade.
5. Click **Run Upgrade**.
   The result of the node manager upgrade is shown in the text field below the controls.
6. When the node manager is successfully upgraded, close the node manager upgrade tool window.

Running the node manager upgrade tool silently

As an alternative to running the node manager upgrade tool interactively, you can run it silently from the command line.

Prerequisites

You have installed the new node manager.

Procedure

1. On the command line, navigate to the directory `<new node manager installation directory>/nm/upgrade`.
   By default, the node manager installation directory is located here: `C:/tibco/tsnm/<version number>` (on Windows) or `/opt/tibco/tsnm/<version number>` (on Linux).

2. To see the parameters that the upgrade tool will use, do one of the following:
   - On Windows, type `upgradetool.bat --h`.
   - On Linux, type `upgradetool.sh --h`.

   The parameters are listed on the command line. Review the list of parameters and specify any that are applicable for your node manager.
3. Run the upgrade command using the desired parameters. For example:.

   ```
   upgradetool.bat --cmd --from old node manager installation dir --to new node manager
    installation dir
   ```

   The node manager upgrade tool runs silently.

Optional upgrades

The following upgrades may or may not apply to your Spotfire implementation.

Upgrading service configurations

Service configuration changes require manual updates if you do not want to use their default values.

To get the correct configuration files, it is recommended that you export both the default new service configuration and your old service configuration from Spotfire Server by using the export-service-config command. Then apply all changes made in the old configuration files to the new configuration files. Then import the new configuration back into Spotfire Server by using the import-service-config command, and use this configuration for your new services.

For more information on how to edit the configuration files, see Manually editing the service configuration files.

For information on the added or changed settings, see the topics for the appropriate configuration files.

Changes introduced in Spotfire 7.6

Spotfire.Dxp.Worker.Web.config

Additional service configuration settings were added for the mini-dump creation if a service goes down unintentionally.

In the `<errorReporting>` section, the following settings were added: `miniDumpSizeLarge="false"` and `miniDumpPath=""`.

The `miniDumpSizeLarge` setting can create a very large dump file that should not be edited unless instructed by Spotfire Support.

Changes introduced in Spotfire 7.9

Spotfire.Dxp.Worker.Host.exe.config

The following proxy handling settings were added, if you need to use proxy handling for communication from the Web Player service or Automation Services to Spotfire Server: `ProxyUsername`, `ProxyPassword` and `<defaultProxy>`.

Spotfire.Dxp.Worker.Automation.config

The section `<Spotfire.Dxp.Automation.Framework>` has been added, where you can specify which directories Automation Services tasks can read from, write to, and delete from.

The settings `useKerberos` and `kerberosIdentity` have been added to be able to run Automation Services jobs as a specified Windows account when delegated Kerberos is used in the environment.

Spotfire.Dxp.Worker.Web.config

The following settings have been added to configure the use of a tool, such as cdb.exe, to automatically capture dumps for hanging service instance processes: `dumpToolPath`, `dumpToolFlagsSmall`, `dumpToolFlagsLarge`.

The settings `useKerberos` and `kerberosIdentity` have been added to be able to run scheduled updates as a specified Windows account when delegated Kerberos is used in the environment.

The setting `allowGcEvenIfAnalysesLoaded` has been added. It allows you to run garbage collection even if analyses are open.

The default value of the setting `requestTimeoutSeconds` has been changed from 300 seconds to 3600 seconds.

Upgrading custom-modified log4j.properties files

For Spotfire Server 7.9, the logging framework was upgraded from Log4j to Log4j2. If you used a custom-modified `log4j.properties` file in any Spotfire Server version between 7.5 and 7.8, you must manually add these modifications to the new `log4j2.xml` file to continue using the same logging properties.

Note that custom edits to the `log4j2.xml` file are intended for settings that are not available in the administrative interface, such as adding log appenders or changing the log size or rotation.

Procedure

1. Open the `log4j.properties` file from your previous Spotfire Server installation.
2. Open the new version of the following Spotfire Server file in an XML editor or a text editor: *<new Spotfire Server installation dir>*`/tomcat/spotfire-config/log4j2.xml`.
3. Add the modifications from the old file to the new file, using the new, XML-based format. For full documentation of the new format, see https://logging.apache.org/log4j/2.x/manual/configuration.html.
4. Save and close the file.
5. Restart the server.

Applying hotfixes to the Spotfire environment

As of Spotfire Server version 10.3.0, server hotfixes can be applied only on the specific service pack version that they were created for. Example: If you currently have Spotfire version 10.3.1, you can only apply server hotfixes for the 10.3.1 version, such as 10.3.1 HF-001, 10.3.1 HF-002, and so on. If you want a hotfix of a different service pack level, such as 10.3.2 HF-001, you must first make sure to upgrade to that service pack (10.3.2) before applying the hotfix.

Client hotfixes have not changed.

For general hotfix information and links to specific information about each hotfix, see Overview of hotfixes for TIBCO Spotfire in the TIBCO Community.

Procedure

1. Sign in to the TIBCO Support website.
2. Click **Downloads** > **Hotfixes**.
3. On the Available Hotfixes page, expand **AvailableDownloads** and **Spotfire**.
4. For each product component in your implementation, locate and select the folder containing the latest applicable hotfix for your product version and click **Download**.

   Service hotfixes are in the **Clients** folder.
5. When the download is complete, unzip the folder's contents and follow the instructions in the `Installation_Instructions.htm` file.

Applying hotfixes for services

Download and install any available hotfixes for your Automation Services, Web Player services, TERR service or Spotfire Service for Python.

Procedure

1. Go to https://support.tibco.com to download the latest hotfix for your services. For instructions, see Applying hotfixes to the Spotfire environment.
2. Deploy the downloaded Spotfire distribution to the Spotfire Server. For instructions, see Adding software packages to a deployment area.
3. Update the services. For instructions, see Updating services.

Backup and restore

To enable recovery after a crash or disaster in your Spotfire environment, it is important that information stored in the system is backed up. Most of this information is stored in the Spotfire database, but some of it is stored on the Spotfire Server.

This manual will not describe how to perform backups, only what to back up. It is assumed that you have some sort of backup software for files and computers, and that you use the backup tools provided with the database. Refer to the database documentation for instructions on how to perform backups.

One can only restore to a machine running the same operating system as the backed up system, since there is a bundled Java runtime with binaries for a specific architecture.

Back up each server in the cluster.

The following sections describe what needs to be backed up.

Backup of Spotfire database

The most important part of the Spotfire environment to back up is the Spotfire database.

It contains tables which store the state of the server, for example the library, preferences, and deployments. Most of the server and service configuration files are also stored in the database. Even if only the database has been backed up, it is still possible to restore most of the functionality after a crash. It is therefore vital that you have a valid and current backup of the Spotfire database.

Verify your backups.

Backup of Spotfire Server

A small set of configuration is unique for each Spotfire Server and is stored on the actual Spotfire Server rather than in the database.

This includes information about how Spotfire Server connects to the Spotfire database, which ports the server should listen to, authentication methods such as Kerberos etc.

During installation the server files are essentially all placed in the installation directory. It should be sufficient to back up this directory, of course it is possible to back up the entire file system.

Once a server has been configured or hotfixed there are no further persistent changes. Log files and other temporary files will change, but a restored backup will have the same functionality.

The configuration which is not in the database includes:

- Listening ports configuration. See The server.xml file for more information.
- Database connection and database drivers. See Database drivers and database connection URLs for more information.
- Logging configuration. See Monitoring and diagnostics for more information.
- Memory configuration. See Virtual memory modification for more information.
- HTTPS. See HTTPS for more information.
- Authentication such as Kerberos or Client Certificates.
- Any other advanced configuration performed in Advanced procedures. When performing advanced configuration, you should always take backup into consideration.

The `bootstrap.xml` file is not stored in the database either. However, since the `bootstrap.xml` file contains a unique server ID, it can not be re-used if a server is restored on another computer. Therefore, in the event of a server crash where the server is restored on another machine, it is recommended to bootstrap the server again.

Whenever you make any configuration changes or have applied a server hotfix, you should also perform a backup of the Spotfire Server installation directory.

Windows Installations

On Windows installations, there is functionality which will not be restored by only recovering the Spotfire Server installation directory:

- Windows Service
- Uninstall functionality
- Start Menu shortcuts

The Windows Service can be (re-)installed using the bat file `service.bat` located in the `<installation dir>\tomcat\bin` directory. Run it on the command line with the following arguments: `C:\tibco\tss\<version>\tomcat\bin>service.bat install`.

Uninstallation can be done by removing the service and simply remove the installation directory.

The Start Menu shortcuts can be backed up by copying them to the server installation directory, back that up, and when restoring, copying these files to the start menu directory.

Linux Installations

On Linux installations, no essential data is placed outside the installation directory by Spotfire Server. If you have a startup script for the server, it will need to be recreated.

Network Considerations

If you are using Kerberos you should note that configuration needed for this to work is tied to a specific machine and cannot be copied easily to a new one.

You should also consider any other conditions in your environment and their implications, such as IP addresses and firewall rules, LDAP restrictions, and anything else that might affect getting a system back up and running.

Backup of services

The service configuration files are stored in the Spotfire database, so there is no need to make additional backups for the services.

If a node or service must be restored, install it again and select the configuration used for the old service.

Information on which resource pools the service instances should be used for is not stored in the database. The new service instances must be assigned to the old resource pools manually.

Uninstallation

To perform a complete uninstallation of your Spotfire environment, the following steps must be completed, in order.

Deleting services

The first step of uninstalling the Spotfire environment is to delete the installed services.

Procedure

1. Go to the Spotfire Server start page and log in as an administrator.
2. Click **Nodes & Services**.
3. On the **Your network** page, under **Select a view**, select **Nodes**.
4. In the left pane, expand the entries under the node and select the service.
5. In the right pane, click **Delete** for each installed service.

Revoking trust of nodes

The second step of uninstalling the Spotfire environment is to revoke the trust for all installed nodes.

For instructions on how to revoke the trust of a node, see Revoking trust of a node.

This must be done for each node in your Spotfire environment.

Uninstalling node manager

The third step of uninstalling the Spotfire environment is to uninstall all node managers.

Windows

Uninstallation of the node manager is performed through the regular Windows procedure. On each machine with a node manager installed, click **Start** > **Control Panel** > **Programs and Features** > **Uninstall or change a program**. Then right-click **TIBCO Spotfire Node Manager** and select **Uninstall**.

RPM Linux

To be able to uninstall on Linux, you must have root access.

On each computer with a node manager installed, uninstall by running the command:

```
rpm -e tss-nm-<version number>
```

Tarball Linux

On each computer with a node manager installed, uninstall by running the following commands:

If the node manager was configured to start on boot, it must be stopped and removed.

To stop the node manager, run the command:

```
service tss-nm-<version number> stop
```

To remove the node manager, run the command:

`chkconfig --del tss-nm-<version number>`

Delete added scripts by running the following commands:

`rm /etc/init.d/tss-nm-<version number>`

`rm /etc/sysconfig/tss-nm-<version number>`

The final step is to remove the folder with node manager files. Do this by running the following command:

`rm -rf <folder where the tarball was installed>`

Uninstalling Spotfire Server

The fourth step of uninstalling the Spotfire environment is to uninstall the Spotfire Server.

If you have placed any additional files in the installation directory or any of its subdirectories, such as Spotfire library export files, you should move these files to a secure location before uninstalling. The installer will remove the installation directory and all its subdirectories.

Windows

Uninstallation of Spotfire Server is performed through the regular Windows procedure. On each computer with a Spotfire Server installed, click **Start** > **Control Panel** > **Programs and Features** > **Uninstall or change a program**. Then right-click **TIBCO Spotfire Server** and select **Uninstall**.

After successful uninstallation, only use-modified files (such as custom JDBC drivers) remain on the computer.

RPM Linux

To be able to uninstall on Linux, you must have root access.

On each computer with a Spotfire Server installed, uninstall the server by running the command:

`rpm -e tss-<version number>`

After a successful uninstallation, only modified files in `tomcat/conf` remain.

Tarball Linux

To be able to uninstall on Linux, you must have root access.

On each computer with a Spotfire Server installed, uninstall the server by running the following commands:

If the Spotfire Server was configured to start on boot, it must be stopped and removed.

To stop the server, run the command:

`service tss-<version number> stop`

To remove the server, run the command:

`chkconfig --del tss-<version number>`

Delete added scripts by running the following commands:

`rm /etc/init.d/tss-<version number>`

`rm /etc/sysconfig/tss-<version number>`

The final step is to remove the folder with Spotfire Server files. Do this by running the following command:

```
rm -rf <folder where the tarball was installed>
```

Advanced procedures

These manual procedures are for setting up various features that are supported by Spotfire. Many of the procedures assume prior knowledge of technologies such as LDAP, Kerberos, Apache httpd, and so on.

Custom configurations for managing space needs

If you need more space for library content, log files, information links, or the files that the Web Player service writes to the hard disk, you can change the default settings to store these items in different directories.

For information about the system requirements for Spotfire Server, see TIBCO Spotfire Server System Requirements.

Links to information for changing settings

Configuration need	For more information
Change the directory where Spotfire Server log files are written.	Follow the instructions in Changing the default location of server logs.
Configure the directory for library imports and exports.	Use the configuration command config-import-export-directory.
Change the maximum size of the cache for the Web Player service Information Links and library content.  If Spotfire Server is configured to cache Information Links and library content, this uses additional disk space. By default, caching is enabled and the max cache size set to 10 GB.	Set the `--max-cache-size` option for the configuration command config-attachment-manager.
Change the amount of disk space available for all of the log files generated by the Web Player service.	Follow the instructions in Customizing the service logging configuration.
Change the location of the temporary directory that the Web Player service uses for temporary files, paging, and caching data (scheduled updates caching and SBDF caching).	Follow the instruction in Changing the default location of the Web Player temporary files.

Changing the default location of the Web Player temporary files

By default the Web Player service stores temporary files, paging, and caching data (scheduled updates caching and Spotfire Binary Data File (SBDF) caching) in the Temp directory inside the service installation directory. If you need to change the location, or if Spotfire Support suggests that you change it, follow this procedure.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the path of the `config.bat` file (`config.sh` on Linux).

   The default file path is `<server installation dir>/tomcat/spotfire-bin`.

2. Export the service configuration by using the export-service-config command.
   Example:

```
config export-service-config --tool-password=mypassword --capability=WEB_PLAYER --
deployment-area=Production
```

3. Open the `Spotfire.Dxp.Worker.Host.exe.config` file in a text editor or XML editor and locate the following section. By default, the exported configuration files are saved to the *<server installation dir>*`/tomcat/spotfire-bin/config/root` directory.

```
<Spotfire.Dxp.Internal.Properties.Settings>
      <setting name="TempFolder" serializeAs="String">
        <value>Temp</value>
      </setting>
```

4. Replace the value `Temp` with the path to the new Temp directory.

   The Temp directory should be located on a local disk.

   Example:

```
<Spotfire.Dxp.Internal.Properties.Settings>
      <setting name="TempFolder" serializeAs="String">
        <value>C:\NewTemp</value>
      </setting>
```

5. Save and close the configuration file.
6. Return to the command line and import the custom configuration using the import-service-config command.
   Example:

```
config import-service-config --tool-password=mypassword --config-name=SampleConfig
```

7. Apply the custom configuration to specific services by using the set-service-config command.
   Example:

```
config set-service-config --tool-password=mypassword --service-id="VALUE" --config-
name=SampleConfig
```

   Use the list-services command to get the service ID.

Result

The configuration setting for the indicated Web Player service is displayed in Nodes & Services, and the Web Player temporary files should be written as specified.

Temporary tablespace

By default, the tablespaces/database files for Spotfire Server with either an Oracle or SQL database uses autoextend/autogrowth. If this does not meet your needs, alter the settings.

You may want to alter the amount that the files are extended with each increment.

For Oracle, review the maxsize for each table space. For SQL, review the unlimited growth property.

Virtual memory modification

If many simultaneous users intend to perform heavy data pivoting via Information Services or in other ways stress the server, you may need to modify the amount of memory available to the virtual computer.

Modifying the virtual memory (server not running as Windows service)

If Spotfire Server is not running as a Windows service, you can modify the virtual memory by following these steps to set up the start script.

Procedure

1. Open the file `<installation dir>/tomcat/bin/setenv.bat` or `<installation dir>/tomcat/bin/setenv.sh` in a text editor.
2. Locate the lines that sets the variables listed below and set the values to the amount of memory you want to allocate (in MB):
   - JvmMs
   - JvmMx
3. Restart the server.

Modifying the virtual memory (server running as Windows service)

If Spotfire Server is running as a Windows service, you can modify the virtual memory by following these steps to set up the start script.

Procedure

1. Stop the Spotfire Server service.
2. On the command line, go to the `<installation dir>/tomcat/bin` directory.
3. Enter the following command: `service.bat remove`
4. Open the `<installation dir>/tomcat/bin/setenv.bat` file in a text editor.
5. Locate the following entries and change the numbers to suitable memory values (in MB):
   - `JvmMs=512`
   - `JvmMx=4096`
6. Save and close the file.
7. Enter the following command: `service.bat install`
8. Start the Spotfire Server service.

Garbage collection logging

When old requests to a service become obsolete, the objects created in memory becomes garbage. By enabling garbage collection logs (GC logs) you can get an understanding of your system's performance and troubleshoot memory issues.

Enabling GC logging (server running as Windows service)

If Spotfire Server is running as a Windows service, you can enable GC logging by following these steps.

Procedure

1. Stop the Spotfire Server service.
2. On the command line, go to the `<installation dir>/tomcat/bin` directory.
3. Enter the following command: `service.bat remove`

4. Uncomment the following line in `service.bat`:

```
set "GC_OPTS=-Xlog:gc*,gc+ergo*=trace:file=%CATALINA_HOME%\logs\gc-%
%t.log:time,level,tags:filecount=5,filesize=25m"
```

5. Enter the following command: `service.bat install`
6. Start the Spotfire Server service.

Enabling GC logging (server running on Windows)

If Spotfire Server is running on Windows (but not running as a Windows service), you can enable GC logging by following these steps.

Procedure

1. Stop the server.
2. Open the file `<installation dir>/tomcat/bin/setenv.bat` in a text editor.
3. Uncomment the following line:

```
set GC_LOG=-Xlog:gc*,gc+ergo*=trace:file=%CATALINA_HOME%\logs\gc-%
%t.log:time,level,tags:filecount=5,filesize=25m
```

4. Start the server.

Enabling GC logging (server running on Linux)

If Spotfire Server is running on Linux, you can enable GC logging by following these steps.

Procedure

1. Stop the server.
2. Open the file`<installation dir>/tomcat/bin/setenv.sh` in a text editor.
3. Uncomment the following line:

```
GC_LOG="-Xlog:gc*,gc+ergo*=trace:file=$CATALINA_HOME/logs/gc-
%t.log:time,level,tags:filecount=5,filesize=25m"
```

4. Start the server.

Spotfire Server public web services APIs

Spotfire Server offers several web services application programming interfaces (APIs) for building custom applications that interact with Spotfire Server. There are both SOAP and REST APIs.

All of the current APIs use OAuth 2.0 for authentication and authorization.

API documentation is available at https://docs.tibco.com/products/tibco-spotfire-server.

Spotfire Server SOAP APIs

These web services APIs use the SOAP protocol for managing administrative tasks programmatically.

Spotfire Server Web Services API

This API contains the following services:

- InformationModelService is for updating the configuration of an Information Model data source.
- LibraryService is for managing the Spotfire library.
- LicenseService is for managing licenses and their functions.
- UserDirectoryService is for managing users and groups.

For information, see the Web Services API Reference at https://docs.tibco.com/products/tibco-spotfire-server.

Additional information

- A description of each web service (a WSDL file) can be retrieved by appending `/wsdl` to each web service URL.
- The WSDL files can be used to generate client proxies that contain all types and methods that can be used.
- The implementing classes may not be called directly from Java code.

Setup

These tasks must be completed before you can use these APIs:

- Register an OAuth 2.0 API client
- Generate client proxies

Optional: Configure Spotfire Server APIs

Spotfire Server REST APIs

These web services APIs use REST technology to manage administrative tasks programmatically.

API for executing Automation Services

This API makes it possible to run Automation Services jobs.

Library Upload API

This API makes it possible to upload SBDF files to the Spotfire library.

The Library Upload API contains the following rate limit capabilities. These configuration properties can be set by using the **`set-config-prop`** command.

- To set the maximum number of ongoing upload jobs per client, use the `public-api.library.upload.limit.max-concurrent-jobs-per-client` parameter. Default: 10.
- To set the maximum number of ongoing upload jobs allowed per server cluster, use the `public-api.library.upload.limit.max-concurrent-jobs` parameter. Default: 1000.
- To set the maximum size (in bytes) for an item to be uploaded, use the `public-api.library.upload.limit.max-upload-size` parameter. Default: 2 GB.

For more information, see set-config-prop and Executing commands on the command line.

The REST API Reference is available at https://docs.tibco.com/products/tibco-spotfire-server.

These APIs do not use HTTP sessions so there is no need to maintain session cookies.

Additional resources

The following resources are available if online API documentation is enabled. (This is enabled by default, but can be toggled on and off by using the **`config-web-service`** command.)

- An interactive documentation page is available on a running server at this address:

  https://*server.example.com*/spotfire/api/swagger-ui.html
- An Open API (Swagger) definition of the REST APIs can be obtained on a running server at this address:

  https://*server.example.com*/spotfire/api/v2/api-docs?group=<group-name>

  where <group-name> is API-specific (see the documentation for the respective API). This can be used for creating client stubs.

Setup

Before using these APIs, you must complete this task: Register an OAuth 2.0 API client.

Optionally, you may want to configure the APIs; see Configure Spotfire Server APIs.

Registering an OAuth 2.0 API client

The Spotfire Server public Web Services APIs use the OAuth 2.0 protocol for authentication and authorization. Therefore, before the API can be used, an OAuth 2.0 API client must be registered.

Procedure

- Use the **`register-api-client`** command.

  For more information, see register-api-client and Executing commands on the command line.

Generating client proxies

To use the Spotfire Server SOAP Web Services APIs, you must generate a client proxy for each web service that you want to use.

Proxies can be generated using the tool of your choice.

Configuring Spotfire Server Web Services APIs

The current Spotfire Server public APIs are enabled by default. You can disable them or make other configuration changes.

Procedure

- Use the **`config-web-service-api`** command.

  For more information, see config-web-service-api and Executing commands on the command line.

Optional security HTTP headers

The Spotfire Server can be configured to include extra security-oriented HTTP headers in its responses.

The headers in this section are optional. Only the header X-Content-Type-Options is included by default.

Enable these headers only if you know how they work and you understand the effects they can have on your deployment.

X-Frame-Options

The X-Frame-Options HTTP header provides basic protection against some clickjacking attacks (also known as UI redress attacks).

The feature can be switched on by running the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line. (For details on using the Spotfire command line, see Executing commands on the command line.)

```
config export-config --force
config set-config-prop -n security.x-frame-options.enabled -v true
config import-config -c "Enabled X-Frame-Options"
```

The feature can be switched off by running the following commands:

```
config export-config --force
config set-config-prop -n security.x-frame-options.enabled -v false
config import-config -c "Disabled X-Frame-Options"
```

When this feature is enabled, the server includes the HTTP header "X-Frame-Options: SAMEORIGIN" in all responses.

The directive can also be customized by running the following commands:

```
config export-config --force
config set-config-prop -n security.x-frame-options.directive -v <value>
config import-config -c "Customized X-Frame-Options directive"
```

`<value>` can be set to any of the following values:

- DENY: Prevents the rendering of the server web page within a frame.
- SAMEORIGIN: Prevents the rendering of the server web page within a frame if origin mismatch.
- ALLOW-FROM: The server web page will be rendered only when framed from the specified location.
- ALLOWALL: Allows rendering within a frame from any location. (This is a non-standard value which is not supported by all browsers.)

X-XSS-Protection

The X-XSS-Protection HTTP header provides basic protection against some XSS attacks by indicating to the browser clients how they should use their built-in XSS protection filter.

This functionality is enabled by default for new Spotfire Server installations, and for installations upgraded from 7.5 or later, but not for installations upgraded from versions that are earlier than 7.5.

The feature can be switched on by running the following commands in the `<server installation dir>/tomcat/spotfire-bin` directory on the command line. (For details on using the Spotfire command line, see Executing commands on the command line.)

```
config export-config --force
config set-config-prop -n security.x-xss-protection.enabled -v true
config import-config -c "Enabled X-XSS-Protection"
```

The feature can be switched off by running the following commands:

```
config export-config --force
config set-config-prop -n security.x-xss-protection.enabled -v false
config import-config -c "Disabled X-XSS-Protection"
```

When this feature is enabled, the server will include the HTTP header "X-XSS-Protection: 1; mode=block" in all responses.

The directive can also be customized by running the following commands:

```
config export-config --force
config set-config-prop -n security.x-xss-protection.directive -v value
config import-config -c "Customized X-XSS-Protection directive"
```

<value> can be set to any of the following values:

- "0"
- "1"
- "1; mode=block"

Make sure to put quotation marks around the last argument on the command line.

HTTP Strict-Transport-Security (HSTS)

The Strict-Transport-Security HTTP header provides support for the HTTP Strict Transport Security (HSTS) standard, as specified by RFC 6797.

It helps to protect against protocol downgrade attacks and cookie hijacking by declaring that user agents, such as web browsers or Spotfire Analyst clients, must interact with the Spotfire Server using secure HTTPS connections.

The feature can be switched on by running the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line. (For details on using the Spotfire command line, see Executing commands on the command line.)

```
config export-config --force
config set-config-prop -n security.hsts.enabled -v true
config import-config -c "Enabled HSTS"
```

The feature can be switched off by running the following commands:

```
config export-config --force
config set-config-prop -n security.hsts.enabled -v false
config import-config -c "Disabled HSTS"
```

When this feature is enabled, the server will include the HTTP header "Strict-Transport-Security: max-age=0" in all responses.

Use the following commands to customize the max-age directive:

```
config export-config --force
config set-config-prop -n security.hsts.max-age-seconds -v <value>
config import-config -c "Customized HSTS max-age directive"
```

<value> can be any positive integer value, representing the number of seconds the HSTS policy should remain in effect.

The `includeSubDomains` directive is by default not included in the HTTP header, but it can be enabled by running the following commands:

```
config export-config --force
config set-config-prop -n security.hsts.include-sub-domains -v true
config import-config -c "Enabled includeSubDomains directive for HSTS"
```

The `includeSubDomains` directive can be excluded from the HTTP header by running the following commands:

```
config export-config --force
config set-config-prop -n security.hsts.include-sub-domains -v false
config import-config -c "Disabled includeSubDomains directive for HSTS"
```

Cache-Control

The Cache-Control header controls how the browser caches web resources. To make sure that no sensitive files are ever stored on the file system, enable the Cache-Control header to prevent the files from being cached by the browser.

The feature can be switched on by running the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line. (For details on using the Spotfire command line, see Executing commands on the command line.)

```
config export-config --force
config set-config-prop -n security.cache-control.enabled -v true
config import-config -c "Enabled Cache-Control"
```

The feature can be switched off by running the following commands:

```
config export-config --force
config set-config-prop -n security.cache-control.enabled -v false
config import-config -c "Disabled Cache-Control"
```

When this feature is enabled, the server will include the HTTP header "Cache-Control: no-cache, no-store, must-revalidate" in all responses.

Use the following commands to customize the header directive:

```
config export-config --force
config set-config-prop -n security.cache-control.directive -v <value>
config import-config -c "Customized Cache-Control directive"
```

Replace `<value>` with any valid cache-control header directive.

You cannot customize the Cache-Control header for files ending with ".html" or attachments with content type "text/html" or "text/plain". These files will always have the value "no-cache, no-store, must-revalidate". They will also get the "Pragma" header set to "no-cache" and the "Expires" header set to "0". The Pragma headers are legacy HTTP 1.0 headers and serve the same purpose as the "Cache-Control" header in HTTP 1.1.

X-Content-Type-Options

The X-Content-Type-Options HTTP header can be used to prevent user agents, such as web browsers or Spotfire Analyst clients, from guessing the MIME content type. Instead, they will always use the declared content type.

The X-Content-Type-Options header is enabled by default.

The feature can be switched off by running the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line:

```
config export-config --force
config set-config-prop -n security.x-content-type-options.enabled -v false
config import-config -c "Disabled X-Content-Type-Options"
```

If switched off, the feature can be switched on again by running the following commands:

```
config export-config --force
config set-config-prop -n security.x-content-type-options.enabled -v true
config import-config -c "Enabled X-Content-Type-Options"
```

For details on using the Spotfire command line, See Executing commands on the command line.

SameSite Cookie Attribute

You might need to change this value in scenarios where the Spotfire Server cookies are used as third party cookies. For example, it might be needed when external web sites and Spotfire are interacting.

Use the server command-line configuration tool to specify the property. For details on using the Spotfire command line, see Executing commands on the command line.

Example:

```
config export-config --force
config set-config-prop --name="security.cookies.same-site" --value="None"
config import-config -c "Cookies SameSite=None"
```

Valid values for the property are:

- None
- Lax
- Unset

The default is `Unset`, which is a special Tomcat value, and which preserves previous behavior.

The values `None` and `Lax` are defined by rfc6265bis.

Changing how long the server waits before assuming that a node manager is offline

You can configure the amount of time that Spotfire Server waits for a node manager to signal its presence. If the node manager does not send a signal within the configured time period, the server assumes that the node is offline. For setups that are experiencing a heavy load, you can raise this value to avoid unnecessarily restarting a node manager.

The default value for this property is 12,000 milliseconds (12 seconds).

Procedure

1. Open a command line and export the active server configuration by using the export-config command; for additional information, see Executing commands on the command line.
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=nodemanager.heartbeat.threshold --value=X
   ```

   where X is the length of time, in milliseconds, that the server will wait for the node manager signal.
3. Import the configuration back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server service.

Disable administration tasks on specific Spotfire Servers

You can block access to administration functionality on one or more servers in a cluster. This means that no logged-in user on that server, regardless of their role, can view or modify administrative information and settings. You can use this feature to prevent Spotfire administrators from accessing the administration UI when logging in from external networks.

For each server, you have the following blocking options:

- Block all areas of the administration UI.
- Block all areas of the administration UI except for Monitoring & Diagnostics.
- Block only Monitoring & Diagnostics.

By default, administrators can perform administration tasks on any server in the cluster.

Based on your Spotfire environment and the configuration you want to achieve, select a procedure to follow.

- To begin with administration functionality enabled on all the servers so that you can then specify the servers on which to block administration functionality, see Disabling administration tasks on specific Spotfire Servers (by selecting servers to disable).
- To begin by blocking administration functionality on all the servers so that you can then specify the servers on which to allow administration functionality, see Disabling administration tasks on specific Spotfire Servers (by selecting servers to enable).

Disabling administration tasks on specific Spotfire Servers (by selecting servers to disable)

You can block access to administration functionality on one or more servers in a cluster. You can use this feature to prevent Spotfire administrators from accessing the administration UI when logging in from external networks.

For more information, see Disable administration tasks on specific Spotfire Servers.

Procedure

1. Open a command line and export the active server configuration by using the export-config command; for additional information, see Executing commands on the command line.
2. To begin with administration functionality enabled on all the servers, enter the following commands on the command line. This sets the property for all Spotfire Servers in the cluster.

   ```
   config set-config-prop --name=security.administration.enabled --value=true
   ```

   ```
   config set-config-prop --name=security.administration.diagnostics-enabled --value=true
   ```

   For information about the command options, see set-config-prop.
3. To set a specific server on which administration tasks will be blocked, enter one or both of the following commands:

   - To block access to all areas of the administration UI except for Monitoring & Diagnostics, enter this command:

     ```
     config set-config-prop --name=security.administration.enabled --value=false --server-
     name=<server alias from the bootstrap.xml file>
     ```

     where "server alias from the bootstrap.xml file" refers to the server on which you want to block access.
   - To block access to the Monitoring & Diagnostics area of the administration UI, enter this command:

     ```
     config set-config-prop --name=security.administration.diagnostics-enabled --
     value=false --server-name=<server alias from the bootstrap.xml file>
     ```

   For the **`server-name`** parameter, make sure to use the name as it appears in the bootstrap file.
4. Repeat step 3 for each Spotfire Server on which administration tasks will be blocked.
5. Import the configuration file back to the Spotfire database by using the import-config command.
6. Restart all the Spotfire Servers in the cluster.

7. To prevent users on external networks from performing administration tasks, make sure that all users who log in to Spotfire from external networks are routed to the servers on which administration functionality is disabled.

Disabling administration tasks on specific Spotfire Servers (by selecting servers to enable)

You can block access to administration functionality on all the servers in a cluster, and then select the servers on which to enable administration functionality. You can use this feature to prevent Spotfire administrators from accessing the administration UI when logging in from external networks.

For more information, see Disable administration tasks on specific Spotfire Servers.

Procedure

1. Open a command line and export the active server configuration by using the export-config command; for additional information, see Executing commands on the command line.
2. To begin by blocking administration functionality on all the servers, enter the following commands on the command line. This sets the property for all Spotfire Servers in the cluster.

   ```
   config set-config-prop --name=security.administration.enabled --value=false
   ```

   ```
   config set-config-prop --name=security.administration.diagnostics-enabled --value=false
   ```

   For information about the command options, see set-config-prop.
3. To set a specific server on which administration tasks will be enabled, enter one or both of the following commands:

   - To enable access to all areas of the administration UI except for Monitoring & Diagnostics, enter this command:

     ```
     config set-config-prop --name=security.administration.enabled --value=true --server-
     name=<server alias from the bootstrap.xml file>
     ```

     where "server alias from the bootstrap.xml file" refers to the server on which you want to enable access.
   - To enable access to the Monitoring & Diagnostics area of the administration UI, enter this command:

     ```
     config set-config-prop --name=security.administration.diagnostics-enabled --value=true
      --server-name=<server alias from the bootstrap.xml file>
     ```

   For the **`server-name`** parameter, make sure to use the name as it appears in the bootstrap file.
4. Repeat step 3 for each Spotfire Server on which administration tasks will be enabled.
5. Import the configuration file back to the Spotfire database by using the import-config command.
6. Restart all the Spotfire Servers in the cluster.
7. To prevent users on external networks from performing administration tasks, make sure that all users who log in to Spotfire from external networks are routed to the servers on which administration functionality is disabled.

Changing the settings that determine when Web Player instances are recycled due to low temporary disk space

By default, Spotfire recycles Web Player instances if their temporary disk space falls below 1500 MB (the "exhausted" level), and remains below that level for one hour. When a service instance is recycled, the corresponding process is restarted; all open analyses on that instance are closed and its temp files are removed. You can change the definition of "exhausted" and the period of time that trigger Web Player recycling.

Only experienced administrators should edit these values.

Procedure

1. On the computer running Spotfire Server, open a command line as an administrator and change the directory to the location of the `config.bat` file (`config.sh` on Linux). The default location is `<server installation dir>/tomcat/spotfire-bin`.
2. On the command line, export the service configuration that you want to modify from Spotfire Server by using the export-service-config command. Specify the service's capability and the deployment area, and optionally the configuration name.

   If you are editing a service configuration that has been applied to an existing service, you must verify the name of the active service configuration before you export it. If the name of the active configuration is not "Default", you must specify the name in the **`export`** command.

   Example for exporting the "Default" Web Player configuration that is in the Production deployment area:

   ```
   config export-service-config --capability=WEB_PLAYER --deployment-area=Production
   ```

   Example for exporting a customized configuration:

   ```
   config export-service-config --config-name=WebPlayerConfiguration
   ```

   For more information, see Manually editing the service configuration files.
3. Edit the period of time that the Web Player service must remain exhausted to trigger recycling:
   a. Open the following file in an XML editor or a text editor: `<server installation dir>\tomcat\spotfire-bin\config\root\Spotfire.Dxp.Worker.Web.config`.
   b. In the file, locate the `recycleWhenOutOfDiskEnabled` setting:

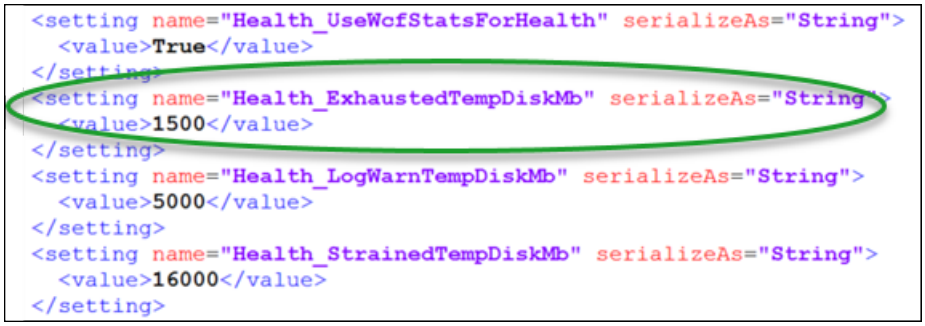
   ```
   <recoverMemory enabled="true"
                  actionWhenOk="0" actionWhenStrained="1" actionWhenExhausted="2"
                  recycleIfScheduledAndCacheEnabled="false"
                  recycleEvenIfScheduledAnalyses="false"
                  triggerEvenIfUsersLoggedIn="true"
                  allowGcEvenIfAnalysesLoaded="false"
                  minMinutesBetweenGc="60" minMinutesBeforeRecycle="300"
                  recycleWhenOutOfDiskEnabled="true"
                  recycleWhenOutOfDiskAfter="01:00:00" />
   <documentCache purgeInterval="300" itemExpirationTimeout="00:00:00"/>
   ```

   c. Make sure that the `recycleWhenOutOfDiskEnabled` setting equals "true".
   d. Edit the time period by changing the value of the `recycleWhenOutOfDiskAfter` setting in the format "hh:mm:ss".
   e. Save and close the file.

4. Edit the amount of free disk space below which recycling is triggered (after the time period specified in step 3d):

   a. Open the following file in an XML editor or a text editor: `<server installation dir>\tomcat\spotfire-bin\config\root\Spotfire.Dxp.Worker.Host.exe.config`.

   b. In the file, locate the "`Health_ExhaustedTempDiskMb`" setting:



   c. Edit the value, in MB, that represents the "exhausted" state by changing the "`Health_ExhaustedTempDiskMb`" setting.

   d. Save and close the file.

5. On the command line, import the customized configuration back into Spotfire Server and name the configuration by using the import-service-config command.

   If the configuration to be imported was created from the default configuration, a name *must* be specified.

   If you are editing already customized configuration files, specifying a name when importing will create a new service configuration. If you import the changed customized configuration without the `--config-name` parameter, the old customized configuration will be replaced.

   Example

   ```
   config import-service-config --config-name=ServiceConfiguration
   ```

   When you install a new service or edit an existing one, you can select the new customized configuration.

6. Optional: To activate the customized configuration for an existing service, enter the following command on the command line:

   ```
   config set-service-config --service-id=value --config-name=ServiceConfiguration
   ```

   Use the list-services command to obtain the service ID.

   Activating the configuration for a Spotfire Web Player service causes its instances to restart.

Setting the maximum execution time for an Automation Services job

This Spotfire Server property indicates how long an Automation Services job can run before the server cancels the job. The default setting for this property is 259,200 seconds (72 hours).

Procedure

1. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command.

2. Enter the following command:

```
config set-config-prop --name="automation-services.max-job-execution-time" --value="X"
```

   where "X" is the length of time, in seconds, that an Automation Services job is permitted to run.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart Spotfire Server.

Setting the maximum inactivity time for an Automation Services job

This Spotfire Server property indicates how long an Automation Services job can remain inactive before the server cancels the job. The default setting for this property is 259,200 seconds (72 hours).

Procedure

1. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command.
2. Enter the following command:

```
config set-config-prop --name="automation-services.job-inactivity-timeout" --value="X"
```

   where "X" is the time period, in seconds, after which the server will cancel an inactive Automation Services job.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart Spotfire Server.

Absolute session timeout and idle session timeout

Absolute session timeout is a recommended security feature, while idle session timeout is mainly a resource management feature.

Absolute session timeout requires all Spotfire users to log in to the program again after the configured amount of time. This is true whether a user has been working in Spotfire the entire time, has left the computer unattended, or has shut the computer down. The data associated with the session remains available to the user so that they can log back in (on the same computer or a different computer) and continue working. The absolute session timeout default is 1,440 minutes (24 hours).

However, because open user sessions tie up system resources that could be used elsewhere, the *idle session timeout* begins its countdown when a user shuts down their computer or the computer is no longer connected to the Spotfire network. If the user does not reactivate their session before the idle session timeout has been reached, the data associated with the session is destroyed and the session's resources become available for other sessions. The idle session timeout default is 30 minutes.

The session is not considered "idle" until the computer shuts down or disconnects from the network because Spotfire Web Player, like many other applications, makes periodic background requests to the server.

Because the login page makes no background requests, when an absolute session timeout occurs, the session data is eventually destroyed when the idle session timeout is reached. This assumes that the user is not immediately logged back in again because they previously selected the **Keep me logged in** check box.

Both idle session timeout and absolute session timeout are set in the `configuration.xml` file. Therefore, in a clustered implementation the setting applies to all the resources in the cluster.

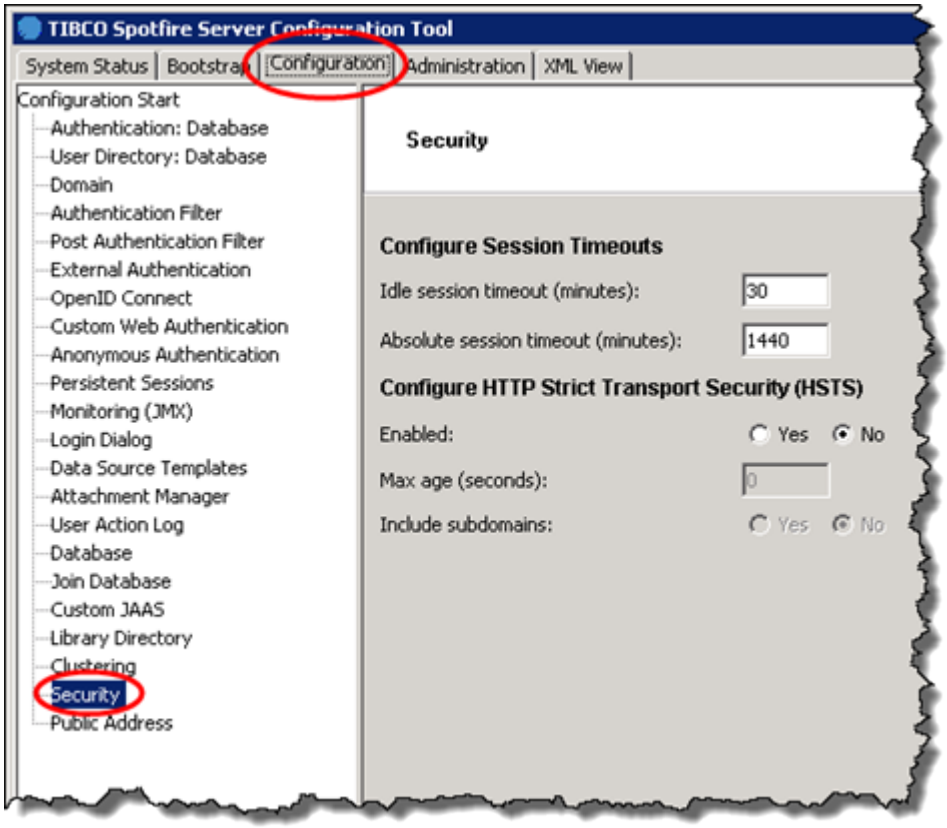
These timeout properties can be configured either in the Spotfire configuration tool or on the command line.

Setting idle session timeout and absolute session timeout by using the configuration tool

Both session timeout values can be adjusted in the **Security** section of the Spotfire configuration tool.

Procedure

1. If the configuration tool is not open, open it; for instructions see Opening the configuration tool.
2. On the Configuration page, at the bottom of the left pane, click **Security**.



3. Under **Configure Session Timeouts** you can change the number of minutes for the idle session timeout and absolute session timeout.
4. Click **Save configuration**.
5. Restart the Spotfire Server.

Setting idle session timeout by using the command line

The primary function of the idle session timeout is to release the resources that are associated with a user session when the computer is inactive for the configured amount of time. The default is 30 minutes.

Procedure

1. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)

2. On the command line, enter the following command:

```
config set-config-prop -n security.idle-session-timeout -v XX
```

where XX is the number of minutes after which an idle user session will be closed.

A negative value for XX indicates that the idle session timeout value that is configured for the container (in the `web.xml` file) will be used. A value of 0 indicates that a user session will never be closed based solely on its inactivity.

3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server.

Setting absolute session timeout by using the command line

The absolute session timeout indicates the number of minutes after which a user must log in to Spotfire again. The default is 1,440 minutes (24 hours).

Procedure

1. Open a command-line interface and export the active configuration (the `configuration.xml` file) by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

```
config set-config-prop -n security.absolute-session-timeout -v XX
```

where XX is the number of minutes after which a user must log in again.

3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server.

Changing whether scheduled updates are sent to exhausted service instances

By default, if all the Web Player instances in an implementation or a site are listed as "exhausted", scheduled update requests for analyses that are not cached will not be sent to a Web Player instance until one becomes available (no longer exhausted). In the same situation, a scheduled update request for an analysis that is already cached *will* be sent to exhausted instances. You can change these defaults by editing the Spotfire Server configuration file.

Procedure

1. On the server computer, export and open the `configuration.xml` file. For detailed instructions on working with this file, see Manually editing the Spotfire Server configuration file.
2. In the `configuration.xml` file, locate the following section:

```
<scheduled-updates>
  ...
  <performance>
      <load-on-exhausted-instances>false</load-on-exhausted-instances>
      <update-exhausted-instances>true</update-exhausted-instances>
  </performance>
  ...
</scheduled-updates>
```

3. To allow scheduled update analyses that are not cached to use exhausted Web Player instances, change the `load-on-exhausted-instances` value to "true".

4. To prevent scheduled update analyses that are cached from using exhausted Web Player instances, change the `update-exhausted-instances` value to "false".

   When a Web Player instance becomes available, the scheduled update is applied only if the rule is still scheduled at that time.

5. Save and close the file.
6. Import the file back to the Spotfire database.
7. Restart the server.

Preventing users from opening scheduled update files outside of their schedule window

Large analysis files are often managed by scheduled updates so that end users can view these files without waiting for them to download. If an end user tries to open one of these scheduled update files outside of its schedule window, however, the file can take a long time to open and may significantly tie up system resources. You can configure the server to block end-user access to these files when the files are not scheduled.

This configuration applies to all scheduled update files. It has no effect on files that are not managed by scheduled updates.

Procedure

1. Open a command-line interface and export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=scheduled-updates.performance.deny-open-when-not-scheduled
    --value=true
   ```

   For information about the command options, see set-config-prop.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server.

Changing whether recovered rules are automatically enabled

When an analysis file is deleted from the library, any scheduled update or routing rule for that file fails. If the analysis file is then imported back to its previous location, the rule is recovered but it does not run because the rule is, by default, in the "disabled" state. You can switch the default for these recovered rules to "enabled".

Procedure

1. Open a command-line interface and export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=scheduled-updates.enable-recovered-rules-automatically --
   value=true
   ```

   For information about the command options, see set-config-prop.
3. Import the configuration file back to the Spotfire database by using the import-config command.

4. Restart the Spotfire Server.

Restarting a node manager to terminate its running jobs

Use this procedure to "refresh" a node when its service instances appear to be running jobs that should have terminated.

Procedure

1. Log on with administrator credentials to the computer on which the node manager was installed.
2. Open the Windows Services list and stop the "TIBCO Spotfire Node Manager" service.
3. Open Windows Task Manager and end all the "Spotfire.Dxp.Worker.Host.exe" processes.
4. Restart the "TIBCO Spotfire Node Manager" service.

Increasing the number of available sockets on Linux

Spotfire Server opens many connections, and each requires a file descriptor. For performance and security reasons, Linux has limited the number of connections that can be opened by a process. You may want to increase this limit.

Procedure

1. Open a terminal as root and go to the directory that contains the following file: `/etc/security/limits.conf`.
2. Edit the file, adding the following text lines in the given order:

   ```
   spotuser soft nofile 8192
   ```

   ```
   spotuser hard nofile 65000
   ```

   where

   - *spotuser* is the account that is running the Spotfire Server.
   - *nofile* refers to the number of files that can be opened by a process.

   The soft limit may be changed later, up to the hard limit value. The hard limit can only be lowered (except by a root user).

   In this example, 8,192 files (including sockets) can be opened. The setting should be high enough for the system, but not too high.

   To test the limit without editing the file, you can add a line similar to the following:

   ```
   ulimit -n 32000
   ```

   where *32000* represents the file limit that you want to test.

   The hard limit can be raised, by a root user, up to the limit indicated in this setting: `/proc/sys/fs/file-max`.

Switching from online to offline administration help

By default, the help button on the administration pages of Spotfire Server opens the online version of this documentation. If you are unable to use the online version, you can switch to the offline version.

Any updates to this documentation will be available at https://docs.tibco.com/products/tibco-spotfire-server.

Procedure

1. On the computer running Spotfire Server, open a command-line interface and go to the following directory: `<server installation dir>/tomcat/spotfire-bin`.
2. On the command line, enter the following commands:

```
config export-config --force
```

```
config set-config-prop -n general.applications.admin.use-online-help -v false
```

```
config import-config -c "Switching to offline administration help"
```

3. Restart the Spotfire Server.

Displaying or hiding the Spotfire Server version

You can configure which users should be able to see information on the Spotfire Server version.

Default mode

By default, information about the Spotfire Server version is present in the About view and in the URL of the online help resources. This information is available to all logged in users. If anonymous authentication is enabled, the information is also available to anonymous users. Users who have not logged in cannot access the version information.

To activate the **default** mode, run the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line:

```
config export-config --force
config set-config-prop -n security.version-settings-mode -v default
config import-config -c "Setting the version settings mode to default"
```

Safe mode

To hide this version information from anonymous users, so that the version information is only available to logged in users, it is possible to activate a **safe** mode.

To active the **safe** mode, run the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line:

```
config export-config --force
config set-config-prop -n security.version-settings-mode -v safe
config import-config -c "Setting the version settings mode to safe"
```

Unsafe mode

To make the version information available to everyone, including anonymous users as well as users who have not logged in, it is possible to active an **unsafe** mode.

To active the **unsafe** mode, run the following commands in the `<server installation directory>\tomcat\spotfire-bin` directory on the command line:

```
config export-config --force
config set-config-prop -n security.version-settings-mode -v unsafe
config import-config -c "Setting the version settings mode to unsafe"
```

This configuration setting does not affect the web client. See the `showAbout` and `showHelp` settings in the Spotfire.Dxp.Worker.Web.config configuration file for information on how to disable these features in the web client.

Hiding the Spotfire header in the user interface

When cobranding Spotfire, you can add your own header. By default, the added custom header and the Spotfire header are both visible in the UIs. If you want, you can hide the Spotfire header.

If you hide the Spotfire header, actions that are available in the Spotfire header will be handled in the custom header.

For information about adding custom headers, see the TIBCO Spotfire® Cobranding manual.

Procedure

1. Open a command-line interface and export the active configuration by using the export-config command. (For details on using the Spotfire command line, see Executing commands on the command line.)
2. On the command line, enter the following command:

   ```
   config set-config-prop --name=general.applications.header.hide-default --value=true
   ```

   For information about the command options, see set-config-prop.
3. Import the configuration file back to the Spotfire database by using the import-config command.
4. Restart the Spotfire Server.

Contacting support

If you encounter an issue that requires assistance from TIBCO Support, consider including the following information (where applicable to your specific issue) when reporting the issue, to help ensure a quick resolution.

- Describe the issue in detail, including any error messages.
- List all products/components and exact versions involved in the issue.
- When was the issue first observed? Has it ever worked in the past? How often does it occur?
- Were any changes made in the environment (on the Spotfire side or externally, such as changes to the operating system/web browser/database/anti-virus software, and so on) around the time that the issue started?
- Are the steps needed to reproduce/trigger the issue known? If so, describe them and, if possible, provide any objects (such as analysis files) that are needed to reproduce it.
- Is the extent of the issue known? For example, does it only affect one/some objects (such as specific servers/analysis files/users), while others work? If so, list any objects that are affected, and also state if there are any known differences between those that work and those that do not.
- Provide logs from the time of the issue. (It is always strongly recommended to submit all available logs). A convenient way to gather the server-side logs is by generating a troubleshooting bundle. For more information, see Troubleshooting bundle.

If you have a way to reproduce the issue, it is recommended to set the logging level to DEBUG (for more information, see Changing server and node logging levels), reproduce the issue, and then provide the captured logs. Remember to reset the logging level after you are done.

After you have gathered the information, submit your issue to TIBCO Support using the TIBCO Customer Support Portal at https://support.tibco.com.

Reference

Spotfire Server files

These files contain configuration information for the server.

For information about using the `configuration.xml` file, see Manual configuration.

For information about the service configuration files, see Service configuration files.

Bootstrap.xml file

The bootstrap configuration file contains the basic information that Spotfire Server requires to bootstrap itself so that it can connect to the Spotfire database and retrieve its configuration.

The bootstrap configuration file is created by running the bootstrap command (or using the configuration tool) . The file must be created in the `<installation dir>\tomcat\webapps\spotfire\WEB-INF` directory (Windows) or the `<installation dir>/tomcat/webapps/spotfire/WEB-INF` directory (Linux). When specifying an alternative bootstrap configuration file path to the bootstrap command, the generated file must be manually copied to this directory before it can be accessed by the server. The file must also be named `bootstrap.xml`.

This is the format of the bootstrap configuration file:

```
<bootstrap>
  <server-name>...</server-name>
  <server>
    <driver-class>...</driver-class>
    <database-url>...</database-url>
    <username>...</username>
    <password>...</password>
  </server>
  <config-tool>
    <driver-class>...</driver-class>
    <database-url>...</database-url>
    <username>...</username>
    <password>...</password>
  </config-tool>
  <server-name>...</server-name>
  <encryption-password>...</encryption-password>
</bootstrap>
```

- The `<config-tool>` section

  This section is optional and not required for running the server itself. It is only required for using the configuration commands to access the database. If the commands are not to be used on a specific server, they can easily be disabled by removing this section.

  The database password stored in this section is protected by a special configuration tool password that is specified when creating the `bootstrap.xml` file. This tool password must be specified whenever running a command that accesses the database.

  The tool password is not related to any administrator user account within the server application itself.

- The `<server-name>` section

  This section contains the server name, which is used for identifying the server, for example when specifying server-specific configuration.

- The `<encryption-password>` section

  This section is optional. If specified, it contains a password to be used for encrypting other passwords that are stored in the database. If not set, a static password is used.

  The same password must be configured for all servers in a cluster.

Server.xml file

Spotfire Server is implemented as a Tomcat web application. For this reason, it uses a standard Tomcat web application configuration file, `server.xml`, to store information it needs when starting. This file is stored in the *<installation dir>*`/tomcat/conf/` directory.

In general, there are two reasons that an administrator might edit this file:

- To change port numbers after installation.
- To tweak Tomcat behavior.

Note that each Spotfire Server in a cluster has a `server.xml` file.

The variable [SpotfirePort] is set when running the Spotfire Server installer. The variable [ServerHostname]-srv is automatically set by the installer by adding the strings -srv to the server's hostname. This variable must not contain any characters that need escaping, such as "."

For details about the `server.xml` syntax, see Apache Tomcat documentation at https://tomcat.apache.org/.

For information on editing the file, see Manually editing the server.xml file.

Server hostname example

spotfireserver1.example.com

By default Spotfire Server has three pre-configured connectors. Connectors with connectorType="registration" and connectorType="backend" should not be touched. The public connector (it has no connectorType specified explicitly) can be modified or commented out for load balancing and other purposes.

Manually editing the server.xml file

The `server.xml` file, a Tomcat web application configuration file, is rarely edited in Spotfire.

For more information about the file, see Server.xml file.

Procedure

1. Open the following file in an XML editor or a text editor: `<server installation dir>/tomcat/conf/server.xml`.
2. Make the necessary changes.
3. Save and close the file.
4. Restart the Spotfire Server.

Krb5.conf file

The `krb5.conf` file contains settings for Kerberos. The unmodified version of the file is presented first, followed by a version with example values.

This is the unmodified file:

```
[libdefaults]
    default_realm = MYDOMAIN
    default_keytab_name = spotfire.keytab
    default_tkt_enctypes = aes128-cts
    default_tgs_enctypes = aes128-cts
    forwardable = true

[realms]
    MYDOMAIN = {
        kdc = mydc.mydomain
        admin_server = mydc.mydomain
        default_domain = mydomain
    }

[domain_realm]
    .mydomain = MYDOMAIN
    mydomain = MYDOMAIN

[appdefaults]
    autologin = true
    forward = true
    forwardable = true
    encrypt = true
```

This is the file with example values:

```
[libdefaults]
    default_realm = RESEARCH.EXAMPLE.COM
    default_keytab_name = spotfire.keytab
    default_tkt_enctypes = aes128-cts
    default_tgs_enctypes = aes128-cts
    forwardable = true

[realms]
    RESEARCH.EXAMPLE.COM = {
        kdc = example-dc.research.example.com
        admin_server = example-dc.research.example.com
        default_domain = research.example.com
    }

[domain_realm]
    .research.example.com = RESEARCH.EXAMPLE.COM
    research.example.com = RESEARCH.EXAMPLE.COM

[appdefaults]
    autologin = true
    forward = true
    forwardable = true
    encrypt = true
```

Ports and firewall configuration

These are the main ports used by Spotfire. The following table indicates their function, the default port number, firewall requirements and, for internal ports, how to change the port when Spotfire has already been installed and configured.

Ports through which Spotfire receives communication (inbound ports) must be opened in any active firewall.

Ports through which Spotfire sends communication (outbound ports) are open by default unless they match a firewall rule that blocks them.

Internal ports

The following ports are used for communication between Spotfire components.

Name of port	Function	Default	Firewall requirements	How to change port
Public HTTP port  The HTTP connector port and the HTTPS connector port are configured independently. You can use either of them or, in some cases, both.	Used for non-secure communication with installed and web clients.	80	On computers running Spotfire Server, these ports must be open. Computers running Spotfire Analyst and web browser clients must have access to these ports. Proxies, and load balancers in front of servers, also require access to these ports.	In the `server.xml` file, edit the relevant `Connector port` parameter. For general instructions, see Manually editing the server.xml file.
HTTPS connector port	Used for secure communication with installed and web clients.	443		
Server back-end registration port	Used for setting up trust between the Spotfire Server and nodes.	9080	On computers running Spotfire Server, these ports must be open. Computers running node managers must have access to these ports.	
Server back-end communication port	Spotfire Server listens to secure traffic from services on the nodes. Used for secure traffic between nodes.	9443		
Node manager registration port	Used for setting up trust between node managers and Spotfire Server.	9080	Computers running Spotfire Server must have access to these ports, and computers running node manager must open these ports and have access. For example, if you run a service such as the TERR service on one node and the Web Player on another node, then the Web Player must have access to the TERR service through its communication port.	Edit the following file: `<node manager installation dir>\nm\config\nodemanager.properties`
Node manager communication port	Used for secure communication within the environment.	9443		
Service communication port	Used by Spotfire Web Player instances and Automation Services instances for secure communication and basic functionality.	9501		In Spotfire Server, in the Nodes & Services area, on the "Your network" page, select a service instance on the left, and then click **Edit** in the upper-right pane.
TERR service communication port	Used by the TERR service for secure communication and basic functionality.	9502		

Name of port	Function	Default	Firewall requirements	How to change port
Spotfire Service for Python communication port	Used by the Spotfire Service for Python for secure communication and basic functionality.	9503		
TERR service engine ports	Used by TERR engines running under the TERR service.	61000-62000	No firewall configuration needed.	For information about changing the TERR service engine ports, see the TERR service configuration information.
Spotfire Service for Python engine ports	Used by Python engines running under the Spotfire Service for Python.	62000-63000	No firewall configuration needed.	For information about changing the Python engine ports, see TIBCO® Spotfire Service for Python configuration information.
Clustering port	Used for secure communication within the environment. This port is the same for all servers in the cluster.	5701	These ports must be open between all the Spotfire Servers in the cluster.	Use the Spotfire configuration tool to change the port for the clustered servers. On the Configuration page, click **Clustering** in the left pane.
Second clustering port	A second clustering port, used by Apache Ignite.	5702  This port number is equal to the first clustering port number plus one.		
Third clustering port	A third clustering port, used by Apache Ignite.	5703  This port number is equal to the first clustering port number plus two.		
JMX RMI port	If JMX RMI access is enabled, Spotfire Server opens a separate port for this purpose.	1099	Computers running monitoring clients must have access to this port.	Use the config-jmx command.

Outbound ports on the server

Spotfire Server uses the following ports to communicate with programs outside of Spotfire. To facilitate this communication, firewalls must allow outgoing traffic through these ports.

Type of port	Function	Default	Firewall requirements
Database communication port	The Spotfire database server listens to this port.	Oracle database: 1521 SQL Server: 1433 PostgreSQL: 5432	Computers running Spotfire Server must have access to this port.
LDAP port	An optional number indicating the TCP port that the LDAP service is listening on.	When using LDAP over TLS, the port number defaults to 389. When using the LDAPS protocol, the port number defaults to 636.	
Global Catalog LDAP port	Active Directory LDAP servers also provide a Global Catalog containing forest-wide information, instead of domain-wide information only.	LDAP: 3268 LDAPS: 3269	
TIBCO Enterprise Message Service (EMS)	This service can be used to trigger scheduled updates. EMS listens to this port.	Non-secure connection: 7222 Secure connection: 7243	
Data connectors For information on available connectors, see "List of Connectors in this Version" in the Spotfire Analyst User's Guide.	Data connectors listen to these ports.	Varies	
Kerberos/GSSAPI	Used by the Kerberos authentication method, as well as when authenticating to LDAP server using the GSSAPI method.	Fixed port 88 on the Active Directory domain controllers	
Microsoft Net Logon, SMB, and CIFS	Used by the NTLM v2 authentication method.	Fixed port 445 on the Active Directory domain controllers	
Open ID Connect providers	Used by the web authentication method.	443	
SMTP port	Used by Automation Services.	25, 2525, or 587 Secure SMTP: 465, 25, or 587	
Databases and other services used by Information Services	JDBC-compliant data sources and other services used by Information Services listen to these ports.	Oracle database: 1521 SQL Server: 1433 Netezza: 5480 Otherwise, it varies.	
JMX RMI port	If JMX RMI access is enabled, Spotfire Server opens a separate port for this purpose.	1099	Computers running monitoring clients must have access to this port.

Server bootstrapping and database connection pool configuration

The Spotfire database holds all user data and most of the configuration for the Spotfire system. To connect to the Spotfire database, Spotfire Server uses a database connection pool.

The `bootstrap.xml` file contains the information that the server needs to connect to the Spotfire database and retrieve the configuration; refer to The bootstrap.xml file. After the server has retrieved the configuration from the database, it re-initializes its database connection pool using information from both the `bootstrap.xml` file, which is present on each server, and any database configuration set for the entire cluster, which is stored as part of the database persisted server configuration.

For the common database configuration tasks, use the commands modify-db-config and set-db-config.

Database connectivity

The Spotfire Server database connection pool implementation is used for two things: connecting to the Spotfire database and connecting to JDBC compliant data sources through Information Services.

Each connection pool (either for Spotfire Server itself or for fetching data) has many parameters; the following are of general interest:

- The `driver-class` parameter contains the JDBC driver class name; see Database drivers and database connection URLs.
- The `url` parameter contains the JDBC connection URL; see Database drivers and database connection URLs.
- The `username` parameter contains the name of the database user to connect as, if applicable.
- The `password` parameter contains the password for the specified database user, if applicable. The password is always encrypted and must therefore be set using the bootstrap command. It cannot be set manually.
- The `min-connections` parameter contains the minimum number of allocated connections.
- The `max-connections` parameter contains the maximum number of allocated connections. Depending on the pooling scheme, the total number of connections created by the server may be higher than the value of this parameter during high load, but all such extra connections will automatically be closed when the load decreases. By setting this parameter to zero or a negative value, connection pooling is effectively disabled and new connections will be continuously created as needed.
- The `pooling-scheme` parameter defines the connection pooling algorithm to be used. There are two possible connection pooling algorithms that determine the way the connection pool operates, "DYNAMIC" and "WAIT". The "WAIT" algorithm is the default.

  When initialized, the connection pool creates a number of idle database connections equal to the `min-connections` parameter. When the connection pool receives a request for a database connection, it checks if the pool contains any idle connections and uses one of those, if available.

  - The "DYNAMIC" pooling scheme—If there are no idle connections in the pool, it automatically creates a new database connection. There is no upper limit for how many connections a connection pool can have open at the same time.
  - The "WAIT" pooling scheme—If there are no idle connections in the pool and the number of already open connections is less than the `max-connections` parameter, it creates a new database connection.

    If the number of already open connections is equal to the `max-connections` parameter, it waits for an active connection to be returned to the pool. If the request cannot be fulfilled within a number of seconds equal to the `login-timeout` parameter, the request times out. In the server

logs entries similar to this appear, "Timeout while waiting for database connection after 10 seconds".

Thus, in WAIT mode, the connection pool can never have more open (active or idle) connections than the value of the `max-connections` parameter. Whenever a database connection is returned, it is put in the pool of idle connections, unless it is used immediately to fulfill an already waiting request.

Idle connections in the database connection pool eventually time out if they are not used. The `connection-timeout` parameter defines how long (in seconds) a connection can remain idle in the connection pool before being closed and discarded.

Database drivers and database connection URLs

The following details and examples show how the database connection URL is constructed.

Supported databases and JDBC drivers

Database	Driver name
Oracle (DataDirect Driver)	tibcosoftwareinc.jdbc.oracle.OracleDriver
Oracle (Oracle JDBC Thin Driver, `ojdbc*.jar`)	oracle.jdbc.OracleDriver
Microsoft SQL Server (DataDirect Driver)	tibcosoftwareinc.jdbc.sqlserver.SQLServerDriver
Microsoft SQL Server (Microsoft JDBC Driver, `sqljdbc*.jar`)	com.microsoft.sqlserver.jdbc.SQLServerDriver
PostgreSQL driver `postgresql*.jar`	org.postgresql.Driver

Database connection URL components

Component	Description
API	Specifies which API to use. This is always jdbc.
Database Driver	Specifies which database driver to use to connect to the database. Default tibcosoftwareinc, which will use the Spotfire DataDirect driver. If you have installed a different driver, you may provide this here.
Server Type	Specifies the type of database server. Either sqlserver or oracle.  Server Type is only applicable when using the DataDirect driver.
Hostname	Specifies the hostname of the database server.
Port	Specifies the port which the database server listens to; for example 1433.
Database name, SID, or service name	Specifies the name (MSSQL, PostgreSQL), SID (Oracle) or Service Name (Oracle) that defines your Spotfire database.
Options	Specifies further options, separated with semicolons. Only necessary if you want to set something specific for your database server, such as a named Instance in an MSSQL server. See the following examples.

Database connection URL examples

Database driver	URL structure	Examples
Oracle (DataDirect Driver)	[API]:[DBDriver]:[ServerType]://[Hostname]:[Port];SID=[SID]	jdbc:tibcosoftwareinc:oracle://dbsrv.example.com:1521;SID=spotfire
Oracle (DataDirect Driver)	[API]:[DBDriver]:[ServerType]://[Hostname]:[Port];ServiceName=[Service Name]	jdbc:tibcosoftwareinc:oracle://dbsrv.example.com:1521;ServiceName= pdborcl.example.com
Oracle (Vendor Driver, ojdbc*.jar)	[API]:[DBDriver]:[DriverType]://[Hostname]:[Port]:SID	jdbc:oracle:thin:@dbsrv.example.com:1521:orcl
Oracle (Vendor Driver, ojdbc*.jar)	[API]:[DBDriver]:[DriverType]://[Hostname]:[Port]/[ServiceName]	jdbc:oracle:thin:@//dbsrv.example.com:1521/pdborcl.example.com
Microsoft SQL Server (DataDirect Driver)	[API]:[DBDriver]:[ServerType]://[Hostname]:[Port];DatabaseName=[DBName]	jdbc:tibcosoftwareinc:sqlserver://dbsrv.example.com:1433;DatabaseName= spotfire_server Example using NTLM Authentication: jdbc:tibcosoftwareinc:sqlserver://dbsrv.example.com:1433;DatabaseName=spotfire_server;AuthenticationMethod=ntlmjava
Microsoft SQL Server (Vendor Driver, sqljdbc*.jar)	[API]:[DBDriver]://[Hostname]:[Port];DatabaseName=[DBName]	jdbc:sqlserver://dbsrv.example.com:1433;DatabaseName=spotfire_server;selectMethod=cursor Example: Making sure that the driver always returns prevents infinite waits during adverse conditions jdbc:sqlserver://dbsrv.example.com:1433;DatabaseName=spotfire_server;lockTimeout=<X, where X is a good value>  Due to a restriction in the vendor Microsoft SQL Server driver, you may need to add the option `responseBuffering=adaptive` to your connection string. This is necessary if you are going to store large analysis files in the library. Example: Using `responseBuffering=adaptive` jdbc:sqlserver://dbsrv.example.com:1433;databaseName=spotfire_server;selectMethod=cursor;responseBuffering=adaptive Example: Using Integrated Authentication jdbc:sqlserver://dbsrv.example.com:1433;DatabaseName=spotfire_server;selectMethod=cursor;integratedSecurity=true;
PostgreSQL	[API]:[DBDriver]://[Hostname]:[Port]/[DBName]	jdbc:postgresql://dbsrv.example.com:5432/spotfire_server

Command-line reference

The command-line commands are listed alphabetically here.

Refer to Configuration and administration commands by function for an easily reviewed functional command grouping, and Configuration using the command line for information on using the Spotfire command line.

In this reference we use the following symbols:

- Angle brackets (< >) indicate mandatory arguments.
- Square brackets ([ ]) indicate optional arguments.
- Curly brackets ({ }) indicate flags that can be specified multiple times.

Arguments can normally be specified in two different formats. For example, the `max cache size` argument may be entered as `--max-cache-size=<value>` or `-m <value>`.

A negative value must be preceded by a backslash in the second argument format, for example `-m \-7`.

add-ds-template

Adds a new data source template.

```
add-ds-template
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
[-e <true|false> | --enabled=<true|false>]
<template definition file>
```

Overview

Use this command to add a new data source template used by Information Services. The name of the template must be unique.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name=value`	Required	none	The name of the data source template to add.
`-e <true|false> --enabled=<true|false>`	Optional	false	Indicates whether the newly created data source template should be enabled.
`<template definition file>`	Required	none	The path to the file containing the data source template definition.

add-member

Adds a user or group as a member of a specified group.

```
add-member
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-g value | --groupname=value>
[-u value | --member-username=value]
[-m value | --member-groupname=value]
```

Overview

Use this command to add an existing user or group as a member of another existing group.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`-g value --groupname=value`	Required	none	The name of the group to which the member should be added. Unless the group is part of the internal SPOTFIRE domain, the name of the group must include the group's domain name, for example "RESEARCH\group" or "group@research.example.com".
`-u value --member-username=value`	Required, unless the `--member-groupname` argument is specified.	none	The name of the user to add as a member of the specified group. Unless the user is part of the configured default domain, the name of the user must include the user's domain name, For example "RESEARCH\user" or "user@research.example.com". The `--member-username` and `--member-groupname` arguments are mutually exclusive.
`-m value --member-groupname=value`	Require, unless the `--member-username` argument is specified.	none	The name of the group to add as a member of the specified group. Unless the group is part of the internal SPOTFIRE domain, the name of the group must include the group's domain name, for example "RESEARCH\group" or "group@research.example.com". The `--member-username` and `--member-groupname` arguments are mutually exclusive.

bootstrap

This command is used to bootstrap the server by creating a new bootstrap configuration file, and a corresponding server node in the database.

To update an existing file, use the update-bootstrap command.

```
bootstrap
[-f | --force]
[-n | --no-prompt]
[-o | --force-encryption-password]
[-c value | --driver-class=value]
[-d value | --database-url=value]
[-u value | --username=value]
[-p value | --password=value]
[-k value | --kerberos-login-context=value]
{-Ckey=value}
[-E <true|false> | --enable-config-tool=<true|false>]
[-t value | --tool-password=value]
[-e value | --encryption-password=value]
[-a value | --server-alias=value]
[-S value | --site-name=value]
{-Avalue}
[bootstrap configuration file]
[-i <true|false> | --use-only-ips=<true|false>]
```

Overview

Use this command to create a new bootstrap configuration file.

Options

Option	Optional or Required	Default Value	Description
`-f` `--force`	Optional	none	Indicates that the tool should overwrite any existing bootstrap configuration file.
`-n` `--no-prompt`	Optional	none	Specifies that the tool should not prompt for missing password arguments.
`-o` `--force-encryption-password`	Optional		When this flag is specified, the operation will be performed even if the encryption password specified does not match the one currently in use.  This option should only be used to recover from a situation where the encryption password currently in use is lost and where there is no remaining `bootstrap.xml` file containing it. If a `bootstrap.xml` file with the current encryption password does exist, use that file together with the config-encryption command to change the encryption password before running this command.
`-c value` `--driver-class=value`	Optional	tibcosoftwareinc.jdbc.oracle.OracleDriver	The name of the JDBC driver class.
`-d value` `--database-url=value`	Optional	jdbc:tibcosoftwareinc:oracle://localhost:1521;SID=orcl	The JDBC URL to the database. Because this argument usually contains special characters, make sure to escape those characters or enclose the values between quotes.
`-u value` `--username=value`	Optional	none	The database account user name.
`-p value` `--password=value`	Optional	none	The database account password.
`-k value` `--kerberos-login-context=value`	Optional	none	If you use the Kerberos protocol to log in to the database, use this argument to specify the name of the JAAS application configuration to be used for acquiring the Kerberos TGT. This JAAS application configuration

Option	Optional or Required	Default Value	Description
			must be registered with Java using a `login.config.url` parameter in the `<server installation dir>\jdk\conf\security\java.security` (Windows) or `<server installation dir>/jdk/conf/security/java.security` (Linux) file.  The Spotfire Server `import-jaas-config` command cannot be used for this purpose because the JAAS application configurations that are imported using this command are stored in the database, which prevents Spotfire Server from using them for creating the initial connection to the database.
`-Ckey=value`	Optional	none	A JDBC connection property. Can be specified multiple times with different keys.
`-E <true|false> --enable-config-tool=<true|false>`	Optional	true	If "true", the `<config-tool>` section should be created. Without this section, the configuration tool cannot be used on this computer. See Bootstrap.xml file.
`-t value --tool-password=value`	Optional	true	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. Can be specified only if a password is given and the argument **`--enable-config-tool`** is set to "true".
`-e value --encryption-password=value`	Optional	none	The password for encrypting passwords that are stored in the database. If you do not set this option, a static password is used. Note that the same password must be configured for all servers in a cluster.
`-a value --server-alias=value`	Optional	The fully qualified host name as determined when this command is run, but it is only ever used as a unique identifier.	The server alias. Used for identifying the server, for example when specifying server-specific configuration.
`-S value --site-name=value`	Required unless there is only one site available (in which case the	Default	The name of the site to which the server should belong. The list-sites command can be used to find names of all available sites.

Option	Optional or Required	Default Value	Description
	server will be placed in that site).		New sites can be created using the create-site command.
`-Avalue`	Optional	The host name(s) and IP address(es) as determined when this command is run.	The possible node backend addresses (host names and IP addresses). Used for internal communication within the Spotfire collective. The addresses will be used in the order they are provided (in cases where there is a need for ordering). This argument may be specified multiple times with different values.
`[bootstrap configuration file]`	Optional	none	The path to the bootstrap configuration file to create. See Bootstrap.xml file.
`-i <true|false> --use-only-ips=<true|false>`	Optional	false	When this flag is specified, auto detection of hostnames and IP addresses will be limited to include only IP addresses. This argument and the `-Avalue` argument are mutually exclusive, and IP addresses are detected automatically only if the `-Avalue` argument is not used.

Examples

- To bootstrap the server to use an Oracle database with the bundled DataDirect JDBC driver:

```
config bootstrap --driver-class=tibcosoftwareinc.jdbc.oracle.OracleDriver --database-
url="jdbc:tibcosoftwareinc:oracle://server:1521;SID=spotfire" --username=spotuser --
password=spotuser
```

- To bootstrap the server to use an Oracle database with the Oracle thin JDBC driver:

```
config bootstrap --driver-class=oracle.jdbc.OracleDriver --database-
url="jdbc:oracle:thin:@server:1521:spotfire" --username=spotuser --password=spotuser
```

- To bootstrap the server to use a Microsoft SQL Server database with the bundled DataDirect JDBC driver:

```
config bootstrap --driver-class=tibcosoftwareinc.jdbc.sqlserver.SQLServerDriver --
database-url="jdbc:tibcosoftwareinc:sqlserver://server:1433;DatabaseName=spotfire_server"
 --username=spotuser --password=spotuser
```

- To bootstrap the server to use a Microsoft SQL Server database with the Microsoft JDBC driver:

```
config bootstrap --driver-class=com.microsoft.sqlserver.jdbc.SQLServerDriver --database-
url="jdbc:sqlserver://server:1433;DatabaseName=spotfire_server" --username=spotuser --
password=spotuser
```

- To specify multiple back-end addresses for the server:

```
config bootstrap -Ahostname.example.com -Ahostname -Aip.x.y.z
```

check-external-library

Checks for inconsistencies between external storage and the Spotfire database.

```
check-external-library
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[--file-storage-path=value]
```

Overview

Use this command to check the consistency between what is stored in external storage (for example Amazon S3 or a file system), and what is stored in the Spotfire database.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`file-storage-path=value`	Optional	none	This parameter can be used to set the base path to an external file system storage.

clear-join-db

Clears the default join database configuration.

```
clear-join-db
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to clear the default join database configuration, which means that the Spotfire database is used as the default join database (the default behavior).

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

clear-preference

Clears a preference for a group.

```
clear-preference
<-g value | --group=value>
<-c value | --category=value>
<-p value | --type=value>
<-n value | --name=value>
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to clear a preference for a group.

Options

Option	Optional or Required	Default Value	Description
`<-g value | --group=value>`	Required	none	The group to clear the preference for.
`<-c value | --category=value>`	Required	none	The preference category to clear.
`<-p value | --type=value>`	Required	none	The preference type to clear.
`<-n value | --name=value>`	Required	none	The preference name to clear.
`[-b value | --bootstrap-config=value]`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`[-t value | --tool-password=value]`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.

config-action-log-database-logger

Configures the user action database logger.

```
config-action-log-database-logger
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--driver-class=value]
[-d value | --database-url=value]
[-u value | --username=value]
[-p value | --password=value]
[--commit-period=value]
[--wait-on-full-queue-time=value]
[--wait-on-empty-queue-time=value]
[--grace-period=value]
[--pruning-period=value]
[--queue-size=value]
[--batch-size=value]
[--thread-pool-size=value]
[--workers=value]
[--block-on-full-queue=<true|false>]
[--prioritized-categories=value]
[--monitoring-retention-span=value]
[--monitoring-average-period=value]
[--log-local-time=<true|false>]
```

Overview

Use this command to configure the user action database logger.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--driver-class=value`	Optional	none	The name of the JDBC driver class.
`-d value --database-url=value`	Optional	none	The JDBC URL to the database. Because this argument usually contains special characters, be sure to escape those characters or enclose the values between quotes.
`-u value --username=value`	Optional	none	The database account username.
`-p value --password=value`	Optional	none	The database account password.
`--commit-period=value`	Optional	none	The frequency (in seconds) that log events should be committed from the queue to the database when the queue is not full.
`--wait-on-full-queue-time=value`	Optional	none	The time (in milliseconds) to wait before retrying to place a new log event on the queue after being rejected by a full queue.
`--wait-on-empty-queue-time=value`	Optional	none	Sets the time (in milliseconds) to wait before trying to create a batch from the queue after an empty queue has been encountered.
`--grace-period=value`	Optional	none	The grace period for the database logger (in seconds). This is the period that the database logger is given at server shutdown to move all items from the queue to the database.
`--pruning-period=value`	Optional	48 hours	The maximum time (in hours) that logged items are kept in the database. Pruning takes place at server startup, and then at one hour intervals, when all items older than the here-specified

Option	Optional or Required	Default Value	Description
			number of hours are deleted. To disable pruning, set this argument to 0.
`--queue-size=value`	Optional	none	The maximum number of log events in the queue.
`--batch-size=value`	Optional	none	The number of log events that should be moved from the queue to the database in each batch insert.
`--thread-pool-size=value`	Optional	none	The number of threads available for the batch insert workers.
`--workers=value`	Optional	none	The maximum number of batch insert workers at any given time.
`--block-on-full-queue=<true|false>`	Optional	none	Specifies whether placing a log event on the queue should be allowed to be blocked indefinitely if the queue is full.
`--prioritized-categories=value`	Optional	none	A comma-separated list of log categories that should have higher priority in the queue.
`--monitoring-retention-span=value`	Optional	none	The length of time monitoring entries should be saved before they get crunched into averages.
`--monitoring-average-period=value`	Optional	none	The period between two averaged measurements.
`--log-local-time=<true|false>`	Optional	If "false", or not set, timestamps will be in UTC time.	Sets whether timestamps should be in local time or not.

config-action-logger

Configures the user action logger.

```
config-action-logger
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--categories=value]
[--file-logging-enabled=<true|false>]
[--database-logging-enabled=<true|false>]
[--monitoring-period=value]
```

Overview

Use this command to configure the user action logger.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--categories=value`	Optional	none	A comma-separated list of the categories that should be logged by the user action logger. To enable logging for all categories, specify "all".
`--file-logging-enabled=<true|false>`	Optional	none	Specifies whether the user action logger should log to file.
`--database-logging-enabled=<true|false>`	Optional	none	Specifies whether the user action logger should log to database.
`--monitoring-period=value`	Optional	none	Specifies how often monitoring properties are reported.

config-action-log-web-service

Configures the action log web service.

```
config-action-log-web-service
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--categories=value]
[--allowedHosts=value]
```

Overview

Use this command to configure the action log web service.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--categories=value`	Optional	none	A comma-separated list of categories that should be allowed to log through the web service. To enable all categories, specify "all".
`--allowedHosts=value`	Optional	none	A regular expression that sets the hosts allowed to use the logger web service. To enable all hosts, specify `.*`

config-anonymous-auth

Configures the anonymous authentication method.

```
config-anonymous-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
```

Overview

Use this command to configure anonymous authentication. Anonymous authentication is always combined with another main authentication method, as configured by the config-auth command. Note that you also must enable the ANONYMOUS\guest account, using the enable-user command, for anonymous authentication to work.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	false	Specifies whether anonymous authentication should be enabled.

config-attachment-manager

Configures the attachment manager.

```
config-attachment-manager
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e value | --max-cache-expiration-time=value]
[-m value | --max-cache-size=value]
[-E <true|false> | --encryption-enabled=<true|false>]
[-k value | --encryption-key-size=value]
```

Overview

Use this command to configure the attachment manager, which handles data transfer (for instance Library downloads and uploads) to and from Spotfire Server.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e value --max-cache-expiration-time=value`	Optional	86400	The maximum idle time (in seconds) after which cache entries are evicted. Setting this parameter to a negative value disables the cache.
`-m value --max-cache-size=value`	Optional	10240	The maximum amount of disk space (in megabytes) used by the cache. Setting this parameter to a negative value disables the cache.
`-E <true|false> --encryption-enabled=<true|false>`	Optional	true	Specifies whether the encryption of temp files is enabled.
`-k value --encryption-key-size=value`	Optional	128	The size of the encryption key used when encrypting temp files.

config-auth

Configures authentication mode and default domain.

```
config-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-a value | --auth-method=value]
[-d | --jaas-database]
[-l | --jaas-ldap]
[-w | --jaas-windows]
[-j value | --jaas-custom=value]
```

```
[-D value | --default-domain=value]
[-p <true|false> | --parse-user-and-domain-name=<true|false>]
[-s value | --site-name=value]
```

Overview

Use this command to configure the authentication mode and to set the default domain.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-a value --auth-method=value`	Optional	none	The authentication method to use. The following methods are supported: BASIC, CLIENT_CERT, NTLM, Kerberos, WEB, and EXTERNAL. The names can be specified in either uppercase or lowercase.
`-d --jaas-database`	Optional	none	Use the Spotfire database authentication source, as configured in the Spotfire-DBLogin JAAS application configuration. This option is permitted only when using the BASIC authentication method. Also, it is mutually exclusive with all other options related to BASIC authentication sources.
`-l --jaas-ldap`	Optional	none	Use the LDAP authentication source, as configured in the SpotfireLDAP JAAS application configuration. This option is permitted only when using the BASIC authentication method. Also, it is mutually exclusive with all other options related to BASIC authentication sources.
`-w --jaas-windows`	Optional	none	Use the Windows NT authentication source, as configured in the SpotfireWindows JAAS application configuration. This option is permitted only when using the BASIC authentication method. Also, it is mutually exclusive with all other options related to BASIC authentication sources.
`-j value --jaas-custom=value`	Optional	none	Use the custom JAAS application configuration with the specified name. This option is permitted only when using the BASIC authentication method. Also, it is mutually exclusive with all other options related to BASIC authentication sources.
`-D value --default-domain=value`	Optional	SPOTFIRE	The name of the default domain. A user belonging to the default domain need not specify domain name as part of his or her user name when logging in to the server.
`-p <true|false> --parse-user-and-domain-name=<true|false>`	Optional	true	Indicates whether the user name consists of both a user and a domain part that should be parsed. it is recommended that you avoid changing the default value of "true", except when you are running the user directory in database mode, and the user names are in either

Option	Optional or Required	Default Value	Description
			NetBIOS name format (domain\user) or email name format (user@domain).
`-s value --site-name=value`	Optional	none	The name of the site for which the configuration should be applied. Any configuration made with this flag will affect only the specified site.

config-auth-filter

Configures the authentication filter.

```
config-auth-filter
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-f value | --filter-class=value]
{-Ikey=value}
[-s <true|false> | --skip-analyst=<true|false>]
```

Overview

Use this command to configure a custom authentication filter.

The Authentication Filter API is deprecated and will be removed in a future release.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-f value --filter-class=value`	Optional	none	The fully-qualified name of a class implementing the javax.servlet.Filter interface.
`-Ikey=value`	Optional	none	The initialization parameters provided to the filter when the init(FilterConfig) method is called. Can be specified multiple times with different keys.
`-s <true|false> --skip-analyst=<true|false>`	Optional	false	Indicates whether the Spotfire Analyst client should be handled by the custom authentication filter.

Example

To set the initialization parameter 'debug' to 'true': `config -Idebug=true`

config-basic-database-auth

Configures the Spotfire database authentication source to use the BASIC authentication method.

```
config-basic-database-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-p <true|false> | --parse-user-and-domain-name=<true|false>]
```

Overview

Use this command to configure the Spotfire database authentication source to use the BASIC authentication method. The configuration is stored in the SpotfireDatabase JAAS application configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-p <true|false> --parse-user-and-domain-name=<true|false>`			This argument is deprecated and is ignored. Use the config-auth command to set the global configuration property.

config-basic-ldap-auth

Configures the LDAP authentication source for use with the BASIC authentication method.

```
config-basic-ldap-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-l value | --ldap-configs=value]
[-w <true|false> | --enable-wildcard-domain=<true|false>]
```

Overview

Use this command to configure the LDAP authentication source to use the BASIC authentication method. The configuration is stored in the SpotfireLDAP JAAS application configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-l value --ldap-configs=value`	Optional	none	A comma-separated list of LDAP configuration references. All referenced LDAP configurations must already exist. To create a new LDAP configuration, use the create-ldap-config command. When specifying more than one reference, make sure to enclose the list of references in double quotes.
`-w <true|false> --enable-wildcard-domain=<true|false>`	Optional	none	Indicates whether the server should attempt to authenticate the user in all domains until an authentication attempt succeeds whenever the user omits the domain name in the account name credential.

config-basic-windows-auth

Configures the Windows NT authentication source to use the BASIC authentication method.

```
config-basic-windows-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-d value | --domains=value]
[-w <true|false> | --enable-wildcard-domain=<true|false>]
```

Overview

Use this command to configure the Windows NT authentication source to use the BASIC authentication method. The configuration is stored in the Spotfire Windows JAAS application configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-d value --domains=value`	Optional	none	A comma-separated list of domain names. When specifying more than one domain name, make sure to enclose the list of names in quotes.
`-w <true|false> --enable-wildcard-domain=<true|false>`	Optional	none	Indicates whether the server should attempt to authenticate the user in all domains until an authentication attempt succeeds whenever the user omits the domain name in the account name credential.

config-client-cert-auth

Configures the CLIENT_CERT authentication method.

```
config-client-cert-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name-attribute=value>
[-d <true|false> | --name-attribute-contains-domain=<true|false>]
```

Overview

Use this command to configure the X.509 certificate name attribute used for the CLIENT_CERT authentication method.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name-attribute=value`	Required	none	The name of the attribute used to extract user names from X.509 certificates. Supported attributes are: • Any attribute that can occur in the certificate subject's distinguished name (for instance "CN") • "DN" (use the whole distinguished name) • Any subject alternative name of type "rfc822Name", "dNSName", "directoryName", "uniformResourceIdentifier", "iPAddress", or "registeredID". To use a subject alternative name, make sure the name attribute has the prefix "subjectAltName:". If more than one subject alternative name is present in the certificates, you can add an index prefixed with a pound sign (#).
`d <true|false> --name-attribute-contains-domain=<true|false>`	Optional	false	Indicates whether the specified name attribute contains a fully-qualified account name, with both a user name part and a domain name part.

config-cluster

Configures clustering.

```
config-cluster
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[-p value | --port=value]
[-s <true|false> | --secure-transport=<true|false>]
```

Overview

Use this command to configure clustering.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	true	No op parameter left for backwards compatibility. The value defaults to true and does not need to be specified.
`-p value --port=value`	Optional	5701	The new value for TCP/IP port used for clustering. Shared among all nodes in cluster.
`-s <true|false> --secure-transport=<true|false>`	Optional	none	The secure transport flag used by Apache Ignite.

config-csrf-protection

Configures the CSRF protection.

```
config-csrf-protection
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-l <true|false> | --legacy-soap=<true|false>]
```

Overview

Use this command to configure the CSRF protection. When the `-l/--legacy-soap` argument isn't provided, the command displays the current configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-l <true|false>` `--legacy-soap=<true|false`	Optional	none	Specifies whether the CSRF protection should be enabled for the legacy SOAP clients.

config-custom-web-auth

Configures custom web authentication.

```
config-custom-web-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[-a value | --authenticator-class=value]
{-Ikey=value}
```

Overview

This command is used for configuring a custom web authenticator that implements a web-based authentication flow (for example, based on OAuth2).

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	true	Specifies whether custom web authentication should be enabled.
`-a value --authenticator-class=value`	Optional	none	The fully qualified name of a class implementing the com.spotfire.server.security.CustomWebAuthenticator interface.
`-Ikey=value`	Optional	none	Initialization parameters that will be provided to the custom web authenticator when the init(CustomWebAuthenticatorContext) method is called. If the name of the parameter ends with [SENSITIVE] it will be stored encrypted in the configuration. This argument may be specified multiple times with different keys.

Examples

To set the initialization parameter 'debug' to 'true': `-Idebug=true`

To set a sensitive parameter where the value should be stored encrypted: `-Iclient.secret[SENSITIVE]=secret123`

config-encryption

Configures the encryption of sensitive information such as service account passwords.

```
config-encryption
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-u | --update-encryption-password]
[-p value | --new-encryption-password=value]
[-n | --no-prompt]
[-f | --force]
```

Overview

Use this command to configure the encryption of sensitive information such as service account passwords, including changing the encryption password.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`-u --update-encryption-password`	Optional	none	When this flag is specified the encryption password will be updated.
`-p value --new-encryption-password=value`	Optional	none	The new encryption password. If no encryption password is given and the `--update-encryption-password` flag is given, then the tool will prompt for the password, unless the `--no-prompt` flag is given.
`-n --no-prompt`	Optional	none	When this flag is specified, the tool will not prompt for any missing password arguments.
`-f --force`	Optional	none	When this flag is specified, the encryption configuration will be updated even if the encryption password in the given bootstrap configuration file does not match the one currently in use.  Any previously configured secret passwords will have to be reconfigured if this option is used.

config-external-auth

Configures the external authentication method.

```
config-external-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[-m value | --declared-auth-method=value]
[-a value | --request-attribute=value]
[-r value | --request-header=value]
[-o value | --request-cookie=value]
[-n value | --custom-authenticator-class-name=value]
[-f <true|false> | --use-authentication-filter=<true|false>]
[-x value | --expression=value]
[-d <true|false> | --downcase=<true|false>]
[-s <true|false> | --require-tls=<true|false>]
[-h value | --allowed-hosts=value]
{-Rvalue}
{-Ikey=value}
```

Overview

This command is used to configure external authentication, which is typically used when a reverse-proxy or similar in front of the Spotfire Server handles authentication. The authentication method can either be used as the main authentication method, as configured by the config-auth command, or as a complementary authentication method where it is combined with the main method. It is typically used as the main method when the clients only can access the server(s) through a proxy or a load-balancer. It is typically used as a complementary method when the clients can access the server(s) both directly and through a proxy or a load-balancer. To use it as a complementary method, simply configure and enable the method using this command. To use it as the main authentication method, first configure and enable the method using this command and then set it to the main method using the **`config-auth`** command.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	true	Specifies whether the external authentication method should be enabled.
`-m value --declared-auth-method=value`	Optional	NTLM	The authentication method that should be declared to clients when external authentication is used as the main authentication method. The following methods are supported: CLIENT_CERT, NTLM, KERBEROS, and WEB.
`-a value --request-attribute=value`	Optional	REMOTE_USER	The name of an HTTP request attribute containing the name of the authenticated user. The `--request-attribute`, `--request-header`, `--request-cookie`, `--custom-authenticator-class-name`, and `--use-authentication filter` arguments are mutually exclusive.
`-r value --request-header=value`	Optional	none	The name of an HTTP header containing the name of the authenticated user. The `--request-attribute`, `--request-header`, `--request-cookie`, `--custom-authenticator-class-name`, and `--use-authentication filter` arguments are mutually exclusive.
`-o value --request-cookie=value`	Optional	none	The name of an HTTP cookie containing the name of the authenticated user. The `--request-attribute`, `--request-header`, `--request-cookie`, `--custom-authenticator-class-name`, and `--use-authentication filter` arguments are mutually exclusive.
`-n value --custom-authenticator-class-name=value`	Optional	none	The name of a class implementing the com.spotfire.server.security.CustomAuthenticator interface that should be used for authentication. Initialization parameters for the Custom Authenticator may be specified using the `-I` argument. The `--request-attribute`, `--request-header`, `--request-cookie`, `--custom-authenticator-class-name`, and `--use-authentication-filter` arguments are mutually exclusive.
`-f <true|false> --use-authentication-filter=<true|false>`	Optional	false	Specifies that the identity of the authenticated user is provided by a custom authentication filter (as the value of the getUserPrincipal<> method of javax.servlet.http.HttpServletRequest).

Option	Optional or Required	Default Value	Description
			 The Authentication Filter API is deprecated and will be removed in a future release; consider using a Custom Authenticator instead. The `--request-attribute`, `--request-header`, `--request-cookie`, `--custom-authenticator-class-name`, and `--use-authentication-filter` arguments are mutually exclusive.
`-x value` `--expression=value`	Optional	none	A regular expression that can be used to filter the username extracted from the specified HTTP request attribute. The value of the regular expression's first capturing group will be used as the new username. A typical scenario is to extract the username from a composite name containing both username and domain name when using the "collapse domains" option. For example, the regular expression "\S+\\<\S+>" can be used to extract the username from a value in the format "domain\username". Make sure to enclose the specified expression in quotes and to quote all special characters that might otherwise be consumed by the command-line shell.
`-d <true|false>` `--downcase=<true|false>`	Optional	false	Specifies whether the username should be converted to lower case.
`-s <true|false>` `--require-tls=<true|false>`	Optional	false	Specifies whether a secure HTTPS connection is required to perform external authentication.
`-h value` `--allowed-hosts=value`	Optional	none	A comma-separated list of hostnames and/or IP addresses of the client computers that are permitted to perform external authentication. If this, or at least one -R argument, is not specified, then all client computers are permitted to perform external authentication. Because this is a potential security risk, it is strongly recommended to restrict the permissions to use this feature. Typically, this feature is locked down so that only proxies or load balancers are permitted to use it. A scenario where all client computers can be allowed to use this feature is when a custom post-authentication filter is also in use. Then this filter would be responsible for performing the final authorization, for example by validating additional HTTP headers.
`-Rvalue`	Optional	none	A regular expression (in the syntax supported by java.util.regex.Pattern) that should match IP addresses of remote hosts that are permitted to perform external authentication. See also the `--`

Option	Optional or Required	Default Value	Description
			`allowed-hosts` argument. This argument can be specified multiple times with different values.
`-Ikey=value`	Optional	none	Specifies initialization parameters that will be provided to the Custom Authenticator when the init(Map<String, String>) method is called. This argument can only be specified together with the `--custom-authenticator-class-name` argument, and may be specified multiple times with different keys. Example: To set the Custom Authenticator initialization parameter "debug" to "true": `-Idebug=true`

config-external-ignite-process

Configures memory settings and JVM options for the external Ignite process.

```
config-external-ignite-process
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--max-memory=value]
[--min-memory=value]
{--jvm-optsvalue}
```

Overview

Use this command to configure memory settings and JVM options for the external Ignite process.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	`configuration.xml`	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--max-memory=value`	Optional	512M	Specifies the maximum memory that is used by the Ignite server process.
`--min-memory=value`	Optional	512M	Specifies the initial, minimum memory that is used by the Ignite server process.
`--jvm-optsvalue`	Optional	• `-XX:+AlwaysPreTouch` • `-XX:+UseG1GC -` • `XX: +ScavengeBeforeFullGC` • `-XX:+DisableExplicitGC`	Specifies the JVM options to use for the Ignite server process. Can be specified multiple times with different values.

config-external-scheduled-updates

Configures external scheduled updates.

```
config-external-scheduled-updates
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --ems-enabled=<true|false>]
[-s value | --server-url=value]
[-u value | --username=value]
[-p value | --password=value]
[-i value | --client-id=value]
[-t value | --topic=value]
[-C value | --reconnect-attempt-count=value]
[-D value | --reconnect-attempt-delay-milliseconds=value]
[-T value | --reconnect-attempt-timeout-milliseconds=value]
[-k value | --keep-alive-minutes=value]
[-S value | --site-name=value]
```

Overview

Use this command to configure external scheduled updates via web service or TIBCO EMS.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --ems-enabled=<true|false>`	Optional	false	The value should be "true" if updates triggered by a message sent from TIBCO Enterprise Message Service is enabled.
`-s value --server-url=value`	Optional, unless `--ems-enabled` is "true"	none	The URL and, if applicable, the port to the EMS server.
`-u value --username=value`	Optional	none	The name of the user that will be used to access the EMS server.
`-p value --password=value>`	Optional	none	The password of the user that will be used to access the EMS server.
`-i value --client-id=value`	Optional, unless `--ems-enabled` is "true"	none	A unique value to identify the EMS connection. If using multiple sites, a unique value should be assigned to each site.
`-t value --topic=value`	Optional, unless `--ems-enabled` is "true"	none	The topic that the EMS durable subscriber should listen to.
`-C value --reconnect-attempt-count=value`	Optional	10	The number of reconnect attempts to be made if a connect fails.
`-D value --reconnect-attempt-delay-milliseconds=value`	Optional	1000	The delay for the reconnect attempts.
`-T value --reconnect-attempt-timeout-milliseconds=value`	Optional	1000	The timeout for the reconnect attempts.
`-k value`	Optional	10	If a schedule has not been set up for when a file will be pre-loaded, specify the number of minutes the file should be kept alive.

Option	Optional or Required	Default Value	Description
`--keep-alive-minutes=value`			
`-S value` `--site-name=value`	Optional	none	The name of the site for which the configuration should be applied. Any configuration made with this flag will affect only the specified site. If a site is not given, the EMS configuration will apply to all the sites.

config-import-export-directory

Configures the library import/export directory.

```
config-import-export-directory
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-p value | --path=value]
```

Overview

Use this command to configure the library import/export directory. All library import and export operations are performed from or to this directory. It can be a local directory, or it can reside on a shared disk.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-p value` `--path=value`	Optional	`<installation directory>/tomcat/application-data/library`	The path to the import/export directory.

config-jmx

Configures the JMX RMI connector.

```
config-jmx
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[-a <true|false> | --authentication-enabled=<true|false>]
[-A <true|false> | --authorization-enabled=<true|false>]
[-s <true|false> | --tls-enabled=<true|false>]
[-n <true|false> | --need-client-auth=<true|false>]
[-R value | --registry-port=value]
[-p value | --connector-port=value]
```

```
[-j value | --jaas-config=value]
```

Overview

Use this command to configure the JMX RMI connector. This connector can be used for connecting to Spotfire Server for monitoring and management purposes.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	false	Specifies whether the RMI connector is enabled.
`-a <true|false> --authentication-enabled=<true|false>`	Optional	true	Specifies whether authentication is enabled for the RMI connector.
`-A <true|false> --authorization-enabled=<true|false>`	Optional	true	Specifies whether authorization is enabled for the RMI connector. Authorization requires authentication to be enabled and works only with the default value of jaas-config.
`-s <true|false> --tls-enabled=<true|false>`	Optional	false	Specifies whether TLS is enabled for the RMI connector.
`-n <true|false> --need-client-auth=<true|false>`	Optional	false	Specifies whether TLS client authentication is required.
`-R value --registry-port=value`	Optional	1099	The port for the RMI registry.
`-p value --connector-port=value`	Optional	1099	The port for the RMI connector.
`-j value --jaas-config=value`	Optional	SpotfireJmx	The JAAS configuration entry to use for authentication. Requires authentication to be enabled. User accounts for the default authentication implementation are created by the create-jmx-user command.

config-kerberos-auth

Configures the authentication service used with the Kerberos authentication method.

```
config-kerberos-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

```
[-S value | --server=value]
[-p value | --service-principal-name=value]
[-k value | --keytab-file=value]
[-r value | --krb5-conf-file=value]
[-d <true|false> | --enable-debug=<true|false>]
[-w value | --worker-delegation-policy=value]
```

Overview

Use this command to configure the authentication service used with Kerberos authentication method.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-S value` `--server=value`	Optional	none	The name of the cluster server to which the specified configuration parameters should be applied. If no name is specified, the parameters apply to all servers in the cluster.
`-p value` `--path=value`	Required	none	The Kerberos service principal name (SPN) used by the server.
`-k value` `--keytab-file=value`	Optional	${catalina.base}/spotfire-config/spotfire.keytab	The path to the Kerberos file containing the keytab entry for the specified SPN. If the specified path contains any Java system properties (for example, as in the default value for this argument), they are automatically expanded.
`-r value` `--krb5-conf-file=value`	Optional	${catalina.base}/spotfire-config/krb5.conf	The path to the Kerberos file containing the Kerberos configuration (`krb5.conf`). If the specified path contains any Java system properties (e.g. as in the default value for this argument), they will automatically be expanded.
`-d <true|false>` `--enable-debug=<true|false>`	Optional	false	Specifies whether extra debug logging should be enabled for the Kerberos authentication service.
`-w value` `--worker-delegation-policy=value`	Optional	none	Configures how delegation of Kerberos credentials should be handled when connecting to a service on a node. When a user's credentials are delegated to a service, the service can in turn use these credentials to connect to data sources, assuming the identity of the user. Connections made without delegation can be configured to use impersonation. There are three options: • REQUIRE - Do not connect to a service unless delegation succeeds. • TRY - Try delegation; if that fails, log in with impersonation. • NEVER - Do not attempt to delegate; always log in with impersonation. By default, Spotfire Server uses the REQUIRE option.

config-ldap-group-sync

Configures group synchronization for an LDAP configuration.

```
config-ldap-group-sync
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<--id=value>
[--group-sync-enabled=<true|false>]
[--schedules=value]
[--clear-schedules]
[--group-names=value]
[--clear-group-names]
[--clear-all]
[--filter-users-by-groups=<true|false>]
[--group-search-filter=value]
[--group-name-attribute=value]
[--supports-member-of=<true|false>]
[--member-attribute=value]
[--ignore-member-groups=<true|false>]
```

Overview

Use this command to configure group synchronization for an LDAP configuration used with the User Directory LDAP provider.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--id=value`	Required	none	Specifies the identifier of the LDAP configuration for which to configure group synchronization.
`--group-sync-enabled=<true|false>`	Optional	true	Specifies whether group synchronization should be enabled for this LDAP configuration.
`--schedules=value`			This argument is deprecated and is replaced with the similarly named argument for the create-ldap-config and update-ldap-config commands, because the synchronization schedules are now used for both user and group synchronization. The argument specifies a comma-separated list of schedules for when the LDAP synchronization should be performed. The schedules are given in a cron-compatible format, where each schedule consists of either five fields or one shorthand label. The five fields are, from left to right, with their valid ranges: minute (0-59), hour (0-23), day of month (1-31), month (1-12) and day of week (0-7, where both 0 and 7 indicate Sunday). A field may also be configured with the wildcard character '*', indicating that any moment in time matches this field. An LDAP synchronization is triggered when all fields match the current time. If both day of month and day of week have non-wildcard values, then only one of them has to match. There are also the following shorthand labels that can be used instead of the full cron expressions: @yearly or @annually: run once a year (equivalent to 0 0 1 1 *) @monthly: run once a month (equivalent to 0 0 1 * *) @weekly: run once a week (equivalent to 0 0 * * 0) @daily or @midnight: run once a day (equivalent to 0 0 * * *) @hourly: run once an hour (equivalent to 0 * * * *) @minutely: run once a minute (equivalent to * * * * *)

Option	Optional or Required	Default Value	Description
			@reboot or @restart: run every time the Spotfire Server is started Consult the Wikipedia article for an overview of the cron scheduler: https://en.wikipedia.org/wiki/Cron.
`--clear-schedules`	Optional		This argument is deprecated and is replaced with the similarly named argument for the update-ldap-config command because the synchronization schedules are now used for both user and group synchronization. By specifying this flag, the LDAP synchronization schedules are cleared from the LDAP configuration. This flag can be used together with the `--schedules` flag to remove all old schedules before adding the new.
`--group-names=value`	Optional	none	Specifies the account names or the distinguished names (DNs) of the groups to be synchronized. When you specify more than one account name or DN, you must separate these using pipe characters (|).
`--clear-group-names`	Optional	none	By specifying this flag, the list of group names to be synchronized are cleared from the LDAP configuration. This flag can be used together with the `--group-names` flag to remove all old group names before adding the new.
`--clear-all`	Optional	none	By specifying this flag, all group synchronization related configuration options are cleared from the LDAP configuration. Starting from Spotfire Server 5.0, it will NOT clear the LDAP synchronization schedules.
`--filter-users-by-groups=<true|false>`	Optional	none	Specifies whether users should be filtered by groups, so that only users who are members of the synchronized groups are synchronized.
`--group-search-filter=value`	Optional, unless the LDAP server type is set to "Custom" using the `--type` parameter.	For Active Directory servers, the parameter value defaults to `objectClass=group`. For Sun ONE Directory Servers, it defaults to `&(|(objectclass=nsManagedRoleDefinition)(objectClass=nsNestedRoleDefinition))(objectclass=ldapSubEntry)`. For Sun Java System Directory Servers, it defaults to `objectClass=groupOfUniqueNames`..	Specifies an LDAP search expression filter to use when searching for groups.
`--group-name-attribute=value`	Optional, unless the LDAP server type is set to	For Active Directory servers, the	Specifies the name of the LDAP attribute containing the group account names.

Option	Optional or Required	Default Value	Description
	"Custom" using the `--type` parameter.	value defaults to sAMAccountName. For any version of the Sun Directory Servers with a default configuration, it defaults to cn.	
`--supports-member-of=<true|false>`	Optional, unless the LDAP server type is set to "Custom" using the `--type` parameter.	none	Specifies whether the LDAP servers support a `memberOf`-like attribute on the user accounts that contain the names of the groups or roles that the users are members of. In general, this is true for all Microsoft Active Directory servers and all types of Sun Directory Servers.
`--member-attribute=value`	Optional, unless the LDAP server type is set to "Custom" using the `--type` parameter.	For Microsoft Active Directory servers, the parameter value defaults to memberOf. For Sun ONE Directory Servers, it defaults to nsRole. For Sun Java System Directory Server version 6.0 or later, it defaults to isMemberOf. To use the roles with the Sun Java System Directory Server, override the default value by setting this argument to "nsRole". For some LDAP servers with configurations of type 'Custom', there is no memberOf-like attribute. In those cases, this argument specifies the LDAP attribute on the group account that contains the names of its members. Note that all configurations of this type will use a far less efficient group	For all LDAP servers with support for a `memberOf`-like attribute, this argument specifies the name of the LDAP attribute on the user account that contains the names of the groups or roles that the user is a member of. In general, this includes all Microsoft Active Directory servers and all types of Sun Directory Servers. For some LDAP servers with configurations of type Custom, there is no memberOf-like attribute. In those cases, this argument specifies the LDAP attribute on the group account that contains the names of its members. All configurations of this type use a far less efficient group synchronization algorithm that generates more traffic to the LDAP servers because Spotfire Server first has to search for the distinguished names (DNs) of the group members within the groups, and then perform repeated look-ups to translate the member DN to the correct account name.

Option	Optional or Required	Default Value	Description
		synchronization algorithm that will generate more traffic to the LDAP servers, because the Spotfire Server will first have to search for the distinguished names (DNs) of the group members within the groups, and then perform repeated lookups to translate the member DN to the correct account name.	
`--ignore-member-groups=<true|false>`	Optional, unless the LDAP server type is set to "Custom" using the `--type` parameter.	For Microsoft Active Directory servers, the parameter value defaults to "false" so all inherited group memberships are correctly reflected. For any version of the Sun Directory Servers, it defaults to "true" because the role and groups mechanisms in those servers automatically include those members.	Determines whether the group synchronization mechanism should recursively traverse the synchronized groups' non-synchronized subgroups and include their members in the search result.

config-ldap-userdir

Configures the LDAP user directory mode.

```
config-ldap-userdir
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-l value | --ldap-configs=value]
[-s <true|false> | --group-sync-enabled=<true|false>]
[-t value | --sleep-time=value]
```

Overview

Use this command to configure the LDAP user directory mode. If no arguments are specified, the command displays the current configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-l value --ldap-configs=value`	Optional	none	A comma-separated list of LDAP configuration references. All referenced LDAP configurations must already exist. To create a new LDAP configuration, use the create-ldap-config command. When specifying more than one reference, make sure to enclose the list of references in quotes.
`-s <true|false> --group-sync-enabled=<true|false>`	Optional	none	This argument is deprecated and is ignored. Use the config-ldap-group-sync command to enable or disable group synchronization for each LDAP configuration instead.
`-t value --sleep-time=value`	Optional	60	The number of minutes between each synchronization. The sleep time setting is used only for LDAP configuration entries without group synchronization schedules. If an LDAP configuration entry has a synchronization schedule defined, then this value is ignored.

config-library-external-data-storage

Configures the external library data storage.

```
config-library-external-data-storage
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-e <true|false> | --enabled=<true|false>>
[-s value | --external-storage=value]
[-f | --force]
```

Overview

Use this command for general configuration of the external library data storage.

When this feature is enabled, the structure of the library is stored in the Spotfire database, while the actual data of library items is stored elsewhere.

The library must be empty when you switch to or from an external data storage. The prescribed procedure for switching is to export the entire library, empty the library, change the configuration, and then import the library. Switching storage modes with items in the library causes data to be lost.

When you change the external library data storage configuration with this command, a query is made to the Spotfire database to make sure that the library is empty. This check can be overridden by using the `--force` argument.

Currently, Spotfire supports two options for external data storage: storing on the server's file system, or storing on Amazon S3. After enabling this feature, you must configure the storage using the config-library-external-file-storage command or config-library-external-s3-storage command.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-e <true|false> --enabled=<true|false>>`	Required	none	Specifies whether external library data storage should be enabled.
`-s value --external-storage=value`	Optional	none	The external storage to use. The following names are valid: FILE_SYSTEM and AMAZON_S3.
`-f --force`	Optional	none	Indicates that the tool should change the library configuration even if the library is not empty.

config-library-external-file-storage

Configures the file system storage of library item data.

```
config-library-external-file-storage
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-p value | --path=value>
```

Overview

Use this command for configuring file system storage of library data.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-p value --path=value`	Required	none	The path to the directory where library data is stored. Supply the value "DEFAULT" to use the Spotfire Server default location for storing library data on file system.

config-library-external-s3-storage

Configures the Amazon S3 storage of library item data.

```
config-library-external-s3-storage
[-c value | --configuration=value]
[--region=value]
[-b value | --bootstrap-config=value]
[--bucket-name=value]
[--key-prefix=value]
[--access-key=value]
[--secret-key=value]
[--endpoint=value]
[--threads=value]
[--chunk-size=value]
[--threshold=value]
```

Overview

Use this command for configuring the Amazon S3 storage of library data.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	`configuration.xml`	The path to the server configuration file.
`--region=value`	Optional	If not explicitly configured, the default region will be used.	The Amazon AWS region to connect to, for example `eu-central-1`. To clear an already configured value, use `--region="NONE"`. If not explicitly configured, an attempt will be made to determine the region from the AWS_REGION system variable and the `~/.aws/config` file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--bucket-name=value`	Optional	none	The Amazon S3 bucket where library data is stored.
`--key-prefix=value`	Optional	none	The optional key prefix to use for allowing data to be stored in the equivalent of a top level folder. The following restrictions are applied: • It must not start with a slash . • It must end with a slash (/). • Valid characters are A-Z, a-z, 0-9, and / (slash).
`--access-key=value`	Optional	none	The access key for connecting to Amazon S3. If set to `default`, an instance of `DefaultAWSCredentialsProviderChain` is created. `DefaultAWSCredentialsProviderChain` can take authentication tokens from environment variables, Java system properties, by way of a `config` file, through the Amazon EC2 container, or through instance profile credentials delivered through the Amazon EC2 metadata service. For more information see the documentation for `DefaultAWSCredentialsProviderChain`.
`--secret-key=value`	Optional	none	The secret key for connecting to Amazon S3.
`--endpoint=value`	Optional	If not explicitly configured, the default region is used.	The Amazon S3 endpoint to connect to. For example, `https://s3.eu-central-1.amazonaws.com`. To clear an already configured value, use `--endpoint="NONE"`.

Option	Optional or Required	Default Value	Description
			 A signing region must also be specified (using the `--region` flag) when setting an endpoint.
`--threads=value`	Optional	none	The maximum number of threads used for uploading to Amazon S3.
`--chunk-size=value`	Optional	none	The maximum number of bytes in a chunk when the data is chunked before transfer to Amazon S3.
`--threshold=value`	Optional	none	Above this value, the number of bytes for the transferred data is split into chunks of a configurable size that are then transferred separately to Amazon S3.

config-login-dialog

Configures the client login dialog behavior.

```
config-login-dialog
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-s value | --show-login-dialog=value]
[-o <true|false> | --allow-work-offline=<true|false>]
[-d value | --offline-days-permitted=value]
[-r <true|false> | --allow-remember-me=<true|false>]
[-u <true|false> | --allow-user-provided-credentials=<true|false>]
[-R value | --rss=value]
```

Overview

Use this command to configure the behavior of the client login dialog.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-s value --show-login-dialog=value`	Optional	standard	Controls whether the log in dialog should be displayed. Valid values are: • always: Show the dialog even if the user selected **Save my login information**. • standard: Show the dialog only if the user did not select **Save my login information**.
`-o <true|false> --allow-work-offline=<true|false>`	Optional	true	Controls whether users should be allowed to work offline or if they must always log in.
`-d value --offline-days-permitted=value`	Optional	-1	Controls how long users can choose to work offline before they are forced to log in. Setting the value to `-1` means that users are never forced to connect to Spotfire Server.
`-r <true|false> --allow-remember-me=<true|false>`	Optional	true	Controls whether a user can select to store the log in information for future automatic login, or if he or she must always provide username and password when logging in.
`-u <true|false> --allow-user-provided-credentials=<true|false>`	Optional	true	Controls whether users should be able to enter their own credentials in the login dialog.
`-R value --rss=value`	Optional	none	The URL to an RSS feed to be shown in the login dialog. The URL may be either an absolute URL or a relative URL (/spotfire/rss.xml) on the Spotfire Server. The feed must be RSS 2.0 compliant. Note that HTML in the RSS feed is not supported.

config-ntlm-auth

Configures the authentication service used with the NTLM authentication method.

```
config-ntlm-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-S value | --server=value]
[-d value | --domain-name=value]
[-D value | --domain-controller=value]
[-a value | --account-name=value]
[-p value | --password=value]
```

```
[-n value | --dns-servers=value]
[-s value | --ad-site=value]
[-t value | --dns-cache-ttl=value]
[-i value | --connection-id-header-name=value]
[-L value | --log-level=value]
{-Pkey=value}
[-C value | --domain-trust-cache-values=value]
```

Overview

Use this command to configure the authentication service used with NTLM authentication method.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-S value --server=value`	Optional	none	The name of the cluster server to which the specified configuration parameters should be applied. If no name is specified, the parameters apply to all servers in the cluster. It is typically used to add a server-specific account name (see the `--account-name` option).
`-d value --domain-name=value`	Required, unless the `--domain-controller` argument is specified, or if the `--server` argument is specified and this parameter is already specified for the global configuration.	none	The DNS name of the Windows domain. The specified domain name automatically resolves into domain controller hostnames. It is also possible to use the `--domain-controller` argument to specify a domain controller hostname directly. The `--domain-name` and `--domain-controller` arguments are mutually exclusive.
`-D value --domain-controller=value`	Required, unless the `--domain-controller` argument is specified, or if the `--server` argument is specified and this parameter is already specified for the global configuration.	none	The DNS hostname of an Active Directory domain controller. It is also possible to use the `--domain-name` argument to specify a domain name that automatically resolves to domain controller hostnames. The `--domain-name` and `--domain-controller` arguments are mutually exclusive.
`-a value --account-name=value`	Optional, unless the **`--server`** argument is specified and this parameter is not already specified for the global configuration.	none	Specifies the fully qualified name of the Active Directory computer account to be used by the NTLM authentication service. This account must be a proper computer account created solely for the purpose of running the NTLM authentication service. It can neither be an ordinary user account, nor an account of an existing computer. Note that the name of an Active Directory computer account always contains a dollar sign, for example, ntlm-svc$@research.example.com. The local part of the account name (excluding the dollar sign) must not exceed 15 characters. On Linux, the parameter value must be enclosed in single quotes because of the dollar sign.

Option	Optional or Required	Default Value	Description
			If there is more than one server in the cluster, each server must use its own account. It is recommended to leave the global configuration without account name and password, and only add them to each server's configuration.
`-p value` `--password=value`	Optional, unless the **`--server`** argument is specified and this parameter is not already specified for the global configuration.	none	Specifies the password for the computer account that is to be used by the NTLM authentication service. It is recommended to leave the global configuration without account name and password, and only add them to each server's configuration.
`-n value` `--dns-servers=value`	Optional	none	A comma-separated list of IP addresses for the DNS servers associated with the Windows domain. When no DNS servers are specified, the NTLM authentication service falls back to the server computer default DNS server configuration.
`-s value` `--ad-site=value`	Optional	none	The Active Directory site where the Spotfire system is located. Specifying an Active Directory site can potentially improve performance because the NTLM authentication service then communicates only with the local domain controllers.
`-t value` `--dns-cache-ttl=value`	Optional	5000 ms.	The length of time (in milliseconds) name server lookups should be cached.
`-i value` `--connection-id-header-name=value`	Optional	none	The name of an HTTP header containing unique connection IDs in environments where the server is located behind a proxy or load-balancer that does not properly provide the server with the client IP address. The specified HTTP header must contain unique connection IDs for each client connection and is thus typically based on the client IP address and the connection port number on the client side.
`-L value` `--log-level=value`	Optional	1	Specifies the level of logging done for NTLM authentication, an integer value ranging from 0 (no logging) to 4 (debug logging).
`-Pkey=value`	Optional	none	Specifies additional properties for the Jespa component, in the form of key-value-pairs. For example: `-Pjespa.key=value`. This argument may be specified multiple times with different keys.
`-C value` `--domain-trust-cache-values=value`	Optional	none	Specifies a mapping between NetBIOS and DNS domain names used for canonicalizing domain names when sufficient information is not provided by the local NETLOGON service. The mapping is given as a comma-separated list of NetBIOS:DNS entries, for example "RESEARCH:research.example.com,HR:hr.example.com",

Option	Optional or Required	Default Value	Description
			and is used for turning a NetBIOS name into a DNS name, or vice versa.

Examples

- Configuring the NTLM authentication service for the research.example.com Windows domain

  Windows command prompt:

  ```
  config config-ntlm-auth --domain-name research.example.com --
  account-name ntlm-svc$@research.example.com --password 53cr3t
  ```

  Linux command shell:

  ```
  config config-ntlm-auth --domain-name research.example.com
   --account-name 'ntlm-svc$@research.example.com' --password
   53cr3t
  ```

- Configuring the NTLM authentication service for using the Active Directory Domain Controller dc.research.example.com

  Windows command prompt:

  ```
  config config-ntlm-auth --domain-controller
   dc.research.example.com --account-name ntlm-svc
  $@research.example.com --password 53cr3t
  ```

  Linux command shell:

  ```
  config config-ntlm-auth --domain-controller
   dc.research.example.com --account-name 'ntlm-svc
  $@research.example.com' --password 53cr3t
  ```

- Configuring the NTLM authentication service for the Active Directory Site VIENNA within the research.example.com Windows domain

  Windows command prompt:

  ```
  config config-ntlm-auth --domain-name research.example.com --
  ad-site=VIENNA --account-name ntlm-svc$@research.example.com
   --password 53cr3t
  ```

  Linux command shell:

  ```
  config config-ntlm-auth --domain-name research.example.com --
  ad-site=VIENNA --account-name 'ntlm-svc$@research.example.com'
   --password 53cr3t
  ```

config-oidc

Configures authentication using OpenID Connect.

```
config-oidc
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[--third-party-login-init-enabled=<true|false>]
[-s | --set-provider]
[-r | --remove-provider]
[-n value | --provider-name=value]
[--provider-enabled=<true|false>]
[--provider-discovery-url=value]
```

```
[--provider-client-id=value]
[--provider-client-secret=value]
[--provider-domain-option=value]
[--provider-domain-name=value]
[--provider-username-claim=value]
[--provider-display-name-claim=value]
[--provider-email-claim=value]
[--provider-domain-claim=value]
[--provider-id-token-signing-alg=value]
[--provider-id-token-signature-verification-disabled=<true|false>]
[--provider-token-endpoint-auth-method=value]
{-Svalue}
[--provider-auth-request-prompt-value=value]
[--provider-clear-custom-params]
{-Pkey=value}
[--provider-bg-color=value]
```

Overview

Use this command to configure authentication against one or more external providers using OpenID Connect. Authentication using OpenID Connect may be combined with username/password-based authentication and/or custom web authentication.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	`configuration.xml`	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	true	Specifies whether OpenID Connect should be enabled.
`-s --set-provider`	Optional	none	Indicates that a provider configuration should be set (replaces the configuration for any existing provider with the same name). Cannot be specified together with `--remove-provider`.
`--third-party-login-init-enabled=<true|false>`	Optional	true	Specifies whether Third Party Login Initiation should be enabled.
`-r --remove-provider`	Optional	none	Indicates that a provider configuration should be removed. Cannot be specified together with `--set-provider`.
`-n value --provider-name=value`	This argument is optional unless either `--set-provider` or `--remove-provider` has been specified.	none	The name of the provider to set or remove. Normally displayed to end users on the login page.
`--provider-enabled=<true|false>`	This argument is optional unless `--set-provider` has been specified.	true	Specifies whether the provider should be enabled.
`--provider-discovery-url=value`	This argument is optional unless `--set-provider` has been specified.	none	The URL to the provider's OpenID Connect Discovery document.
`--provider-client-id=value`	This argument is optional unless `--set-`	none	The client ID given by the provider during registration.

Option	Optional or Required	Default Value	Description
	`provider` has been specified.		
`--provider-client-secret=value`	This argument is optional unless `--set-provider` has been specified.	none	The client secret given by the provider during registration.
`--provider-domain-option=value`	Optional	use_domain_claim	The way the domain of authenticated users will be established. Can be one of the following. • `use_domain_claim` • `use_static_domain` • `parse_username_claim`
`--provider-domain-name=value`	This argument is optional unless the value of the `--provider-domain-option` is 'use_static_domain'.	By default the value of the 'issuer' claim is used.	The domain name to assign to the authenticated users.
`--provider-username-claim=value`	Optional	sub	The name of the claim to use as username for the authenticated users. Can be `email`, for example. The name of the claim is case sensitive.  Only `sub` is guaranteed to be a unique and stable identifier.
`--provider-display-name-claim=value`	Optional	name	The name of the claim to use as the display name for the authenticated users. The name of the claim is case sensitive.
`--provider-email-claim=value`	Optional	By default, all algorithms listed as supported in the Discovery Document will be accepted.	The name of the claim to use as email address for the authenticated users. The name of the claim is case sensitive.
`--provider-username-claim=value`	Optional	sub	The name of the claim to use as username for the authenticated users. Can be `email`, for example.  Only `sub` is guaranteed to be a unique and stable identifier.
`--provider-id-token-signing-alg=value`	Optional	By default, all algorithms listed as supported in the	The ID token signature algorithm to expect.

Option	Optional or Required	Default Value	Description
		Discovery Document will be accepted.	
`--provider-id-token-signature-verification-disabled=<true|false>`	Optional	false	Indicates that signature verification of ID tokens should be disabled. This should normally only be specified if the provider does not sign the ID tokens.
`--provider-token-endpoint-auth-method=value`	Optional	By default one of the algorithms listed as supported in the Discovery Document will be used.	The authentication method to use when communicating with the provider's Token Endpoint. Can be one of the following. • `client_secret_basic` • `client_secret_post` • `client_secret_jwt` `private_key_jwt` is not supported.
`-Svalue`	Optional	`openid`, `profile`, `email`	A scope to include in the authentication request (besides `openid`, that is always included). This argument can be specified multiple times with different values.
`--provider-auth-request-prompt-value=value`	Optional	By default the parameter will be omitted from the request.	The value to give the `prompt` request parameter when making the authentication request. Controls how the provider prompts the end user. May be one of the following. • `none` • `login` • `consent` • `select_account`
`--provider-clear-custom-params`	Optional	none	Custom parameters are cleared from the provider configuration. This flag can be used together with the `-Pkey` flag to remove all old custom parameters before adding the new.
`-Pkey=value`	Optional	none	A custom parameter included in the authentication request. Must not be any of the parameters controlled through other settings (such as `scope` or `prompt`). Can be specified multiple times with different keys.
`--provider-bg-color=value`	Optional	none	The background color of the provider's button on the login page (when applicable), as a hexadecimal color value.

config-persistent-sessions

Configures the persistent sessions ("remember me") feature.

```
config-persistent-sessions
```

```
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
[-t value | --expiration-time=value]
[-s <true|false> | --sliding-expiration=<true|false>]
```

Overview

Use this command to configure the persistent sessions feature. Persistent sessions allows users to be remembered after a successful login. This means that the user will not have to log in again for a period of time (even if the user, for example, closes the browser).

This feature is only applicable when using username and password based authentication.

This feature is currently only applicable for users (such as Spotfire Web Player users) logging in through a web browser. To configure the behavior of the Spotfire client, use the config-login-dialog command.

Persistent sessions can be invalidated using the invalidate-persistent-sessions command.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false>` `--enabled=<true|false>`	Optional	true	Specifies whether the persistent sessions feature should be enabled.
`-t value` `--expiration-time=value`	Optional	2592000	Specifies the time in seconds until a persistent session will expire and the user will have to re-authenticate.
`-s <true|false>` `--sliding-expiration=<true|false>`	Optional	false	Specifies whether the expiration time should be reset each time the user is authenticated using the persistent session cookie. Note that setting this to "true" means that the user may actually never have to log in again.

Related concept

Persistent Spotfire sessions

config-post-auth-filter

Configures the post-authentication filter.

```
config-post-auth-filter
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-f value | --filter-class=value]
[-s value | --filter-config=value]
```

```
{-Ikey=value} [--clear-init-parameters]
[-d value | --default-filter-config=value]
```

Overview

Use this command to configure the post-authentication filter. If no argument is provided, the command simply lists the current configuration and exits.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-f value --filter-class=value`	Optional	none	The fully-qualified name of the class implementing the com.spotfire.server.security.PostAuthenticationFilter API. If the argument is `none`, the current value of this configuration option is cleared.
`-s value --filter-config=value`	Optional	none	This argument is deprecated and will be removed in a future release. Please use initialization parameters instead. The filter configuration. The semantics of the configuration argument is specific to the actual filter implementation. For example, it could be a configuration name, a file name, or a list of key/value pairs. If the argument is `none`, the current value of this configuration option is cleared.
`-Ikey=value`	Optional, and may be specified multiple times with different keys.	none	This argument specifies initialization parameters that will be provided to the PostAuthenticationFilter when the init(PostAuthenticationFilterInitContext) method is called. If the name of the parameter ends with [SENSITIVE] it will be stored encrypted in the configuration. Example of how to set the parameter **`debug`** to 'true' and the sensitive parameter **`secret`** to 'changeme': `-ldebug=true -Isecret[SENSITIVE]=changeme`
`--clear-init-parameters`	Optional	none	By specifying this flag, the list of initialization parameters is cleared. This flag can be used together with the `-I` argument to remove all old initialization parameters before adding the new ones.
`-d value --default-filter-config=value`	Optional	none	The configuration for the default filter that is always in place. Valid arguments are `block` and `autocreate`.

THE DEFAULT FILTER IMPLEMENTATION

The default implementation of the post authentication filter can be used for access control if you are using an external authentication source, such as LDAP or Windows NT Domain, in combination with the Database User Directory mode. If you are using a different combination of authentication and user directory, the filter has no effect.

The default implementation has two different modes:

- The user is allowed access only if the user already exists in the user directory (to configure this use `--default-filter-config=block`).
- The user is allowed access regardless of whether the user already exists in the user directory. The user will then be added to the user directory (to configure this use `--default-filter-config=autocreate`).

EXAMPLES

Example of how to configure the default filter to block users not in the user directory (the default behavior):

```
config-post-auth-filter --default-filter-config=block
```

Example of how to configure the default filter to automatically create users not in the user directory:

```
config-post-auth-filter --default-filter-config=autocreate
```

Example of how to configure a custom filter implementation with two initialization parameters, 'debug' and 'secret':

```
config-post-auth-filter --filter-class=com.example.MyPostAuthenticationFilter -Idebug=true -
Isecret[SENSITIVE]=changeme
```

config-public-address

This command has been replaced by **`set-public address`**.

See set-public-address.

config-scheduled-updates-retries

Configures scheduled updates retries.

```
config-scheduled-updates-retries
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-u value | --update-interval-seconds=value]
[-s <true|false> | --stop-updates-after-repeated-fail-enabled=<true|false>]
[-f value | --fails-before-stop=value]
[-o <true|false> | --stop-only-when-cached=<true|false>]
[-a <true|false> | --always-retry-when-scheduled=<true|false>]
[-d <true|false> | --stop-updates-destination-unavailable=<true|false>]
```

Overview

Use this command to configure scheduled updates retries following update failures.

The number of retries was previously set by using the `stopUpdatesAfterRepeatedFail` setting in the `Spotfire.Dxp.Worker.Web.config` file.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-u value --update-interval-seconds=value`	Optional	60	How often the server checks whether any scheduled updates should be retried. This is set in seconds. Min value is 30, and max value 3600 (one hour).
`-s <true|false> --stop-updates-after-repeated-fail-enabled=<true|false>`	Optional	true	Set to "true" to limit the number of times the server tries to update an analysis if the update initially fails. If set to "false", the server will retry the update every `update-interval-seconds` until the analysis is successfully updated.
`-f value --fails-before-stop=value`	Optional	10	Specify the number of times to retry a scheduled update before stopping. Only applies if `stop-updates-after-repeated-fail-enabled` is set to "true".
`-o <true|false> --stop-only-when-cached=<true|false>`	Optional	false	If an analysis is not cached and this option is set to "true", the server will retry the scheduled update every `update-interval-seconds` until the analysis is loaded. In this case, the `fails-before-stop` setting is ignored. If set to "false", the server will stop trying to update an analysis as specified in `fails-before-stop`, regardless of whether the analysis is cached. Only applies if `stop-updates-after-repeated-fail-enabled` is set to "true".
`-a <true|false> --always-retry-when-scheduled=<true|false>`	Optional	true	Set to "true" to reset the counter for `fails-before-stop` and retry each time the analysis is scheduled to be updated. Only applies if `stop-updates-after-repeated-fail-enabled` is set to "true".
`-d <true|false> --stop-updates-destination-unavailable=<true|false>`	Optional	true	Set to "true" to stop retrying on destinations that are offline/unavailable. Only applies if `stop-updates-after-repeated-fail-enabled` is set to "true".

config-two-factor-auth

Configures two-factor authentication.

```
config-two-factor-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-e <true|false> | --enabled=<true|false>]
```

Overview

Use this command to configure two-factor authentication. If no argument is provided, the command simply lists the current configuration and exits.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-e <true|false> --enabled=<true|false>`	Optional	none	Specifies whether or not two-factor authentication should be enabled.

config-userdir

Configures the user directory.

```
config-userdir
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-m value | --mode=value]
[-C <true|false> | --collapse-domains=<true|false>]
[-S <true|false> | --safe-synchronization=<true|false>]
[-s value | --domain-name-style=value]
[-u <true|false> | --unsafe-domain-name-style-allowed=<true|false>]
[-n value | --site-name=value]
```

Overview

Use this command to configure the user directory.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-m value --mode=value`	Optional	database	The name of the user directory mode to use. Supported values are `database`, `ldap`, and `Windows`. The current value will not be changed unless the argument is explicitly specified.
`-C value --collapse-domains=value`	Optional	false	Indicates whether or not external domains should be collapsed into the internal SPOTFIRE domain, which is the domain used when running the user directory in database mode. The current value will not be changed unless the argument is explicitly specified. ⚠ When this feature is enabled, all users will belong to the same domain. If there are multiple users with the same account name from different external domains, they will now share a single Spotfire account. Because this could pose a security problem, this feature should be used with care.
`-S <true|false> --safe-synchronization=<true|false>`	Optional	false	When this option is set to "true", the user directory will not disable users that it cannot find during LDAP or Windows NT synchronization. This flag has no effect if the user directory is running in Database mode. The current value will not be changed unless the argument is explicitly specified.
`-s value --domain-name-style=value`	Optional	dns	The domain name style used by the server. Supported values are `dns` and `netbios`. The current value will not be changed unless the argument is explicitly specified.
`-u <true|false> --unsafe-domain-name-style-allowed=<true|false>`	Optional	false	When this option is set to "true", the server will allow incompatible domain name style settings, instead of refusing to start. This option should be used with care; it can potentially lead to many users and groups being imported to the user directory with invalid domain names.
`-n value --site-name=value`	Optional	none	The name of the site for which the configuration should be applied. This

Option	Optional or Required	Default Value	Description
			flag will only have effect when used in conjunction with the `--mode` flag.

config-web-service-api

Configures the public Web Service API.

```
config-web-service-api
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-s <true|false> | --soap-enabled=<true|false>]
[-r <true|false> | --rest-enabled=<true|false>]
[-d <true|false> | --online-doc-enabled=<true|false>]
[-w <true|false> | --wsdls-require-auth=<true|false>]
```

Overview

Use this command to configure the public Web Service API (both SOAP and REST services). When none of the arguments (apart from `--configuration` and `--bootstrap-config`) are provided, the command displays the current configuration. To use the API, an OAuth 2.0 client must be registered using the register-api-client command.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-s <true|false> --soap-enabled=<true|false>]`	Optional	none	Specifies whether the public SOAP Web Service API should be enabled.
`-r <true|false> --rest-enabled=<true|false>]`	Optional	none	Specifies whether the public REST Web Service API should be enabled.
`-d <true|false> --online-doc-enabled=<true|false>`	Optional	none	Specifies whether the online API documentation should be enabled.
`-w <true|false> --wsdls-require-auth=<true|false>`	Optional	none	Specifies whether the WSDL files for the SOAP API should be available without authentication.

config-windows-userdir

Configures the Windows user directory mode.

```
config-windows-userdir
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-d value | --domains=value]
[-t value | --sleep-time=value]
[--schedules=value]
```

Overview

Use this command to configure the Windows user directory mode. If no arguments are specified, the command displays the current configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-d value --domains=value`	Optional	none	A comma-separated list of domain names. When specifying more than one domain name, make sure to enclose the list of names in quotes.
`-t value --sleep-time=value`	Optional	60 minutes	The number of minutes between each synchronization. The `--sleep-time` and `--schedules` arguments are mutually exclusive. If neither the --sleep-time argument nor the --schedules argument is specified, the synchronization is performed with a sleep time of 60 minutes.
`--schedules=value`	Optional	none	A comma-separated list of schedules for when the synchronization should be performed. The `--sleep-time` and `--schedules` arguments are mutually exclusive. The schedules are given in a cron-compatible format, where each schedule consists of either five fields or one shorthand label. Make sure to enclose the value in double quotes. The five fields are, from left to right, with their valid ranges: minute (0-59), hour (0-23), day of month (1-31), month (1-12) and day of week (0-7, where both 0 and 7 indicate Sunday). You can configure a field with the wildcard character *, indicating that any moment in time matches this field. An LDAP synchronization is triggered when all fields match the current time. If both day of month and day of week have non-wildcard values, then only one of them has to match. You can use the following shorthand labels instead of the full cron expressions: @yearly or @annually: run once a year (equivalent to 0 0 1 1 *) @monthly: run once a month (equivalent to 0 0 1 * *) @weekly: run once a week (equivalent to 0 0 * * 0) @daily or @midnight: run once a day (equivalent to 0 0 * * *) @hourly: run once an hour (equivalent to 0 * * * *) @minutely: run once a minute (equivalent to * * * * *) @reboot or @restart: run every time Spotfire Server is started

Option	Optional or Required	Default Value	Description
			Consult the Wikipedia article for an overview of the cron scheduler: http://en.wikipedia.org/wiki/Cron.

copy-group-membership

Copies group membership from one principal to another.

```
copy-group-membership
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-u value | --oldusername=value]
[-g value | --oldgroupname=value]
[-n value | --newusername=value]
[-p value | --newgroupname=value]
```

Overview

Use this command to copy the group memberships assigned to an existing user or group to another existing user or group. Only one existing principal to copy from should be given and only one principal to copy to should be given. The principal will only get memberships that it does not already have.

This will not be logged to the Action Log.

Only direct membership will be copied (that is, membership explicitly set for a certain principal and memberships that the principal inherited).

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-u value --oldusername=value`	Optional	none	The name of an existing user to copy group membership from. Unless the user is part of the configured default domain, the name of the user must include the user's domain name, for example 'DOMAIN\\user' or 'user@domain'.
`-g value --oldgroupname=value`	Optional	none	The name of an existing group to copy group membership from. Unless the group is part of the configured default domain, the name of the group must include the group's domain name, for example 'DOMAIN\\group' or 'group@domain'.
`-n value --newusername=value`	Optional	none	The name of an existing user to copy group membership to. Unless the user is part of the configured default domain, the name of the user needs to include the user's domain name, for example 'DOMAIN\\user' or 'user@domain'.
`-p value --newgroupname=value`	Optional	none	The name of an existing group to copy group membership to. Unless the group is part of the configured default domain, the name of the group needs to include the group's domain name, for example 'DOMAIN\\group' or 'group@domain'.

copy-library-permissions

Copy library permissions from one principal to another.

```
copy-library-permissions
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-u value | --oldusername=value]
[-g value | --oldgroupname=value]
[-n value | --newusername=value]
[-p value | --newgroupname=value]
```

Overview

Use this command to copy library permissions from an existing user or group to another existing user or group. Only one existing principal to copy from should be given and only one principal to copy to should be given. The principal will only get permissions that it does not already have.

This will not be logged to the Action Log.

A permission entry, for example "Browse + Access", counts as two permission entries when summing up how many new permissions have been added.

Only explicit permissions will be copied (permissions explicitly set for a certain principal, and not permissions given through group membership).

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`u value --oldusername=value`	Optional	none	The name of an existing user to copy library permissions from. Unless the user is part of the configured default domain, the name of the user must include the user's domain name ('DOMAIN\user' or 'user@domain').
`g value --oldgroupname=value`	Optional	none	The name of an existing group to copy library permissions from. Unless the group is part of the configured default domain, the name of the group must include the group's domain name ('DOMAIN\group' or 'group@domain').
`n value --newusername=value`	Optional	none	The name of an existing user to copy library permissions to. Unless the user is part of the configured default domain, the name of the user must include the user's domain name ('DOMAIN\user' or 'user@domain').
`p value --newgroupname=value`	Optional	none	The name of an existing group to copy library permissions to. Unless the group is part of the configured default domain, the name of the group must include the group's domain name ('DOMAIN\group' or 'group@domain').

copy-rules-to-site

Copies routing rules and schedules from one site to another

```
copy-rules-to-site
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-F value | --source-site-name=value>
<-T value | --target-site-name=value>
[-r value | --resource-pool-name=value]
[-u <true|false> | --use-default-resource-pool=<true|false>]
[-d <true|false> | --disabled=<true|false>]
[-R value | --rule-conflict-resolution=value]
[-S value | --schedule-conflict-resolution=value]
[-e <true|false> | --test-run=<true|false>]
```

Overview

Use this command to copy all the routing rules and schedules from the source site to the target site.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-F value --source-site-name=value`	Required	none	The name of the site from which the routing rules and schedules will be copied.
`-T value --target-site-name=value`	Required	none	The name of the site into which the routing rules and schedules will be copied.
`-r value --resource-pool-name=value`	Optional	none	A resource pool name that can be used if the resource pool for a given rule is not found.
`-u <true|false> --use-default-resource-pool=<true|false>`	Optional	false	If enabled and the resource pool for a given rule is not found, the default resource pool will be used instead, and the instances count will be automatically reset to one instance.
`-d <true|false> --disabled=<true|false>`	Optional	false	If true, all the rules will be copied in a disabled state.
`-R value rule-conflict-resolution=value`	Optional	fail	Defines how to handle copying a rule if there already exists a rule with the same name and the same file/user/group in the target site. The argument can be one of: fail (default), replace, or skip.
`-S value --schedule-conflict-resolution=value`	Optional	rename	Defines how to handle copying a shared schedule if there already exists a shared schedule with the same name in the target site. The argument can be one of: rename (default), or replace. If the schedules are identical, the schedule in the target site will remain as it was.
`-e <true|false> --test-run=<true|false>`	Optional	false	If true, the copy will not actually take place, but the command will produce a preview of the import status of each rule/schedule.

create-default-config

Creates a new server configuration file containing the default configuration.

```
create-default-config
[-f | --force]
[export file]
```

Overview

Use this command to export a default server configuration to a file. The configuration in the file can be edited and then imported into the server database using the import-config command.

Options

Option	Optional or Required	Default Value	Description
`-f` `--force`	Optional	none	Indicates that the tool should overwrite an existing destination file.
`[export file]`	Optional	configuration.xml	The path to the configuration file that will be created.

create-jmx-user

Creates a new JMX user account.

```
create-jmx-user
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
[-p value | --password=value]
[-l value | --access-level=value]
```

Overview

Use this command to create a new JMX user account. The account can be used only to access status information for the server through the JMX protocol. It cannot be used by users logging in to the server using a Spotfire client or an HTML browser.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-u value --username=value`	Required	none	The name of the JMX user to create.
`-p value --password=value`	Optional	none	The new JMX user password.
`-l value --access-level=value`	Optional	r	The access level for the new user. Can be either r or rw. A user with the rw access level can read and modify any writable attributes.

create-join-db

Configures the default join database.

```
create-join-db
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-t value | --type=value>
<-d value | --database-url=value>
<-u value | --username=value>
[-p value | --password=value]
[-i value | --min-connections=value]
[-a value | --max-connections=value]
[-v | --validate]
```

Overview

Use this command to configure the default join database.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --type=value`	Required	none	The database type and the driver to use. Must match the type name of one of the enabled data source templates.
`-d value --database-url=value`	Required	none	The JDBC URL to the database. Because this argument usually contains special characters, be sure to escape those characters or enclose the values in quotes.
`-u value --username=value`	Required	none	The database account username.
`-p value --password=value`	Optional	none	The database account password.
`-i value --min-connections=value`	Optional	0	The minimum number of connections to keep in the connection pool.
`-a value --max-connections=value`	Optional	0	The maximum number of connections to keep in the connection pool.
`-v --validate`	Optional	none	Indicates whether the created configuration should be validated by attempting to connect to the database using the specified connection information.

create-ldap-config

Creates a new LDAP configuration for authentication and/or the user directory LDAP provider.

```
create-ldap-config
[-c value | --configuration=value]
[-b value |--bootstrap-config=value]
<--id=value>
[--discover]
[-t value |--type=value]
[-s value | --servers=value]
[-n value |--context-names=value]
[-u value | --username=value]
[-p value |--password=value]
[--schedules=value]
[--user-search-filter=value]
[--user-name-attribute=value]
[--authentication-attribute=value]
```

```
[--security-authentication=value]
[--referral-mode=value]
[--referral-mode-root-dse=value]
[--request-control=value]
[--page-size=value]
[--import-limit=value]
[--user-display-name-attribute=value]
[--group-display-name-attribute=value]
{-Ckey=value}
{-Rvalue}
{-Svalue}
[--connection-timeout=value]
[--read-timeout=value]
```

Overview

Use this command to create a new LDAP configuration for authentication and/or user directory back-end.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--id=value`	Required	none	Specifies the identifier for the LDAP configuration to be created.
`--discover`	Optional	none	Specifies whether to attempt to automatically create an LDAP configuration based on the information available from the DNS service. The discover mode works only when the desired LDAP server has registered SRV records in the DNS service used by the computer where this command is being invoked. This is typically the case for Active Directory LDAP servers. This argument is mutually exclusive with the `-t/ --type`, `-s/--servers`, and `-n/--context-names` arguments.
`-t value --type=value`	Required, unless the `--discover` option is used	none	The type of LDAP server. The following names are valid types: • ActiveDirectory • SunOne • SunJavaSystem • Custom If you specify any of the first three types, a type-specific configuration template is automatically applied in runtime, so that the most fundamental configuration options are automatically configured. If you specify a "Custom" LDAP server type, there is no such configuration template, and you must specify explicitly all the configuration options. When you use a custom LDAP configuration for authentication or with the User Directory LDAP provider, you must specify the arguments `--user-search-filter` and `--user-name-attribute`. If you use such an LDAP configuration for group synchronization, you must also specify additional parameters when running the config-ldap-group-sync command. See the help topic for that command for more information.
`-s value --servers=value`	Required, unless the `--discover` option is used	The LDAP protocol port number defaults to 389.	A whitespace-separated list of LDAP server URLs. An LDAP server URL has the format <protocol>://<server>[:<port>]:

Option	Optional or Required	Default Value	Description
		The LDAPS protocol port number defaults to 636. Active Directory LDAP servers also provide a Global Catalog containing forest-wide information, instead of domain-wide information only. By default, the Global Catalog LDAP service listens on port number 3268 (LDAP) or 3269 (LDAPS).	• `<protocol>`: Either "LDAP" or "LDAPS". • `<server>`: The fully qualified DNS name of the LDAP server. • `<port>`: Optional. Indicates the port number that the LDAP service is listening on. Spotfire Server does not expect search base, scope, filter, or other additional parameters after the port number in the LDAP server URLs. Such properties are specified using other configuration options for this command. Examples: LDAP server URLs • `LDAP://myserver.example.com` • `LDAPS://myserver.example.com` • `LDAP://myserver.example.com:389` • `LDAPS://myserver.example.com:636` • `LDAP://myserver.example.com:3268` • `LDAPS://myserver.example.com:3269`
`-n value` `--context-names=value`	Required, unless the `--discover` option is used	none	A list of distinguished names (DNs) of the containers holding the LDAP accounts to be visible within the Spotfire Server. When you specify more than one DN, you must separate the DNs using pipe characters (`|`). If the specified containers contain a large number of users, of which only a few should be visible in Spotfire Server, you can specify a custom user search filter to include only the designated users (see the `--user-search-filter` argument). Examples: • `CN=users,DC=example,DC=com` • `OU=project-x,DC=research,DC=example,DC=com`
`-u value` `--username=value`	Required	none	The name of the LDAP service account to use when searching for users (and optionally also groups) in the LDAP server. This service account does not need to have write permissions, but it must have read permissions for all configured context names (LDAP containers). For most LDAP servers, the account name is the account's distinguished name (DN). For Active Directory, the account name can also be specified in the forms `ntdomain\name` and `name@dnsdomain`. Examples: • `CN=spotsvc,OU=services,DC=research,DC=exam`

Option	Optional or Required	Default Value	Description
			• `RESEARCH\spotsvc` (Note: Active Directory only) • `spotsvc@research.example.com` (Note: Active Directory only)
`-p value` `--password=value`	Optional	none	The password for the LDAP service account.
`--schedules=value`	Optional	@daily, @restart	A comma-separated list of schedules for when the LDAP synchronization should be performed. The schedules are given in a cron-compatible format, where each schedule consists of either five fields or one shorthand label. Make sure you enclose the value in double quotes. The five fields are, from left to right, with their valid ranges: minute (0-59), hour (0-23), day of month (1-31), month (1-12) and day of week (0-7, where both 0 and 7 indicate Sunday). You can also configure a field with the wildcard character *, indicating that any moment in time matches this field. An LDAP synchronization is triggered when all fields match the current time. If both day of month and day of week have non-wildcard values, then only one of them has to match. You can also use following shorthand labels instead of the full cron expressions: • `@yearly` or `@annually`: run once a year (equivalent to 0 0 1 1 *) • `@monthly`: run once a month (equivalent to 0 0 1 * *) • `@weekly`: run once a week (equivalent to 0 0 * * 0) • `@daily` or `@midnight`: run once a day (equivalent to 0 0 * * *) • `@hourly`: run once an hour (equivalent to 0 * * * *) • `@minutely`: run once a minute (equivalent to * * * * *) • `@reboot` or `@restart`: run every time the Spotfire Server is started Refer to the Wikipedia overview article on the cron scheduler.
`--user-search-filter=value`	Optional, but it must be specified for custom LDAP configurations, either when running this command or the	For Active Directory servers, the parameter value defaults to '(&(objectClass=user)(!(objectClass=computer)))'	Specifies an LDAP search expression filter to use when searching for users. If you need to identify a subset of users in the specified LDAP containers who should be allowed access to Spotfire Server, you can specify a more detailed user search filter. For example, the search expression can be expanded so that it also

Option	Optional or Required	Default Value	Description
	update-ldap-config command.	For any version of the Sun Directory Servers, it defaults to 'objectClass=person'.	puts restrictions on which groups the users belong to, or which roles they have. • For Active Directory servers, access can be restricted to only those users belonging to a certain group by using a search expression with the pattern `&(objectClass=user)(memberOf=<groupDN>)` where `<groupDN>` is replaced by the real DN of the group to which the users must belong. If the users are divided among multiple groups, use the pattern `&(objectClass=user)(|(memberOf=<firstDN>)(memberOf=<secondDN>))`. Add extra `(memberOf=<groupDN>)` sub-expressions as needed. Active Directory example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)` • For a Sun Java System Directory Server version 6 and later, you can achieve the same effect by using a search expression with the pattern `&(objectClass=person)(isMemberOf=<groupDN>)`. If the users are divided among multiple groups, use the pattern `&(objectClass=person)(|(isMemberOf=<firstDN> )(isMemberOf=<secondDN>))`. Add extra `(isMemberOf=<groupDN>)` sub-expressions as needed. Sun Java System Directory Server example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)` • For Sun ONE Directory Servers and newer Sun Java System Directory Servers or the older iPlanet Directory Server, you can restrict access to only those users having certain specific roles. The search expression for role filtering must match the pattern `&(objectClass=person)(nsRole=<roleDN>)`. If multiple roles are of interest, use the pattern `&(objectClass=person)(|(nsRole=<firstDN>)(nsRole=<secondDN>)`. Add extra `(nsRole=<roletDN>)` sub-expressions as needed. Sun ONE Directory Servers example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)`

Option	Optional or Required	Default Value	Description
			The syntax of LDAP search expression filters is specified by the RFC 4515 document. Consult this documentation for information about more advanced filters.
`--user-name-attribute=value`	Optional, unless the LDAP server type is set to "Custom" using the `--type` parameter.	For Active Director servers, the value defaults to sAMAccountName. For a Sun Java System Directory Server or any older Sun ONE Directory Server or iPlanet Directory Server with a default configuration, it defaults to 'uid'.	Specifies the name of the LDAP attribute containing the user account names.
`--authentication-attribute=value`	Optional; use only for advanced setups. It is not set by default.	none	Specifies the name of the LDAP attribute containing a user identity that can be used for binding (authenticating) to the LDAP server. This attribute fills no purpose in most common LDAP configurations, but it can be useful in more advanced setups where the distinguished name (DN) does not work for authentication, or where users should be able to log in using a username that does not map directly to an actual LDAP account. • If you set up SASL with DIGEST-MD5 in an Active Directory environment, the DN does not work for authentication, and the `userPrincipalName` attribute must be used instead. The `--authentication-attribute` argument should then be set to "userPrincipalName" and the `--user-name-attribute` argument should be set to "sAMAccountName". (The latter value is the default value for an Active Directory LDAP configuration, so there is no need to set it explicitly.) See also the `--security-authentication` argument. • When you set up SASL with GSSAPI in an Active Directory environment, the DN does not work for authentication and the sAMAccountName or userPrincipalName attribute must be used instead. The `--authentication-attribute` argument should be set to "sAMAccountName" or "userPrincipalName", and the `--user-name-attribute` argument should be set to "sAMAccountName". (The latter value is the default value for an Active Directory LDAP configuration, so there is no need to set it explicitly.) See also the `--security-authentication` argument. Example:

Option	Optional or Required	Default Value	Description
			If you set the `--user-name-attribute` argument to "cn" and the `--authentication-attribute` argument to "userPrincipalName" in an Active Directory environment, the users can log in to Spotfire Server using their CN attribute values, but underneath the hood, Spotfire Server actually uses the `userPrincipalName` attribute value of the LDAP account with the matching CN for the actual authentication.
`--security-authentication=value`	Optional; use only in advanced setups.	`simple`	Specifies the security level to use when binding to the LDAP server: • To enable anonymous binding, it should be set to "none". • To enable plain username/password authentication, it should be set to "simple". • To enable SASL authentication, it should be set to the name of the SASL mechanism to be used, for instance "DIGEST-MD5" or "GSSAPI". Use multiple `-C` arguments to set the additional JNDI environment properties that the SASL authentication mechanism typically requires. If you set up SASL with DIGEST-MD5 in an Active Directory environment, all accounts must use reversible encryption for their passwords. This is typically not the default setting for the domain controller. The `--authentication-attribute` argument must also be used to specify the `userPrincipalName` attribute for the actual authentication to work correctly. If you set up SASL with GSSAPI in an Active Directory environment, the `--authentication-attribute` argument must be used to specify either the `sAMAccountName` or the `userPrincipalName` attribute, and the custom property `kerberos.login.context.name` must be mapped to the JAAS application configuration `SpotfireGSSAPI`. This, in turn, requires a fully working Kerberos configuration file at `<server installation dir>/tomcat/spotfire-config/krb5.conf`.
`--referral-mode=value`	Optional	follow	Specifies how LDAP referrals should be handled. Valid arguments: • follow (automatically follow any referrals). Recommended. • ignore (ignore referrals)

Option	Optional or Required	Default Value	Description
			• throw (fail with an error)
`[--referral-mode-root-dse=value]`	Optional	If not explicitly set, the value for `--referral-mode` is used.	Specifies how LDAP referrals should be handled when looking up the RootDSE. Valid arguments are: • follow (automatically follow any referrals) • ignore (ignore referrals) • throw (fail with an error)
`--request-control=value`	Optional	probe	Determines the type of LDAP controls to be used for executing search queries to the LDAP server. The default behavior is to probe the LDAP server for the best supported request control. The paged results control is always preferred, because it provides the most efficient way of retrieving the query result set. You can use the virtual list view control for the same purpose if the paged results control is not supported. The virtual list view control is used automatically, together with a sort control. Both the paged results control and the virtual list view control support a configurable page size, set by the `--page-size` argument. • To explicitly configure the server for probing, set the argument value to "probe". • To configure the server for the paged results control, set the argument value to "PagedResultsControl". • To request the virtual list view control, set the argument value to "VirtualListViewControl". • To completely disable request controls, set the argument value to "none".
`--page-size=value`	Optional	2000 for both the paged results control and the virtual list view control.	Specifies the page size to be used with the paged results control or the virtual list view control when performing search queries to the LDAP server.
`--import-limit=value`	Optional	No import limit	Specifies a threshold that limits the number of users that can be imported from an LDAP server to Spotfire Server in one query. This can be used to prevent accidentally flooding the Spotfire user directory when you integrate with an LDAP server with tens or even hundreds of thousands of users. By setting an import limit, you can be sure that an unexpected high number of users will not affect server performance. To request unlimited import explicitly, set the parameter value to "-1". All positive

Option	Optional or Required	Default Value	Description
			numbers are treated as an import limit. For most cases it is recommended that you leave this parameter untouched.
`--user-display-name-attribute=value`	Optional	none	Specifies the name of the LDAP attribute containing the user display names.
`--group-display-name-attribute=value`	Optional	none	Specifies the name of the LDAP attribute containing the group display names.
`-Ckey=value`	Optional; can be specified multiple times with different keys.	none	Specifies additional JNDI environment properties to use when connecting to the LDAP server. Example: The equivalent of specifying the `--security-authentication=DIGEST-MD5` argument is `-Cjava.naming.security.authentication=DIGEST-MD5`.
`-Rvalue`	Optional; can be specified multiple times with different values.	If this argument is not specified, the Java defaults are used.	Specifies the protocols to be used for LDAPS when connecting to the LDAP server. Example: To enable only TLSv1.2 `> -RTLSv1.2`
`-Svalue`	Optional; can be specified multiple times with different values.	If this argument is not specified, the Java defaults are used.	Specifies the cipher suites to be used for LDAPS when connecting to the LDAP server. Example: To enable only these two cipher suites `> -STLS_DHE_RSA_WITH_AES_128_GCM_SHA256 -STLS_DHE_RSA_WITH_AES_256_GCM_SHA384`
`--connection-timeout=value`	Optional	No timeout (see description)	Specifies the connection timeout. The value must be a non-negative integer representing the timeout in milliseconds. A value less than or equal to zero results in no timeout, effectively waiting until the connection times out on the TCP network level.
`--read-timeout=value`	Optional	No timeout (see description)	Specifies the read timeout. The value must be a non-negative integer representing the timeout in milliseconds. A value less than or equal to zero results in no timeout, effectively waiting until the connection times out on TCP network level.

EXAMPLES

Create an LDAP configuration for Active Directory:

```
create-ldap-config --id="ldap1" --type="ActiveDirectory"
```

```
  --servers="ldap://dc01.research.example.com:3268 ldap://dc02.research.example.com:3268"
 --context-names="OU=project-x,DC=research,DC=example,DC=com|
OU=phbs,DC=management,DC=example,DC=com"
  --username="ldapadmin@research.example.com" --password="s3cr3t"
  --schedules="@daily"
```

Create an LDAP configuration for SunONE:

```
create-ldap-config --id="ldap1" --type="SunONE"
  --servers="ldap://directory.research.example.com:389" --context-names="OU=project-
x,DC=research,DC=example,DC=com|OU=phbs,DC=management,DC=example,DC=com"
  --username="ldapadmin" --password="s3cr3t"
  --schedules="@daily"
```

Create an LDAP configuration for Sun Java System Directory:

```
create-ldap-config --id="ldap1" --type="SunJavaSystem"
  --servers="ldaps://directory.research.example.com:636" --context-names="OU=project-
x,DC=research,DC=example,DC=com|OU=phbs,DC=management,DC=example,DC=com"
  --username="ldapadmin" --password="s3cr3t"
  --schedules="@daily"
```

Create an LDAP configuration for a custom LDAP server:

```
create-ldap-config --id="ldap1" --type="Custom"
  --servers="ldap://directory.research.example.com" --context-names="OU=project-
x,DC=research,DC=example,DC=com|OU=phbs,DC=management,DC=example,DC=com"
  --user-name-attribute="cn" --search-filter="&(objectClass=person)
(isMemberOf=cn=projectX,dc=example,dc=com)"
  --username="ldapadmin" --password="s3cr3t"
  --schedules="@daily"
```

Create an LDAP configuration using the discover mode:

```
create-ldap-config --id="ldap1" --discover
  --username="ldapadmin@research.example.com" --password="s3cr3t"
  --schedules="@daily"
```

create-scheduled-jobs

Creates scheduled Automation Services jobs from a JSON file created by the user.

```
create-scheduled-jobs
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-p value | --local-file-path=value>
[-e <true|false> | --enabled=<true|false>]
[-s value | --site-name=value]
```

Overview

Use this command to create scheduled Automation Services jobs from a local JSON file that is created by the user. At least one Spotfire Server instance must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-p value --local-file-path=value`	Required	none	Full path to the local JSON file that the user created. This file contains the path of the Automation Services job to be scheduled and the schedule details. For the required structure of this file, see the example below this table.
`-e <true|false> --enabled=<true|false>`	Optional	false	Optional flag to specify whether the scheduled jobs are enabled when created.
`-s value --site-name=value`	Optional	none	The name of the site for which the scheduled jobs should be created. If no site is given, the scheduled jobs are created for the default site.

JSON file for scheduling Automation Services jobs

A JSON file conforming to the following structure must be created and saved locally before running the **`create-scheduled-job`** command. This example contains two schedules for the Automation Services job, but you may add as many as necessary.

```
[ { "libraryItemPath" : "/asjob/job1",
"ruleType" : "asjob",
"schedules" : [
{
"startTime" : "17:00:00",
"daysOfTheWeek" : [ "SUNDAY", "MONDAY", "TUESDAY", "WEDNESDAY",
 "THURSDAY", "FRIDAY", "SATURDAY" ],
"timeZone" : "America/Los_Angeles"
},{
"startTime" : "01:00:00",
"daysOfTheWeek" : [ "SUNDAY", "MONDAY", "TUESDAY", "WEDNESDAY",
 "THURSDAY", "FRIDAY", "SATURDAY" ],
"timeZone" : "America/Los_Angeles"
 } ]
```

Parameters for JSON file

Parameter	Description
libraryItemPath	The path to the Automation Services job to be scheduled.
ruleType	"asjob", for Automation Services job, is currently the only option.
startTime	The time, in the format HH:MM:SS, that the job should run.
daysOfTheWeek	The days of the week that the job should run (at the time specified in the previous parameter).
timeZone	Time zone, in the Area/City format, for the times that the job should run. Use the time zone names that are listed in the iana (Internet Assigned Numbers Authority) Time Zone Database at https://www.iana.org/time-zones.

Command example

```
config create-scheduled-jobs --local-file-path=C:\tibco
\Scheduled_AS_jobs\WeeklySalesReport.json --enabled=true --site-
name=West Coast
```

create-site

Creates a new site.

```
create-site
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-s value | --site-name=value>
[-d value | --display-name=value]
```

```
[-a value | --public-address=value]
[-i <true|false> | --ignore-existing=<true|false>]
```

Overview

Use this command to create a new site to which servers may be assigned.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-s value --site-name=value`	Required	none	The name of the site that will be created. Used for identifying the site, e.g. when specifying site-specific configuration.
`-d value --display-name=value`	Optional	none	The display name of the site that should be created. May help users quickly identify which server to connect to (in an environment with multiple Spotfire systems).
`-a value --public-address=value`	Optional	none	The public address of the site, for example 'http[s]://host[:port]/'. If no public address is set, it will be automatically determined during Spotfire Server startup. To change the value later on, use the set-public-address command. **Notes:** • It is recommended to specify the public address when creating a site. • If the public address is later changed, first restart the servers and then restart all the service instances to propagate the change.
`-i <true|false> --ignore-existing=<true|false>`	Optional	false	If set, an error creating a duplicate site will be ignored.

create-user

Creates a new user account.

```
create-user
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
[-p value | --password=value]
```

```
[-d value | --display-name=value]
[-e value | --email=value]
```

Overview

Use this command to create a new user account. This user can then be promoted to administrator using the promote-admin command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`u value --username=value`	Required	none	The name of the new user.
`-p value --password=value`	Optional	none	The new user's password.
`-d value --display-name=value`	Optional	none	The new user's display name.
`-e value --email=value`	Optional	none	The new user's email address.

delete-disabled-users

Deletes disabled user accounts.

```
delete-disabled-users
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-a <true|false> | --keep-once-active-users=<true|false>]
[-m <true|false> | --keep-group-members=<true|false>]
[-p <true|false> | --keep-users-with-library-permissions=<true|false>]
[-l <true|false> | --keep-library-authors=<true|false>]
[-f | --force]
```

Overview

Use this command to delete disabled user accounts from the user directory.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-a <true|false> --keep-once-active-users=<true|false>`	Optional	true	Indicates whether all users who have logged in at least once should be kept.
`-m <true|false> --keep-group-members=<true|false>`	Optional	true	Indicates whether all users who are members of at least one group should be kept.
`-p <true|false> --keep-users-with-library-permissions=<true|false>`	Optional	true	Indicates whether all users who have explicit library permissions should be kept.
`-l <true|false> --keep-library-authors=<true|false>`	Optional	true	Indicates whether all users who have created or modified any library item should be kept.
`-f --force`	Optional	none	Indicates that users should be deleted without need for further confirmation.

delete-disconnected-groups

Deletes disconnected groups.

```
delete-disconnected-groups
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-f | --force]
```

Overview

Use this command to delete from the user directory disconnected groups that have been previously synchronized from an LDAP directory.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-f --force`	Optional	none	Indicates that groups should be deleted without need for further confirmation.

delete-jmx-user

Deletes a JMX user.

```
delete-jmx-user
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
```

Overview

Use this command to delete a user who can access the server through JMX.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-u value --username=value`	Required	none	The name of the user to be deleted.

delete-library-content

Deletes library content.

```
delete-library-content
[-b value | --bootstrap-config=value]
```

```
[-t value | --tool-password=value]
<-i value | --items=value>
[-d | --database]
[-e | --external]
```

Overview

Use this command to delete a library items from the Spotfire database or from external storage on Amazon S3.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-i value --items=value`	Required	none	A comma-separated list of items (GUIDs) to delete.
`-d --database`	Optional	none	Deletes entries in the Spotfire library database.
`-e --external`	Optional	none	Deletes entries in external storage.

delete-node

Deletes a specified node.

```
delete-node
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --id=value>
```

Overview

Use this command to delete a specified node, after which it will no longer be a part of the collective. To use this command, at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --id=value`	Required	none	The ID of the node that should be deleted. The list-nodes command can be used to find the IDs of all nodes.

delete-oauth2-client

Deletes a specified OAuth2 client.

```
delete-oauth2-client
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
 <-i value | --client-id=value>
```

Overview

Use this command to delete a specified OAuth2 client. To use this command at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-k value` `--keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value` `--client-id=value`	Required	none	The ID of the client to be deleted. The list-oauth2-clients command can be used to find the IDs of all clients.

delete-service-config

Deletes a service configuration.

```
delete-service-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-c value | --config-name=value>
```

Overview

Use this command to delete a service configuration. If the configuration is currently assigned to a service, that service will be reverted to the default configuration.

Options

Option	Optional or Required	Default Value	Description
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-c value` `--config-name=value`	Required	none	The name of the configuration that should be deleted.

delete-site

Deletes a site.

```
delete-site
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-s value | --site-name=value>
[-i value | --target-site=value]
[-f | --force]
```

Overview

Use this command to delete a site. To delete a site that currently contains nodes, the `--target-site` argument must be specified. All nodes in the site will then be moved to the specified site.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-s value --site-name=value`	Required	none	The name of the site that will be deleted.
`-i value --target-site=value`	Optional unless the site being deleted contains nodes. If the argument is not present and there are rules, scheduled updates, or resource pools in the deleted site, these will also be removed.	none	The name of a site into which any nodes, routing rules, scheduled updates, or resource pools in the site being deleted should be moved.
`-f --force`	Optional	none	Indicates whether the site's routing rules, scheduled updates, and resource pools should be deleted along with the site.

delete-user

Deletes a user account.

```
delete-user
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
```

Overview

Use this command to delete a user account.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-u value --username=value`	Required	none	The name of the user to be deleted.

demote-admin

Revokes full administrator privileges for a user.

```
demote-admin
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
```

Overview

Use this command to revoke administrator privileges for a user by removing the user account from the Administrator group.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-u value --username=value`	Required	none	The name of the user for which to revoke the administrator privileges. Unless the user is part of the configured default domain, the name of the user needs to include the user's domain name, for example DOMAIN\user or user@domain.

download-troubleshooting-bundle

Downloads the troubleshooting bundle.

```
download-troubleshooting-bundle
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
[-d value | --days-to-include=value]
[-o value | --output-file-name=value]
[-f | --force]
[-e <true|false> | --external-include=<true|false>]
{-ivalue | --server-idvalue}
```

Overview

Use this command to download the troubleshooting bundle.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-d value --days-to-include=value`	Optional	none	Number of days for which the logs are to be included in the troubleshooting bundle.
`-o value --output-file-name=value`	Optional	none	The location/filename of the output file along with the extension (.zip).
`-f --force`	Optional	false	Option to overwrite the output file if it already exists.
`e <true|false> --external-include=<true|false>`	Optional	true	Option to include external content, i.e. content related to remote nodes. This option is valid only if a `server-idvalue` is not specified.
`-ivalue --server-idvalue`	Optional, and may be specified multiple times with different values.	none	The ID of the server to obtain data from. Use the list-nodes command to see the list of servers/nodes.

enable-user

Enables or disables a user account in the Spotfire database.

```
enable-user
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-u value | --username=value]
[-a | --all]
[-e <true|false> | --enabled=<true|false>]
```

Overview

Use this command to enable or disable a user account in the Spotfire database. A disabled user account does not have access to the Spotfire Server.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See the Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`u value --username=value`	Optional, and mutually exclusive with the **`--all`** flag.	none	The user account that should be enabled or disabled. The specified user must not be a built-in system account.
`-a --all`	Optional, and mutually exclusive with the **`--username`** argument.	none	Updates the enabled status for all the users, except for built-in system accounts. The guest account can only be disabled using this flag. If it needs to be enabled, it must be specified explicitly using the **`--username`** argument.
`-e <true|false> --enabled=<true|false>`	Optional	true	Specifies whether the user account should be enabled or disabled.

export-config

Exports a server configuration from the server database to the current working directory as a `configuration.xml` file.

```
export-config
[-f | --force]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-h value | --hash=value]
[export file]
```

Overview

Use this command to export a server configuration from the server database to a file. The configuration in the file can be edited and then imported back into the server database using the import-config command.

Options

Option	Optional or Required	Default Value	Description
`-f` `--force`	Optional	none	Indicates that the tool should overwrite an existing destination file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-h value` `--hash=value`	Optional	none	The (possibly abbreviated) hash of the configuration to export. Must consist of at least 6 hexadecimal characters.
`[export file]`	Optional	configuration.xml	The path to the configuration file that will be created.

export-ds-template

Exports the definition of a data source template.

```
export-ds-template
[-f | --force]
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
[template definition file]
```

Overview

Use this command to export to a file the definition of a data source template used by Information Services.

Options

Option	Optional or Required	Default Value	Description
`-f` `--force`	Optional	none	Indicates whether the tool should overwrite an existing destination file.
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value` `--name=value`	Required	none	The name of the data source template for which to export the definition.
`[template definition file]`	Optional	template.xml	The path to the definition file to create.

export-groups

Exports groups from the user directory.

```
export-groups
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-m <true|false> | --include-member-groups=<true|false>]
[-u <true|false> | --include-member-users=<true|false>]
[-g <true|false> | --include-guids=<true|false>]
[-s <true|false> | --use-stdf=<true|false>]
[-n <true|false> | --include-name-row=<true|false>]
[export file]
[-f | --force]
```

Overview

Use this command to export all groups from the user directory. The exported groups can be imported on a different server.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-m <true|false> --include-member-groups=<true|false>`	Optional	false	Indicates whether the group hierarchy information (groups in groups) should be included. Can be used in conjunction with the `--include-member-users` argument to include all information.
`-u <true|false> --include-member-users=<true|false>`	Optional	false	Indicates whether the group hierarchy information (users in groups) should be included. Can be used in conjunction with the `--include-member-groups` argument to include all information.
`-g <true|false> --include-guids=<true|false>`	Optional	false	Indicates whether the globally unique identifier (GUID) of each group should be included.
`-s <true|false> --use-stdf=<true|false>`	Optional	true	Indicates whether the exported file should be created in Spotfire Text Data Format. If "false", plain CSV format is used.
`-n <true|false> --include-name-row=<true|false>`	Optional	false	Indicates whether the exported file should include a column name row. Applicable only when `--use-stdf` is set to "false" because STDF always includes a name row.
`[export file]`	Optional	groups.txt	The path to the file to create.
`-f --force`	Optional	none	Indicates that the tool should overwrite an existing destination file.

export-library-content

Exports content from the library.

```
export-library-content
[-f | --force]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-p value | --file-path=value>
<-u value | --user=value>
[-a <true|false> | --include-access-rights=<true|false>]
<-i value | --item-type=value>
<-l value | --library-path=value>
[--treat-as-single-item=<true|false>]
```

Overview

Use this command to export content from the library.

Options

Option	Optional or Required	Default Value	Description
`-f` `--force`	Optional	none	Indicates that the tool should overwrite any already existing file with the same name as specified in the path argument. All parts of the existing file (path.part0.zip, path.part1.zip, and so on) are also deleted.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-p value` `--file-path=value`	Required	none	The file system path to where the item should be exported.
`-u value` `--user=value`	Required	none	The user performing the export should be a Library Administrator. The name of the user needs to include the user's domain name, for example DOMAIN \user or user@domain, unless the user is part of the configured default domain.
`-a <true|false>` `--include-access-rights=<true|false>`	Optional	true	Specifies if access rights should be exported.
`-i value` `--item-type=value`	Required	none	Indicates which item types should be exported from the library. It is possible to export all items, or all items of a certain type, from a folder. It is also possible to export a single item of a certain type. When exporting the content of a folder, valid values are: `all_items`, `data_files`, `analysis_files`, `data_access`, `datafunctions`, `colorschemes`, `automation_job`, and `information_model`. When exporting a single item, valid values are, for example: `dxp`, `sbdf`, `connectiondatasource`, `query`, `asjob`, `column`, `procedure`, `analyticmodel`, `dxpscript`, `filter`, `datafunction`, `datasource`, `colorscheme`, `dataconnection`, and `join`.
`-l value` `--library-path=value`	Required	none	The path in the library where the content is exported from. When exporting folder content, a path to the folder must be specified. When exporting a single item, a path to that specific item must be specified. The path must start with a slash (/). If the entire library should be exported, the path should be "/".
`--treat-as-single-item=<true|false>`	Optional	false	To specify if the given library path is to be treated as a single item.

export-rules

Exports routing rules and schedules, including scheduled Automation Services jobs, from the server.

```
export-rules
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
[-p value | --export-file-path=value]
[-f | --force]
```

Overview

Use this command to export all the routing rules and schedules, including scheduled Automation Services jobs, from the server. The exported rules may be imported on a different server.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-p value --export-file-path=value`	Optional	rules.json	The path to the file to create.
`-f --force`	Optional	none	The force flag indicates whether the tool overwrites an existing destination file.

export-service-config

Exports a service configuration.

```
export-service-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n value | --config-name=value]
[-a value | --capability=value]
[-d value | --deployment-area=value]
[-f | --force]
[destination directory]
```

Overview

Use this command to export a service configuration for editing. The edited configuration can be imported using the import-service-config command. Either specify a configuration name or, to export a default configuration, a capability, and a deployment area.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-n value --config-name=value`	Required, unless the `--capability` and `--deployment-area` arguments are specified (in which case this argument cannot be specified).	none	The name of the configuration that should be exported.
`-a value --capability=value`	Required, unless the `--config-name` argument is specified (in which case this argument cannot be specified).	none	The name of a capability for which the default configuration should be exported. The possible values can be found using the list-service-configs command. This argument must be specified together with the `--deployment-area` argument.
`-d value --deployment-area=value`	Required, unless the `--config-name` argument is specified (in which case this argument cannot be specified).	none	The name of a deployment area for which the default configuration should be exported. This argument must be specified together with the `--capability` argument.
`-f --force`	Optional	none	Indicates that the tool should overwrite any existing destination directory.
`[destination directory]`	Optional	config	The destination directory to which the configuration should be exported.

export-users

Exports users from the user directory.

```
export-users
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-i value | --include-password-hashes=value]
[-s value | --use-stdf=value]
```

```
[-g value | --include-guids=value]
[-n value | --include-name-row=value]
[export file]
[-f | --force]
```

Overview

Use this command to export all users from the user directory. The exported users can be imported on a different server.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-i value --include-password-hashes=value>`	Optional	false	Indicates whether the exported file should include the password hashes. Passwords are relevant only if you use the Spotfire database for authentication.
`-s value --use-stdf=value`	Optional	true	Indicates whether the exported file should be created in Spotfire Text Data Format. If `false`, plain CSV format is used.
`-g value --include-guids=value`	Optional	false	Indicates whether the Globally Unique Identifier (GUID) of each user should be included.
`-n value --include-name-row=value`	Optional	false	Indicates whether the exported file should include a column name row. Applicable only when `--use-stdf` is set to `false` because STDF always includes a name row.
`[export file]`	Optional	users.txt	The path to the file to create.
`-f --force`	Optional	none	Indicates that the tool should overwrite an existing destination file.

find-analysis-scripts

Finds scripts, data functions, and custom queries in files in the library.

```
find-analysis-scripts
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-p value | --library-parent-path=value]
[-e value | --library-search-expression=value]
```

```
[-s <true|false> | --auto-trust-scripts=<true|false>]
[-d <true|false> | --auto-trust-data-functions=<true|false>]
[-q <true|false> | --auto-trust-custom-queries=<true|false>]
[-n | --no-prompt]
[-i | --single-threaded]
[-v | --verbose]
[output directory]
```

Overview

Use this command for analyzing all files (of relevant types) in the library to locate scripts, data functions, and custom queries so that these (after review) can be trusted. The output of the command is a (possibly empty) report of all findings, and a (possibly empty) script that can be used for trusting the scripts, data functions, and custom queries in bulk.

Options

Option	Optional or Required	Default Value	Description
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-p value` `--library-parent-path=value`	Optional	none	The path to a library folder with the files that should be analyzed (files in sub-folders will also be included).
`-e value` `--library-search-expression=value`	Optional	none	A library search expression matching the files that should be analyzed. For information about available search parameters, see "Searching the library" in the Spotfire Analyst User's Guide.
`-s <true|false>` `--auto-trust-scripts=<true|false>`	Optional	false	This flag indicates that any untrusted scripts that are found should be trusted automatically. Use this option with care. The scripts SHOULD BE REVIEWED before being trusted.
`-d <true|false>` `--auto-trust-data-functions=<true|false>`	Optional	false	This flag indicates that any untrusted data functions that are found should be trusted automatically. Use this option with care. The data functions SHOULD BE REVIEWED before being trusted.
`-q <true|false>` `--auto-trust-custom-queries=<true|false>`	Optional	false	This flag indicates that any untrusted custom queries that are found should be trusted automatically. Use this option with care. The custom queries SHOULD BE REVIEWED before being trusted.
`-n` `--no-prompt`	Optional	none	This flag indicates that the tool should proceed without printing a warning about the potentially long execution time and prompting for confirmation to continue.
`-i` `--single-threaded`	Optional	none	This flag indicates that the analysis should be done in a single thread. (This will reduce CPU, memory, and network usage, but increase execution time.)

Option	Optional or Required	Default Value	Description
			 If the tool runs out of memory when executing the command, then specifying this flag might help.
`-v` `--verbose`	Optional	none	This flag indicates that verbose progress output should be given.
`[output directory]`	Optional	find-analysis-scripts	The directory to which the output of the tool will be written. If the directory does not exist it will be created (in the current working directory, if the path is relative).

Examples

- To run the analysis only against items that do not exceed a certain size:

```
config find-analysis-scripts --library-search-
expression="content_size:<500MB"
```

- To run the analysis only against the items in a particular folder:

```
config find-analysis-scripts --library-parent-path="/
Production/Critical"
```

- To run the analysis in a single thread (to reduce memory and CPU consumption):

```
config find-analysis-scripts --single-threaded
```

- To automatically trust any data functions after running the analysis:

```
config find-analysis-scripts --auto-trust-data-functions
```

find-analysis-urls

Finds URL references in analysis (.dxp) files in the library.

```
find-analysis-urls
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n | --no-prompt]
[-r <true|false> | --resume=<true|false>]
[-v | --verbose]
[output directory]
```

Overview

Use this command for analyzing all analysis (.dxp) files in the library looking for URL references used by the "Image from URL" and "Web Page Panel" features so that these (after review) can be added to a white list. The output of the command is a (possibly empty) report of all findings, and a (possibly empty) suggested white list.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-n --no-prompt`	Optional	none	This flag indicates that the tool should proceed without printing a warning about the potentially long execution time and prompting for confirmation to continue.
`-r <true|false> --resume=<true|false>`	Optional	true	This flag indicates whether the tool will resume any previous execution (requires the existence of a `progress.xml` file in the output directory).
`-v --verbose`	Optional	none	This flag indicates that verbose progress output should be given.
`output directory`	Optional	find-analysis-urls	The directory to which the output of the tool will be written. If the directory does not exist it will be created (in the current working directory, if the path is relative).

help

Displays the help overview or a specific help topic.

```
help
[topic name]
```

Overview

Use this command to display the help overview or a specific help topic.

Options

Option	Optional or Required	Default Value	Description
`[topic name]`	Optional	none	The name of the help topic to be displayed.

import-config

Imports a server configuration from a file to the server database.

```
import-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-c value | --comment=value>
[-d <true|false> | --delete-file=<true|false>]
[import file]
```

Overview

Use this command to import a server configuration from a file to the server database and to set it as the current configuration. Such a server configuration file can be generated either by running the export-config command or by creating a new default configuration by using the create-default-config command. If an identical configuration file already exists in the server database, the existing configuration will have its description and modification date updated.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-c value --comment=value`	Required	none	A comment describing the reason for the configuration change. Make sure to enclose the specified comment in quotation marks and to quote all special characters that might otherwise be consumed by the command line shell.
`-d <true|false> --delete-file=<true|false>`	Optional	false	Indicates whether the imported configuration file should be deleted from the file system after a successful import.
`[import file]`	Optional	configuration.xml	The path to the configuration file to import.

import-groups

Imports groups to the user directory.

```
import-groups
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-m <true|false> | --include-member-groups=<true|false>]
[-u <true|false> | --include-member-users=<true|false>]
[-g <true|false> | --include-guids=<true|false>]
```

```
[-n <true|false> | --has-name-row=<true|false>]
[import file]
```

Overview

Use this command to import all groups in a given file to the user directory. The groups can be imported including membership information or as a simple list.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	configuration.xml	The path to the configuration file to create.
`-m <true|false> --include-member-groups=<true|false>`	Optional	false	Indicates whether the group hierarchy information (groups in groups) should be included. Can be used in conjunction with the `--include-member-users` argument to include all information.
`-u <true|false> --include-member-users=<true|false>`	Optional	false	Indicates whether the group hierarchy information (users in groups) should be included. Can be used in conjunction with the `--include-member-groups` argument to include all information.
`-g <true|false> --include-guids=<true|false>`	Optional	false	Indicates whether globally unique identifiers (GUIDs) in the file should be included.
`-n <true|false> --has-name-row=<true|false>`	Optional	false	Indicates whether the file contains a name row. Applicable only when the file is in plain CSV format because the Spotfire Text Data Format (STDF) always has a name row.
`[import file]`	Optional	groups.txt	The path to the file to import.

import-jaas-config

Imports new JAAS application configurations into the server configuration.

```
import-jaas-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-f | --force]
<-j value | --jaas-config-file=value>
[-n value | --name=value]
```

Overview

Use this command to import new JAAS application configurations into the server configurations.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-f --force`	Optional	none	Indicates that the JAAS application configurations should be imported into the server even if other configurations with the same names already exist. When this argument is enabled, the old configurations are overwritten
`-j value --jaas-config-file=value`	Required	none	The path to the JAAS application configuration file. The file is expected to be in the standard JAAS application configuration format.
`-n value --name=value`	Optional	none	The names of the JAAS application configurations to be imported into the server. Multiple names must be comma-separated and enclosed between quotes. If this argument is omitted, then all JAAS application configurations within the specified file are imported.

import-library-content

Imports content into the library.

```
import-library-content
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-p value | --file-path=value>
<-m value | --conflict-resolution-mode=value>
<-u value | --user=value>
[-e <true|false> | --prune-empty-directories=<true|false>]
[-a <true|false> | --include-access-rights=<true|false>]
[-i value | --item-type=value]
[-l value | --library-path=value]
```

Overview

Use this command to import content into the library.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	true	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-p value --file-path=value`	Required	none	The file system path to the file that should be imported into the library. This should be the result of a previous library export and with a name ending with .part0.zip. If the export consists of several parts (ending with .part1.zip and so on), these must be placed in the same folder.
`-m value --conflict-resolution-mode=value`	Required	none	Sets the conflict resolution mode that should be used if there is a conflict with existing content in the library path given. The conflict resolution mode is applied for each conflicting item that is imported. Valid values are `KEEP_NEW`, `KEEP_OLD`, and `KEEP_BOTH`.
`-u value --user=value`	Required	none	The user performing the import should be a Library Administrator. Unless the user is part of the configured default domain, the name of the user needs to include the user's domain name, like DOMAIN\user or user@domain.
`-e <true|false> --prune-empty-directories=<true|false>`	Optional	false	Specifies if empty directories should be created.
`-a <true|false> --include-access-rights=<true|false>`	Optional	true	Specifies if access rights should be imported.
`-i value --item-type=value`	Optional	all_items	Which item types that should be imported into the library. Valid values are: `all_items`, `colorschemes`, `information_model`, `analysis_files`, and `datafunctions`.
`-l value --library-path=value`	Optional	/	The path in the library where the content is imported. The path must specify an existing folder in the library.

import-rules

Imports routing rules and schedules to the server.

```
import-rules
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value>
<-p value | --exported-file-path=value>
[-r value | --resource-pool-name=value]
[-u <true|false> | --use-default-resource-pool=<true|false>]
[-d <true|false> | --disabled=<true|false>]
[-s value | --site-name=value]
[-R value | --rule-conflict-resolution=value]
[-S value | --schedule-conflict-resolution=value]
[-e <true|false> | --test-run=<true|false>]
[-i <true|false> | --ignore-unavailable-files=<true|false>]
```

Overview

Use this command to import all the routing rules and schedules from the given file to the server.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command prompts the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-p value --exported-file-path=value`	Required	none	The path to the file containing the rules and schedules to import.
`-r value --resource-pool-name=value`	Optional	none	A resource pool name that can be used if the resource pool for a given rule is not found. The **`--resource-pool-name`** and **`--use-default-resource-pool`** arguments are mutually exclusive.
`-u <true|false> --use-default-resource-pool=<true|false>`	Optional	false	If enabled and the resource pool for a given rule is not found, the default resource pool is used instead, and the instances count is automatically reset to one instance. The **`--resource-pool-name`** and **`--use-default-resource-pool`** arguments are mutually exclusive.
`-d <true|false> --disabled=<true|false>`	Optional	false	If `true`, all of the rules are imported in a disabled state.
`-s value --site-name=value`		none	The name of a site into which the routing rules and schedules are imported.
`-R value rule-conflict-resolution=value`	Optional	fail	Defines how to handle importing a rule if there already exists a rule with the same name and the same file/user/group. The argument can be one of the following. • `fail` • `replace` • `skip`
`-S value --schedule-conflict-resolution=value`	Optional	rename	Defines how to handle copying a shared schedule if there already exists a shared schedule with the same name in the target

Option	Optional or Required	Default Value	Description
			server. The argument can be one ofthe following. • `rename` • `replace` If the schedules are identical, the schedule in the target server remains as it was. If the names are the same but the schedules are different, the **`schedule-conflict-resolution`** parameter determines whether the schedule in the target server should be renamed or replaced.
`-e <true|false> --test-run=<true|false>`	Optional	false	If `true`, the import does not take place. The command produces a preview of the import status of each rule/schedule.
`[-i <true|false> --ignore-unavailable-files=<true|false>]`	Optional	false	Defines how to scheduled jobs if jobs or files are not available. If `true`, then the associated scheduled jobs or routing rules are not created.

import-scheduled-updates

Imports scheduled updates from previous Spotfire Web Player versions, from either a local file or the library.

```
import-scheduled-updates
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
[-p value | --local-file-path=value]
[-n value | --library-file-name=value]
[-r value | --resource-pool-name=value]
[-z value | --time-zone-id=value]
[-e <true|false> | --enabled=<true|false>]
[-i value | --instances-count=value]
[-s value | --site-name=value]
```

Overview

Use this command to import scheduled updates from previous Spotfire Web Player versions, from either a local file or the library. At least one Spotfire Server instance must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-p value --local-file-path=value`	Optional	none	Full path to the local scheduled updates file. Mutually exclusive with the `library-file-name`.
`-n value --library-file-name=value`	Optional	none	Name of the scheduled updates file in the library (specified in the previous Spotfire Web Player configuration). Mutually exclusive with the `local-file-path`.
`-r value --resource-pool-name=value`	Optional		Optional resource pool for the scheduled updates. If unspecified, default routing applies.
`-z value --time-zone-id=value`	Optional	none	Optional time zone ID in the Area/City format, for example "America/Los_Angeles" or "Europe/Brussels" (a full list is available in the server). If unspecified, server time zone applies.
`-e <true|false> --enabled=<true|false>`	Optional	false	Optional flag to specify if the scheduled updates are enabled when imported.
`-i value --instances-count=value`	Optional	1	Optionally specifies on how many Spotfire Web Player instances the scheduled updates should run. '0' means all available.
`-s value --site-name=value`	Optional	none	The name of the site that the scheduled updates should be imported to. If no site is given, the scheduled updates will be imported to the default site.

import-service-config

Imports a service configuration.

```
import-service-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n value | --config-name=value]
[-d | --delete-directory]
[source directory]
[--apply=<true|false>]
```

Overview

Use this command to import a service configuration. The imported configuration can be assigned to a service using the set-service-config command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-n value --config-name=value`	Optional	none	The name to give to the configuration. If no name is given, the existing configuration will be overwritten. Note that default configurations cannot be overwritten, so if the configuration to be imported was created from a default configuration, a name must be specified.
`-d --delete-directory`	Optional	none	Indicates whether or not the source directory should be deleted after a successful import.
`[source directory]`	Optional	config	The source directory containing the configuration that should be imported.
`--apply=<true|false>`	Optional	false	If you want to apply the imported service config without explicitly running the 'set-service-config' command. Note that all running instances (if any) of the service will be restarted.

import-users

Imports users to the user directory.

```
import-users
[-b value | --bootstrap-config=value]
```

```
[-t value | --tool-password=value]
[-i <true|false> | --include-passwords=<true|false>]
[-h <true|false> | --hash-passwords=<true|false>]
[-g <true|false> | --include-guids=<true|false>]
[-n <true|false> | --has-name-row=<true|false>]
[import file]
```

Overview

Use this command to import all users in a given file to the user directory. The users can be imported with or without passwords.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-i <true|false> --include-passwords=<true|false>`	Optional	false	Indicates whether passwords in the file should be included.
`-h <true|false> --hash-passwords=<true|false>`	Optional	false	Indicates whether the included passwords should be hashed during import. Should be false if the users have previously been exported from a Spotfire Server because those passwords are already hashed.
`-g <true|false> --include-guids=<true|false>`	Optional	false	Indicates whether the globally unique identifiers (GUIDs) in the file should be included.
`-n <true|false> --has-name-row=<true|false>`	Optional	false	Indicates whether the file contains a name row. Applicable only when the file is in plain CSV format because the Spotfire Text Data Format (STDF) always has a name row.
`[import file]`	Optional	users.txt	The path to the file to import.

invalidate-persistent-sessions

Invalidates all persistent sessions.

```
invalidate-persistent-sessions
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-u value | --username=value]
[-a | --all]
```

Overview

Use this command to invalidate persistent sessions for a specified user or for all users.

After the persistent sessions have been invalidated, the user(s) must re-authenticate when they next log in. Currently active sessions will remain active until the next idle timeout or absolute timeout (whichever happens first), after which the user will have to re-authenticate.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See The bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See The bootstrap.xml file for more information.
`-u value --username=value`	Required, unless the `--all` flag has been specified	none	The user for which all persistent sessions should be invalidated. Must not be specified together with the `--all` flag.
`-a --all`	Required, unless the `--username` argument has been specified	none	Indicates that all persistent sessions for all users should be invalidated. Must not be specified together with the `--username` argument.

Related concept

Persistent Spotfire sessions

list-active-service-configs

Lists active (configured) service configurations.

```
list-active-service-configs
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-s value | --site-name=value]
```

Overview

Use this command to list the active (configured) service configurations. See also the list-service-configs command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-s value --site-name=value`	Optional	Default	The name of the site for which to list the active service configurations. The list-sites command can be used to find names of all available sites.

list-addresses

Lists the addresses of a node.

```
list-addresses
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n value | --node-id=value]
```

Overview

Use this command to list the configured addresses of a node. The addresses can be configured using the set-addresses command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-n value --node-id=value`	Required	The default value is taken from the file specified with `--bootstrap-config`.	The ID of the node for which addresses should be listed. The list-nodes command can be used to find the IDs of all nodes in the collective.

list-admins

Lists the server administrators.

```
list-admins
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list the server administrators. Only direct members of the Administrator group are shown.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it. Refer to Bootstrap.xml file.

list-auth-config

Displays the current authentication configuration.

```
list-auth-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to display the current authentication configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

list-certificates

Lists the certificates that establish the trust between components within the Spotfire collective.

```
list-certificates
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-v | --valid]
[-e | --expired]
[-r | --revoked]
[-p | --pending]
```

Overview

Use this command to list the certificates that establish the trust between components within the Spotfire collective. By default, the tool displays all certificates issued by the internal CA. The output from the tool can be restricted by specifying one or more of the flags.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-v --valid`	Optional	none	When this flag is specified, the tool displays all valid certificates.
`-e --expired`	Optional	none	When this flag is specified, the tool displays all expired certificates.
`-r --revoked`	Optional	none	When this flag is specified, the tool displays all revoked certificates.
`-p --pending`	Optional	none	When this flag is specified, the tool displays all pending certificates.

list-configs

Lists all available server configurations.

```
list-configs
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-i | --include-incompatible]
[-h value | --hash-abbrev=value]
```

Overview

Use this command to list the available configurations. The current configuration is indicated by an asterisk in the left column.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-i --include-incompatible`	Optional	none	Indicates whether to include configurations incompatible with the current server version.
`-h value --hash-abbrev=value`	Optional	7	The number of hexadecimal digits (between 6 and 40) to which you want to abbreviate the configuration hash.

list-deployment-areas

Lists the deployment areas.

```
list-deployment-areas
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list the deployment areas as well as display the default deployment area.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it. Refer to Bootstrap.xml file.

list-ds-template

Lists the data source templates.

```
list-ds-template
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to list the data source templates.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

list-groups

Lists all groups.

```
list-groups
[-l value | --limit=value]
[-s value | --search-expression=value]
[-m | --list-members]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list all groups in the user directory.

Options

Option	Optional or Required	Default Value	Description
`-l value` `--limit=value`	Optional	20	The maximum number of groups to list.
`-s value` `--search-expression=value`	Optional	none	A search expression that can be used to search only for groups with names matching the expression.
`-m value` `--list-members`	Optional	none	Determines whether to list the members.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.

list-jaas-config

Lists the JAAS application configurations.

```
list-jaas-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--xml]
[JAAS application configuration name]
```

Overview

Use this command to display the server JAAS application configurations. (It cannot display system JAAS application configurations.)

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--xml`	Optional	none	Specifies if the JAAS application configurations should be displayed in XML format, as it is stored within the `configuration.xml` file.
`[JAAS application configuration name]`	Optional	none	The names of the JAAS application configuration to display. Multiple names must be comma-separated and enclosed between quotes. If this argument is omitted, then all JAAS application configurations are displayed.

list-jmx-users

Lists all JMX users.

```
list-jmx-users
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list all users who can access the server through JMX. The result contains the user name and access level of each user.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.

list-ldap-config

Displays LDAP configurations.

```
list-ldap-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[--xml]
[LDAP configuration id]
```

Overview

Use this command to display LDAP configurations.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--xml`	Optional	none	Specifies that the LDAP configuration should be displayed in XML format instead of the standard JAAS application configuration format.
`[LDAP configuration id]`	Optional	none	Specifies the identifier of the LDAP configuration to be displayed. If no identifier is specified, then all LDAP configurations are displayed.

list-ldap-userdir-config

Lists the configuration for the user directory LDAP mode.

```
list-ldap-userdir-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to list the configuration for the user directory LDAP mode.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

list-licenses

Lists the currently known licenses and license functions.

To get the licenses, you first must deploy Spotfire.

```
list-licenses
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list the license and license functions.

To get the licenses, you first must deploy Spotfire. Licenses will be listed by their technical names and not their display names (for example, Spotfire.Dxp.WebPlayer, rather than TIBCO Spotfire Consumer).

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.

list-logging

Lists logging templates for a specified node.

```
list-logging
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --id=value>
```

Overview

Use this command to list available logging templates for a node.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --id=value`	Required	none	The ID of the server or node manager for which the logging templates are to be listed. The list-nodes command can be used to find the IDs of all nodes.

list-nodes

Lists the nodes in the collective.

```
list-nodes
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-e | --exclude-trusted]
```

Overview

Use this command to list the nodes in the collective.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-e --exclude-trusted`	Optional	none	Indicates whether trusted nodes should be excluded.

list-ntlm-auth

Displays the NTLM authentication service configuration.

```
list-ntlm-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-S value | --server=value]
```

Overview

Use this command to display the NTLM authentication service configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-S value --server=value`	Optional	none	The name of the cluster server whose configuration should be displayed. If no name is specified, the global parameters common to all servers in the cluster are displayed.

list-oauth2-clients

Lists registered OAuth2 clients.

```
list-oauth2-clients
```

```
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
```

Overview

Use this command to list registered OAuth2 clients. Use the show-oauth2-client command to see the full configuration of a client. To use this command at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.

list-online-servers

Lists all online servers.

```
list-online-servers
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list all servers in the cluster that are currently online.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See the Bootstrap.xml file.

Output

A table of all servers in the cluster that are currently online. An asterisk in the left column is used to indicate that the server is the current primus server (responsible for handling tasks such as the synchronization of LDAP groups).

Example

```
P  Server Name           IP Address    Version
   server1.example.com   192.0.2.1     7.0.0.70
*  server2.example.com   192.0.2.2     7.0.0.60
   server3.example.com   192.0.2.3     7.0.0.70
```

list-post-auth-filter

Displays the current post-authentication filter configuration.

```
list-post-auth-filter
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to display the post-authentication filter configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

list-service-configs

Lists available service configurations.

```
list-service-configs
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-c value | --capability=value]
[-a value | --deployment-area=value]
[-e | --exclude-default-configs]
```

Overview

Use this command to list the available service configurations. The configurations can be exported using the export-service-config command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-c value --capability=value`	Optional	none	The name of the capability for which to list configurations.
`-a value --deployment-area=value`	Optional	none	The name of the deployment area for which to list configurations.
`-e --exclude-default-configs`	Optional	none	Indicates whether default configurations should be excluded.

list-service-instances

Lists the service instances in the collective.

```
list-service-instances
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-v <true|false> | --verbose=<true|false>]
```

Overview

Use this command to list the service instances in the collective.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-v <true|false> --verbose=<true|false>`	Optional	false	Show verbose information about the service.

list-services

Lists the installed services in the collective.

```
list-services
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-v <true|false> | --verbose=<true|false>]
```

Overview

Use this command to list the installed services in the collective.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-v <true|false> --verbose=<true|false>`	Optional	false	Show verbose information about the service.

list-sites

Lists the sites in the collective.

```
list-sites
```

```
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list the sites in the collective.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.

list-userdir-config

List the current user directory configuration.

```
list-userdir-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to list the current user directory configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

list-users

Lists all users.

```
list-users
[-f | --force-synchronization]
[-l value | --limit=value]
[-s value | --search-expression=value]
```

```
[-d | --display-name-search]
[-e <true|false> | --exclude-disabled=<true|false>]
[--list-extended-information]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to list all users in the user directory. It does not work when using the user directory Windows provider.

Options

Option	Optional or Required	Default Value	Description
`-f --force-synchronization`	Optional	none	Indicates that the command should force a user directory synchronization before attempting to list the users. This argument has no effect if the user directory is running in database mode.
`-l value --limit=value`	Optional	100	The maximum number of users to list.
`-s value --search-expression=value`	Optional	none	A search expression that can be used to search only for users with names matching the expression.
`-d --display-name-search`	Optional	none	Indicates whether the search expression should be used to match display names rather than user names.
`-e value --exclude-disabled=<true|false>`	Optional	false	Indicates whether disabled users should be excluded.
`--list-extended-information`	Optional	false	Indicates whether extended information such as display name, email, and last login time should be displayed for each user.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See the Bootstrap.xml file.

list-windows-userdir-config

Lists the configuration for the user directory Windows NT mode.

```
list-windows-userdir-config
```

```
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to list the configuration for the user directory Windows NT mode.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

manage-deployment-areas

Manages the deployment areas.

```
manage-deployment-areas
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-R | --reset-all-group-areas]
[-r | --reset-group-area]
[-s | --set-group-area]
[-c | --create-area]
[-D | --delete-area]
[-d | --default-area]
[-g value | --group-name=value]
[-a value | --area-name=value]
```

Overview

Use this command to change the deployment area for groups, change the default deployment area, and create and remove deployment areas.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`-R --reset-all-group-areas`	Optional	none	Use if all specified areas for all groups should be removed. This does not affect the default area or any content on the areas. Users are using the default area after running this command. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-r --reset-group-area`	Optional	none	Use if an area for a specific group should be removed. This does not affect the default area or any content on the area. If a user is not a member of any group with a specified area, the default area is used. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-s --set-group-area`	Optional	none	Use if an area should be set for a specific group. A user that is a member of this group gets access to the specified area instead of the default area. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-c --create-area`	Optional	none	Specifies that a new area should be created. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-D --delete-area`	Optional	none	Specifies that an existing area should be deleted. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-d --default-area`	Optional	none	Specifies that a the default area should be changed. The `--reset-all-group-areas`, `-reset-group-area`, `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area` arguments are mutually exclusive.
`-g value --group-name=value`	Optional	none	The name of the group. Applicable for `--reset-all-group-areas`, `--reset-group-area`, and `--set-group-area`.
`-a value --area-name=value`	Optional	none	The name of the area. Applicable for `--set-group-area`, `--create-area`, `--delete-area`, and `--default-area`.

modify-db-config

Modifies the common database connection configuration.

```
modify-db-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-l value | --login-timeout=value]
[-o value | --connection-timeout=value]
[-i value | --min-connections=value]
[-a value | --max-connections=value]
[-p value | --pooling-scheme=value]
[-q value]
{-Ckey=value}
[-e <true|false> | --clear-connection-properties=<true|false>]
```

Overview

Use this command to modify the common configuration for the connection to the Spotfire Server database. This configuration (which affects all servers) is merged with the configuration in the `bootstrap.xml` file on each server.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-l value --login-timeout=value`	Optional	none	The maximum time (in seconds) to wait for a connection to become available.
`-o value --connection-timeout=value`	Optional	none	The maximum time (in seconds) that a connection can stay idle in the connection pool before being closed and discarded.
`-i value --min-connections=value`	Optional	none	The minimum number of connections to keep in the connection pool.
`-a value --max-connections=value`	Optional	none	The maximum number of connections to keep in the connection pool.
`-p value --pooling-scheme=value`	Optional	none	The connection pooling algorithm to be used. Valid values are: • `WAIT`: The `--max-connections` parameter is strictly respected. • `DYNAMIC`: The number of connections can occasionally exceed the configured maximum number.
`-q value`	Optional	none	An SQL query that should be run directly after a connection has been created.
`-Ckey=value`	Optional	none	A JDBC connection property that is added to the existing list of connection properties. Several properties can be specified. (Can be specified multiple times with different keys.)
`-e <true|false> --clear-connection-properties=<true|false>`	Optional	false	Clears the existing list of connection properties.

Examples

Setting the maximum number of connections in the pool:

```
config modify-db-config  --max-connections=100
```

Setting the pooling scheme:

```
config modify-db-config  --pooling-scheme=WAIT
```

Setting the size of the statement pool of the DataDirect driver:

```
config modify-db-config  -CMaxPooledStatements=20
```

modify-ds-template

Modifies a data source template.

```
modify-ds-template
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
[-e <true|false> | --enable=<true|false>]
[-r value | --rename=value]
[-d value | --definition=value]
```

Overview

Use this command to modify a data source template used by Information Services.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name=value`	Required	none	The name of the data source template to modify.
`-e <true|false> --enable=<true|false>`	Optional	none	Indicates whether the data source template should be enabled. If no argument is given, the value is unchanged.
`-r value --rename=value`	Optional	none	The name to rename the data source template to. If no argument is given, the value is unchanged.
`-d value --definition=value`	Optional	none	The path to the file containing a new data source template definition. If no argument is given, the value is unchanged.

promote-admin

Assigns full administrator privileges to a user.

```
promote-admin
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
```

Overview

Use this command to promote a user to administrator by adding the user account to the Administrator group.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-u value --username=value`	Required	none	The name of the user to be promoted to administrator. Unless the user is part of the configured default domain, the name of the user must include the user's domain name, as in "DOMAIN\user" or "user@domain".

register-api-client

Registers a new API client.

```
register-api-client
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-n value | --name=value>
{-Svalue}
[-p value | --client-profile=value]
{-Gvalue}
{-Rvalue}
{-Cvalue}
[-r <true|false> | --require-end-user-consent=<true|false>]
```

Overview

Use this command to register a new OAuth2 client that can access the public web service APIs. All information needed to use the client, including a client ID and a client secret, will be shown after successful completion of the command. To use this command at least one server in the collective must be running. The list-oauth2-clients command can be used to find the IDs of all existing clients. Registered clients can be deleted using the delete-oauth2-client command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-n value --name=value`	Required	none	The name of the client to be created. Only used for display purposes, not guaranteed to be unique.
`-Svalue`	Required, but may be specified multiple times with different values.	none	A scope (case sensitive) that the client should be authorized to request. Refer to the API documentation for valid values.
`-p value --client-profile=value`	Optional	other	The client profile, can be one of 'web', 'native', or 'other'.
`-Gvalue`	Optional, and may be specified multiple times with different values.	client_credentials The value 'refresh_token' can only be specified together with 'authorization_code'.	The grant types that the client should be able to use. Can be one of 'authorization_code', 'client_credentials', or 'refresh_token'. The default value is 'client_credentials'. The value 'refresh_token' can only be specified together with 'authorization_code'.
`-Rvalue`	Optional, and may be specified multiple times with different values.	none	An authorized redirect URI. Must be specified when using the 'authorization_code' grant type, unless the client profile is 'native'. Must be an absolute URI. Must have a protocol. Cannot contain a query or fragment component.
`-Cvalue`	Optional, and may be specified multiple times with different values.	none	An authorized custom URI scheme. May only be specified when the client profile is 'native'.
`-r <true|false> --require-end-user-consent=<true|false>`	Optional	The default value depends on the client profile ('false' for the client profile 'other', 'true' for for all other client profiles).	Indicates whether the client should be required to request end-user consent (when using the 'authorization_code' grant). This argument is optional. If the 'security.oauth2.client.must-require-consent' configuration property is 'true', then the value of this argument must be 'true' unless the value of **`--client-profile`** is 'other'.

Examples

In all examples below, the client wants to be able to perform uploads to the Spotfire library.

- Register a client with the profile 'other' (e.g. a headless application acting on its own behalf):

```
register-api-client --name="Other client" -Sapi.rest.library.upload --client-profile=other
 -Gclient_credentials
```

- Register a client with the profile 'web' (e.g. a server-side web application), acting on behalf of an end-user:

```
register-api-client --name="Web client" -Sapi.rest.library.upload --client-profile=web -
Gauthorization_code -Rhttps://example.com/foo/return
```

- Register a client with the profile 'web' (e.g. a server-side web application), acting on behalf of an end-user, that may use long-lived refresh tokens (for continued access when the end-user isn't present):

```
register-api-client --name="Web client" -Sapi.rest.library.upload -Soffline --client-
profile=web -Gauthorization_code -Grefresh_token -Rhttps://example.com/foo/return
```

- Register a client with the profile 'native' (e.g. an iOS app), acting on behalf of an end-user, which should be allowed to make requests on behalf of the user without the user's explicit permission:

```
register-api-client --name="Native client" -Sapi.rest.library.upload --client-
profile=native -Gauthorization_code --require-end-user-consent=false
```

register-job-sender-client

Registers a new Automation Services Client Job Sender client.

```
register-job-sender-client
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-n value | --name=value>
```

Overview

Use this command to register a new OAuth2 client that can be used with the Automation Services Client Job Sender. All information needed to use the client, including a client ID and a client secret, will be shown after successful completion of the command. To use this command, at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-n value --name=value`	Required	none	The name of the client to be created. Only used for display purposes, and not guaranteed to be unique.

remove-config-property

Removes the value(s) of a specific configuration property.

```
remove-config-prop
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
```

Overview

Use this command to remove the value(s) of a specific configuration property. Note that this command does not perform any validation of the provided values - use with caution.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name=value`	Required	none	The name of the configuration property.

Example

To remove the configuration setting for the absolute session timeout:

```
config remove-config-prop --name="security.absolute-session-
timeout"
```

remove-ds-template

Removes a data source template.

```
remove-ds-template
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
```

Overview

Use this command to remove a data source templates.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name=value`	Required	none	The name of the data source template to remove.

remove-jaas-config

Removes the specified JAAS application configurations from the server configuration.

```
remove-jaas-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
```

Overview

Use this command to remove JAAS application configurations from the server.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value --name=value`	Required	none	The names of the JAAS application configurations to be removed from the server. Multiple names must be comma-separated and enclosed between quotes.

remove-ldap-config

Removes LDAP configurations.

```
remove-ldap-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<LDAP configuration ids>
```

Overview

Use this command to remove LDAP configurations.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`<LDAP configuration ids>`	Required	none	Specifies a comma-separated list of identifiers of the LDAP configurations to be removed.

remove-license

Removes a license from a group.

```
remove-license
<-g value | --group=value>
<-l value | --license=value>
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to remove a license from a group.

Options

Option	Optional or Required	Default Value	Description
`-g value` `--group=value`	Required	none	The group to have its licenses removed.
`-l value` `--license=value`	Required	none	The license to remove.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value` `--tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.

reset-trust

Resets the trust within the Spotfire collective.

```
reset-trust
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-d | --delete]
[-f | --force]
```

Overview

Use this command to reset the trust within the Spotfire collective by revoking all the certificates in the internal CA. When the `--delete` argument is provided, the certificates are deleted instead of revoked.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-d --delete`	Optional	none	When this flag is specified, the tool deletes the certificates in the internal CA instead of just revoking them.
`-f --force`	Optional	none	When this flag is specified, the tool revokes or deletes the certificates in the internal CA without requiring any confirmation.

revoke-consent

Revokes consent that a specific user has given an OAuth2 client (or all such clients).

```
revoke-consent
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
[-i value | --client-id=value]
```

Overview

Use this command to revoke the consent that a specific user has given to a specific OAuth2 client, or to all such clients.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-u value --username=value`	Required	none	The name of the user for which to revoke consent.
`-i value --client-id=value`	Optional	none	The ID of the client for which to revoke consent. If no client ID is given, then consent by the user is revoked for all clients.

run

Runs a configuration script.

```
run
<script file>
{-Vkey=value}
[-E | --include-environment]
[-F | --fail-on-undefined-variable]
```

Overview

Use this command to run a configuration script.

Options

Option	Optional or Required	Default Value	Description
`<script file>`	Required	none	The name of the script to be executed.
`-Vkey=value`	This argument is optional and may be specified multiple times with different keys.	none	A script variable, will take precedence over SET command in the script file.
`-E` `--include-environment`	Optional	none	When set, all environment variables are included and can be used as script variables.  the precedence order is as follows. 1. command line variables. 2. script variables (using SET command). 3. environment variables.
`-F` `--fail-on-undefined-variable`	Optional	none	When set, the command fails if an undefined variable is found.

Script Syntax

Each line must contain the name of a command and its arguments. Arguments can be quoted using either single or double quotation marks. Lines beginning with a hash character (#) are regarded as comments and have no effect. Lines ending with a backslash character (\) are continued on the next line with the backslash character removed before parsing.

The special script command "**echo**" can be used to echo messages to the console. See Script language.

s3-download

Downloads the data of library items in Amazon S3 storage.

```
s3-download
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-i value | --items=value>
<-d value | --destination=value>
```

Overview

Use this command to download the data of library items in Amazon S3 storage.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it. Refer to Bootstrap.xml file.
`-i value --items=value`	Required	none	A comma-separated list of the library items (GUIDs) to download.
`-d value --destination=value`	Required	none	The directory where the downloaded items should be saved.

set-addresses

Sets the addresses for a Spotfire Server node.

```
set-addresses
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n value | --node-id=value]
{-Avalue}
[-d | --auto-detect]
[-i <true|false> | --use-only-ips=<true|false>]
```

Overview

Use this command to set the (back-end) addresses (host names and IP addresses) of the Spotfire Server node, used for internal communication within the Spotfire collective. Ensure that the node can be reached on all addresses. The back-end ports *must* be reachable through the configured addresses, and the front-end port may be reachable through the configured addresses.

The server being configured must be offline when running the command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-n value --node-id=value`	Optional	The default value is taken from the file specified with `--bootstrap-config`.	The ID of the node for which the addresses should be set. The list-nodes command can be used to find the IDs of all nodes in the collective.
`-Avalue`	Required, unless the `--auto-detect` flag is specified, and may be specified multiple times with different values.	The default value is the host name(s) and IP address(es) as determined when this command is run.	The possible node backend addresses (host names and IP addresses). Used for internal communication within the Spotfire collective. The addresses will be used in the order they are provided (in cases where there is a need for ordering). The `-Avalue` and `--auto-detect` arguments are mutually exclusive.
`-d --auto-detect`	Required, unless at least one `-Avalue` argument is specified.	none	If specified, this argument indicates that the addresses should be determined automatically. Must only be specified when configuring the addresses of the server node where the command is run. The `-Avalue` and `--auto-detect` arguments are mutually exclusive.
`-i <true|false> --use-only-ips=<true|false>`	Optional	false	When this flag is specified, auto detection of hostnames and IP addresses will be limited to include only IP addresses. This argument and the `-Avalue` argument are mutually exclusive, and IP addresses are detected automatically only if the `-Avalue` argument is not used.

set-config

Sets the current server configuration.

```
set-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-h value | --hash=value>
<-c value | --comment=value>
```

Overview

Use this command to set the current configuration to one of the existing configurations. See list-configs for more information.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-h value --hash=valuey`	Required	none	The (possibly abbreviated) hash of the configuration to set. Must be at least the first six hexadecimal characters of the hash.
`-c value --comment=value`	Required	none	A comment describing the reason for the configuration change.

set-config-list-prop

Sets one or more values of a specific configuration list property.

```
set-config-list-prop
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
<-i value | --item-name=value>
{-Vvalue}
```

Overview

Use this command to set the value of a specific configuration list property. There must be at most one such property, and the value of the property must be representable as a string.

This command does not perform validation of any of the provided values; use with caution.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	`configuration.xml`	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value` `--name=value`	Required	none	The name of the configuration property.
`-i value` `--item=value`	Required	none	The name of each list item for the specified property.
`-Vvalue`	Required	none	The new value of the configuration property. It replaces any existing value. Can be specified multiple times with different values.

Example

To set two specific enabled TLS cipher suites for JMX connections:

```
set-config-list-prop --name="jmx.rmi.enabled-tls-
cipher-suites" --item-name="enabled-tls-cipher-
suite" -VTLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 -
VTLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
```

set-config-map-prop

Sets one or more values of a specific configuration map property.

```
set-config-list-prop
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
<-i value | --item-name=value>
{-Vkey=value}
```

Overview

use this command to set the value of a specific configuration map property. There must be at most one such property, and the value of the property must be representable as a string.

This command does not perform validation of any of the provided values; use with caution.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	`configuration.xml`	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value` `--name=value`	Required	none	The name of the configuration property.
`-i value` `--item=value`	Required	none	The name of each map for the specified property.
`-Vkey`	Required	none	The new value of the configuration property. It replaces any existing value. Can be specified multiple times with different values.

Example

To set a Post-Authentication Filter parameter called `"debug"` to `"true"`:

```
set-config-map-prop --name="security.post-authentication-
filter.init-parameters" --item-name="parameter" -Vdebug=true
```

set-config-prop

Sets the value of a specific configuration property.

```
set-config-prop
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<-n value | --name=value>
<-v value | --value=value>
[-e <true|false> | --encrypt=<true|false>]
```

Overview

Use this command to set the value of a specific configuration property. There must be at most one such property and the value of the property must be representable as a string.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-n value` `--name=value`	Required	none	The name of the configuration property.
`-v value` `--value=value`	Required	none	The new value of the configuration property. This will replace any existing value.
`-e <true|false>` `--encrypt=<true|false>`	Optional	false	Indicates whether the value should be stored encrypted.

Example

To set the absolute session timeout to one hour:

```
config set-config-prop --name="security.absolute-session-timeout"
 --value="60"
```

set-db-config

Sets the common database connection configuration.

```
set-db-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
[-l value | --login-timeout=value]
[-o value | --connection-timeout=value]
[-i value | --min-connections=value]
[-a value | --max-connections=value]
[-p value | --pooling-scheme=value]
[-q value]
{-Ckey=value}
```

Overview

Use this command to set the common configuration for the connection to the Spotfire Server database. This configuration (which affects all servers) is merged with the configuration in the `bootstrap.xml` file on each server.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-l value --login-timeout=value`	Optional	10	The maximum time (in seconds) to wait for a connection to become available.
`-o value --connection-timeout=value`	Optional	600	A comma-separated list of the library items (GUIDs) to download.
`-i value --min-connections=value`	Optional	5	The minimum number of connections to keep in the connection pool.
`-a value --max-connections=value`	Optional	40	The maximum number of connections to keep in the connection pool.
`-p value --pooling-scheme=value`	Optional	WAIT	The connection pooling algorithm to be used. Valid values are: • `WAIT`: The --max-connections parameter is strictly respected. • `DYNAMIC`: The number of connections can occasionally exceed the configured maximum number.
`-q value`	Optional	none	An SQL query that should be run directly after a connection has been created.
`-Ckey=value`	Optional	none	A JDBC connection property. Several properties can be specified.

Examples

To set the maximum number of connections in the pool:

```
config set-db-config  --max-connections=100
```

To set the pooling scheme:

```
config set-db-config  --pooling-scheme=WAIT
```

To set the size of the statement pool of the DataDirect driver:

```
config set-db-config   CMaxPooledStatements=20
```

set-license

Sets a license and license functions for a group. To see the currently available licenses and license functions, use the `list-licenses` command.

```
set-license
<-g value | --group=value>
<-l value | --license=value>
[-f value | --functions=value]
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to set a license and license functions for a group.

Options

Option	Optional or Required	Default Value	Description
`-g value` `--group=value`	Required	none	The group that should get the licenses set.
`-l value` `--license=value`	Required	none	The license to set. If no license function is provided using the `--functions` parameter, then all license functions belonging to that license are inherently enabled.

`-f value`
`--`
`functions=value`
	Optional	none	The license functions to enable.

`-b value`
`--bootstrap-`
`config=value`
	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

`-t value`
`--tool-`
`password=value`
	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.

set-logging

Set logging for a specified node.

```
set-logging
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --id=value>
[-p value | --local-file-path=value]
[-n value | --template-file-name=value]
```

Overview

Use this command to set specific logging levels using a custom properties file/template on a specified node.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --id=value`	Required	none	The ID of the server or node manager for which the logging templates/file is to be applied. The list-nodes command can be used to find the IDs of all nodes.
`-p value --local-file-path=value`	Optional	none	The full path of the logging file that will be used to set logging levels.
`-n value --template-file-name=value`	Optional	none	The template file name which should be used to set the loggers for the node. The list-logging command can be used to find the template files of a node.

set-preference

Sets a preference for a group.

```
set-preference
<-g value | --group=value>
<-c value | --category=value>
<-p value | --type=value>
<-n value | --name=value>
[-v value | --value=value]
<-T value | --value-type=value>
{-Vvalue}
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to set a preference for a group.

There is no validation of entered values when using this command. If invalid values are entered, then nothing will happen and you cannot see any changes in the Administration Manager. If a preference value has been previously specified for a group using Administration Manager, you can use show-preference to see the current values and their formats before using the `set-preference` command.

Options

Option	Optional or Required	Default	Description
`<-g value | --group=value>`	Required	none	The group to set the preference for.
`<-c value | --category=value>`	Required	none	The preference category to set.
`<-p value | --type=value>`	Required	none	The preference type to set.
`<-n value | --name=value>`	Required	none	The preference name to set.
`[-v value | --value=value]`	Optional	none	The single preference value to set. Used for all value types that are not arrays.
`<-T value | --value-type=value>`	Required	`raw`	The value type for the preference to set. Valid values are `boolean`, `integer`, `string`, `stringArray` and `raw` (i.e., what is stored in the database).
`{-Vvalue}`	Optional	none	The preference array value to set , used with value type stringArray. This argument can be specified multiple times with different values.
`[-b value | --bootstrap-config=value]`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`[-t value | --tool-password=value]`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.

set-public-address

Configures the public address.

```
set-public-address
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-s value | --site-name=value]
[-u value | --url=value]
```

Overview

Use this command to configure the public address that should be used when generating absolute URLs. A public address must be configured if the Spotfire Server is accessed through a load balancer or reverse proxy.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`-s value --site-name=value`	Optional if there is a local server (in which case the site of that server will be used) or if there is only one site available (in which case that site will be used).	none	The name of the site for which to set the public address. The list-sites command can be used to find names of all available sites.
`-u value --url=value`	Optional	none	The public address URL to use, for example "http[s]://host[:port]/". If no URL is specified, any existing value will be cleared and the public address will be automatically determined during Spotfire Server startup.

If the public address is changed after the system is initially started, first restart the servers and then restart all the service instances to propagate the change.

set-server-service-config

Sets the configuration for a service running in Spotfire Server (typically the Spotfire Web Player front-end).

```
set-server-service-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-s value | --site-name=value]
[-a value | --capability=value]
[-c value | --config-name=value]
```

Overview

Use this command to set the configuration for a service running in Spotfire Server.

After setting the configuration, you must restart the affected servers.

To configure a service running on a remote node, use the set-service-config command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-s value --site-name=value]`	Optional if there is a local server (in which case the site of that server will be used) or if there is only one site available (in which case that site will be used).	none	The name of the site for which to set the configuration. The list-sites command can be used to find names of all available sites.
`-a value --capability=value`	Optional	WEB_PLAYER	The name of the capability for which to set the configuration.
`-c value --config-name=value`	Optional	none	The name of the configuration that should be set. If no configuration name is specified, the service will revert to the default configuration.

set-service-config

Sets the configuration for a service running on a remote node.

```
set-service-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-s value | --service-id=value>
[-c value | --config-name=value]
[-f | --force]
```

Overview

Use this command to set the configuration for a service running on a remote node. Note that all running instances (if any) of the service will be restarted.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-s value --service-id=value`	Required	none	The ID of the service for which the service should be set.
`-c value --config-name=value`	Optional	none	The name of the configuration that should be set. If no configuration name is specified, the service reverts to the default configuration.
`-f --force`	Optional	none	Indicates that the service configuration should be set without need for further confirmation.

set-site

Sets the site to which a node should belong.

```
set-site
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-n value | --node-id=value]
<-s value | --site-name=value>
```

Overview

Use this command to assign a node to a site.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-n value --node-id=value`	Optional	The default value is taken from the file specified with `--bootstrap-config`.	The ID of the node for which the site should be set. The list-nodes command can be used to find the IDs of all nodes in the collective.
`-s value --site-name=value`	Required	none	The name of the site to which the node should belong. The list-sites command can be used to find names of all available sites. New sites can be created using the create-site command.

set-user-password

Sets a new password for a given user.

```
set-user-password
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-u value | --username=value>
[-p value | --password=value]
```

Overview

Use this command to set the password for a specific user account.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-u value --username=value`	Required	WEB_PLAYER	The name of the user for which the password should be set.
`-p value --password=value`	Optional	none	The new password.

show-basic-ldap-auth

Shows the LDAP authentication source for use with the BASIC authentication method.

```
show-basic-ldap-auth
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to show the LDAP authentication source(s) for use with the BASIC authentication method. The configuration is stored within the Spotfire LDAP JAAS application configuration.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

show-config-history

Shows the configuration history.

```
show-config-history
```

```
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-h value | --hash-abbrev=value]
```

Overview

Use this command to show the configuration history. The most recent entry is the current configuration.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-h value --hash-abbrev=value`	Optional	7	The number of hexadecimal digits to abbreviate the configuration hash to. Must be a number between 6 and 40.

show-deployment

Shows the current deployment.

```
show-deployment
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-a value | --area=value]
[-s | --show-ids]
```

Overview

Use this command to show the current deployment in a given area.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-a value --area=value`	Optional	none	The deployment area for which to show the current deployment. If no area is specified, the deployment of the default area is showed.
`-s --show-ids`	Optional	none	Indicates whether the package IDs should be included in the output. A package ID is needed to remove a specific package using the update-deployment command. For more information, see update-deployment.

show-import-export-directory

Shows the library import/export directory.

```
show-import-export-directory
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to display the library import/export directory. All library import and export operations are done from and to this directory, which can be a local directory or can reside on a shared disk.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

show-join-database

Shows the configured default join database.

```
show-join-database
```

```
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
```

Overview

Use this command to show the configured default join database, used by Information Services.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value` `--bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.

show-library-permissions

Shows permissions set in the library.

```
show-library-permissions
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-l value | --library-path=value>
[-r <true|false> | --recursive=<true|false>]
[-x <true|false> | --expand-groups=<true|false>]
[-d <true|false> | --downward=<true|false>]
[-p value | --path-to-report=value]
[-f <true|false> | --force-overwrite=<true|false>]
```

Overview

Use this command to create a report file that shows the permissions in the library.

Permissions are set on directories. if no permission is set, the directory inherits the permissions from the directory above.

You can use this command in three different ways:

- It can show if any permissions are set explicitly on a directory.
- It can show what permissions are in effect on a certain directory. If no permissions are set on the directory itself, it will continue upwards until it finds the directory from which the permissions are inherited (see `recursive` option).
- It can be used to report on all directories with permissions explicitly set in a branch of the directory (see the `downward` option).

The resulting file should be possible to read in Spotfire. It has headers that explain the display in the different columns.

This command may take some time to run. Also, you may need to increase the Java memory allocation to run the command, especially if the users are displayed.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file help topic for more information.
`-l value --library-path=value`	Required	none	The path in the library to start to report with (must start with a /).
`-r <true|false> --recursive=<true|false>`	Optional	false	If no permission is set on this directory, continue upwards until permissions are found.
`-x <true|false> --expand-groups=<true|false>`	Optional	false	Specifies whether groups are expanded to show their members. Members of the Administrator and Library Administrator group can see all content. When `expand-groups` is "true", these implicit rights are also taken into account, and these groups and their members are also displayed.
`-d <true|false> --downward=<true|false>`	Optional	false	Lists permissions on an entire branch of the library, and shows only folders where permissions are set explicitly. (This option takes precedence over the recursive option.)
`-p value --path-to_report=value`	Optional	none	The name of the report file that should be generated. If not provided, an automatic name is generated.
`-f <true|false> --force-overwrite=<true|false>`	Optional	false	If a name for the report file is provided but a file with that name already exists, set this option to "true" to overwrite the existing file.

show-licenses

Shows licenses set on the server.

```
show-licenses
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-l value | --license=value]
[-x <true|false> | --expand-groups=<true|false>]
[-p value | --path-to-report=value]
```

```
[-f <true|false> | --force-overwrite=<true|false>]
```

Overview

Use this command to create a report file that shows the licenses set on the server.

You can read the resulting file in Spotfire. The file has headers that explain the contents displayed in the columns. The column "From Group" contains the group on which the license is explicitly set. For every group that has a license set explicitly, the resulting groups and users (if the `expand` option is set) are shown once.

Users get the sum of all licenses (and functions). When you analyze the file, note that a user and a license might occur more than once if the user gets its licenses from more than one group with explicit licenses set.

This command may take some time to run. Also, you may need to increase the Java memory allocation to run the command, especially if the users are displayed.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file help topic for more information.
`-l value --license=value`	Optional	none	An optional, comma-separated list of licenses. If provided, the report contains only these licenses. If an invalid entry is given, the valid licenses are displayed.
`-x <true|false> --expand-groups=<true|false>`	Optional	false	Specifies whether groups are expanded to show their members. Members of the Administrator and Library Administrator group can see all content. When `expand-groups` is "true", these implicit rights are also taken into account, and these groups and their members are also displayed.
`-p value --path-to_report=value`	Optional	none	The name of the report file that should be generated. If not provided, an automatic name is generated.
`-f <true|false> --force-overwrite=<true|false>`	Optional	false	If a name for the report file is provided but a file with that name already exists, set this option to "true" to overwrite the existing file.

show-oauth2-client

Shows the configuration of a specified OAuth2 client.

```
show-oauth2-client
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --client-id=value>
[-s <true|false> | --show-client-secret=<true|false>]
```

Overview

Use this command to show the full configuration, possibly including the client secret, of a registered OAuth2 client. To use this command at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --client-id=value`	Required	none	The client ID of the client for which to show the configuration. The list-oauth2-clients command can be used to find the IDs of all clients.
`-s <true|false> --show-client-secret=<true|false>`	Optional	false	Indicates whether the client secret should be shown.

show-preference

Shows a preference for a group.

```
show-preference
<-g value | --group=value>
<-c value | --category=value>
<-p value | --type=value>
<-n value | --name=value>
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
```

Overview

Use this command to show a preference for a group.

Options

Option	Optional or Required	Default	Description
`<-g value | --group=value>`	Required	none	The group to show the preference for.
`<-c value | --category=value>`	Required	none	The preference category to show.
`<-p value | --type=value>`	Required	none	The preference type to show.
`<-n value | --name=value>`	Required	none	The preference name to show.
`[-b value | --bootstrap-config=value]`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`[-t value | --tool-password=value]`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.

switch-domain-name-style

Switches the domain names for all users and groups from one style (DNS or NetBIOS) to the other (for all configured domains).

```
switch-domain-name-style
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-n value | --new-domain-name-style=value>
```

Overview

Use this command to switch the domain names for all existing users and groups from one style (DNS or NetBIOS) to the other (for all configured domains). The new domain name style must first be configured using the config-userdir command. Note that this command is only applicable when using a user directory in LDAP mode against Active Directory.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See the Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. See Bootstrap.xml file.
`-n value --new-domain-name-style=value`	Required	none	The new domain name style. Valid values are `dns` and `netbios`.

test-jaas-config

Tests a JAAS application configuration.

```
test-jaas-config
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-c value | --configuration=value]
<-j value | --jaas-configuration=value>
<-u value | --username=value>
[-p value | --password=value]
```

Overview

Use this command to test a JAAS application configuration by performing a login attempt, using the specified credentials. It can test either a configuration stored in the server database or a configuration stored in an exported configuration file. To test a configuration stored in a configuration file, use the `--configuration` argument. Otherwise the configuration stored in the database is tested. If the JAAS login module requires a connection to the server database, the `--configuration` argument cannot be used.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. Can be specified if a password is given and `--enable-config-tool` argument is set to `true` (the default).
`-c value --configuration=value`	Optional	none	The path to an exported server configuration file. If this parameter is omitted, the application attempts to retrieve the configuration parameters from the server database using the file `bootstrap.xml`, specified by the `--bootstrap` argument.
`-j value --jaas-configuration=value`	Required	none	The name of the JAAS application configuration to test.
`-u value --username=value`	Required	none	The name of the user to log in as.
`-p value --password=value`	Optional	none	The password of the user to log in as. If the password is omitted, the command prompts the user for it.

trust

Trusts scripts, data functions, or custom queries in a file in the library.

```
trust
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-i value | --library-file-id=value>
{-Svalue}
{-Qvalue}
```

Overview

Use this command to trust scripts, data functions, or custom queries in file in the library. The input to the command is typically generated using the find-analysis-scripts command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-i value --library-file-id=value`	Required	none	The ID of the library file with which to associate the trust.
`-Svalue`	Required, unless the **`-Q`** argument has been specified.	true	The hash of a script or data function which should be trusted. May be specified multiple times with different values.
`-Qvalue`	Required, unless the **`-S`** argument has been specified.	none	The hash of a custom query which should be trusted. May be specified multiple times with different values.

trust-node

Trusts a specified node.

```
trust-node
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --id=value>
```

Overview

Use this command to trust a specified node, after which it will be a part of the collective. To use this command, at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See the Bootstrap.xml file for more information.
`k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --id=value`	Required	none	The ID of the node that should be trusted. The list-nodes command can be used to find the IDs of all nodes waiting to be trusted.

untrust

Untrusts scripts, data functions, or custom queries in a file (or all files) in the library.

```
untrust
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-i value | --library-file-id=value]
{-Svalue}
{-Qvalue}
[-n | --no-prompt]
```

Overview

Use this command to untrust scripts, data functions, or custom queries in the library that a Script Author or Custom Query Author has previously trusted. The input to the command is typically generated using the find-analysis-scripts command.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`-i value --library-file-id=value`	Optional	none	The ID of the library file with which the trust is associated. If no ID is specified then the changes will apply to all files in the library.
`-Svalue`	Optional	none	The hash of a script or data function which should be untrusted. If neither this nor **`-Q`** is specified, then the changes will apply to all scripts, data functions, and custom queries in the file (or all files if no library file ID is specified). May be specified multiple times with different values.
`-Qvalue`	Optional	none	The hash of a custom query which should be untrusted. If neither this nor **`-S`** is specified, then the changes will apply to all scripts, data functions, and custom queries in the file (or all files if no library file ID is specified). May be specified multiple times with different values.
`-n --no-prompt`	Optional	none	This flag indicates that the tool should proceed without printing a warning about the potentially severe consequences and prompting for confirmation to continue.

untrust-node

Untrusts a specified node.

```
untrust-node
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
[-k value | --keystore-file=value]
<-i value | --id=value>
```

Overview

Use this command to untrust a specified node, after which it will no longer be a part of the collective. To use this command, at least one server in the collective must be running.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end-user for it on the console. See Bootstrap.xml file for more information.
`k value --keystore-file=value`	Optional	none	The location of the keystore containing the certificates used for securing internal communication.
`-i value --id=value`	Required	none	The ID of the node that should be untrusted. The list-nodes command can be used to find the IDs of all trusted nodes.

update-bootstrap

Updates an existing bootstrap configuration file.

```
update-bootstrap
[-c value | --driver-class=value]
[-d value | --database-url=value]
[-u value | --username=value]
[-p value | --password=value]
[--clear-username-and-password]
[-k value | --kerberos-login-context=value]
[--clear-kerberos-login-context]
{-Ckey=value}
[--clear-connection-properties]
[--disable-config-tool]
[--enable-config-tool]
[-t value | --tool-password=value]
[-a value | --server-alias=value]
[-r | --prompt]
[bootstrap configuration file]
```

Overview

Use this command to update an existing bootstrap configuration file. To create a new file, use the bootstrap command. Server addresses can be set using the set-addresses command. The encryption password can be updated by using the config-encryption command. The site to which the server belongs can be changed by using the set-site command.

Options

Option	Optional or Required	Default Value	Description
`-c value` `--driver-class=value`	Optional	none	This argument specifies the name of the JDBC driver class. If not specified, the previous value is kept. Note that if you change driver you will likely also have to modify the URL (using the `--database-url` argument).
`-d value` `--database-url=value`	Optional	none	This argument specifies the JDBC URL to the database. If not specified, the previous value is kept. Because this argument usually contains special characters, make sure to escape those characters or enclose the values between quotes.
`-u value` `--username=value`	Optional	none	This argument specifies the database account's username. If not specified, the previous value (if any) is kept.
`-p value` `--password=value`	Optional	none	This argument specifies the database account's password. If not specified, the previous value (if any) is kept. Use the `--prompt` flag to indicate that the tool should prompt for the password.
`--clear-username-and-password`	Optional	none	When this flag is specified, any existing username and password will be removed. Use this to switch from username/password-based authentication to Kerberos or NTLM. Cannot be specified together with the `--username`, `--password`, or `--tool-password` arguments.
`-k value` `--kerberos-login-context=value`	Optional	none	This argument specifies the name of the JAAS application configuration to be used for acquiring the Kerberos TGT, when using the Kerberos protocol to log in to the database. If not specified, the previous value (if any) is kept unless the `--clear-kerberos-login-context` flag is specified. The JAAS application configuration must be registered with the JVM using a login.config.url parameter in the `<server installation dir>\jdk\conf\security\java.security` file (Windows) or `<server installation dir>/jdk/conf/security/java.security` file (Linux). The Spotfire Server import-jaas-config command cannot be used for this purpose because the JAAS application configurations that are imported using this command are stored in the database itself, which prevents the Spotfire Server from using them for creating the initial connection to the database.
`--clear-kerberos-login-context`	Optional	none	When this flag is specified, any previous Kerberos login context will be cleared. Cannot be specified together with the `--kerberos-login-context` argument.
`-Ckey=value`	Optional	none	A JDBC connection property. Several properties may be specified. If not specified, the previous values (if any) are kept unless the `--clear-connection-properties` flag is specified. This

Option	Optional or Required	Default Value	Description
			argument may be specified multiple times with different keys.
`--clear-connection-properties`	Optional	none	When this flag is specified, any previous connection properties will be cleared. Cannot be specified together with the `-C` argument.
`--disable-config-tool`	Optional	none	When this flag is specified the `config-tool` section (if any) will be removed from the bootstrap configuration file. Disables the use of the configuration tool with this bootstrap configuration file. Cannot be specified together with the `--enable-config-tool` argument. If neither the `--disable-config-tool` nor the `--enable-config-tool` argument is specified, the capability will remain as before.
`--enable-config-tool`	Optional	none	When this flag is specified, a `config-tool` section will be added to the bootstrap configuration file. Enables the use of the configuration tool with this bootstrap configuration file. Cannot be specified together with the `--disable-config-tool` argument. If neither the `--disable-config-tool` nor the `--enable-config-tool` argument is specified, the capability will remain as before.
`-t value --tool-password=value`	Optional	none	This argument specifies the password needed to execute most configuration tool commands. If not specified, the previous value (if any) is kept. Use the `--prompt` flag to indicate that the tool should prompt for the password.
`-a value --server-alias=value`	Optional	none	The server alias. Used for identifying the server, for example when specifying server-specific configuration. If not specified, the previous value is kept.
`-r --prompt`	Optional	none	When this flag is specified, the tool will prompt for any missing password arguments.
`bootstrap configuration file`	Optional	none	This argument specifies the path to the bootstrap configuration file to create. See Bootstrap.xml file for more information about this file.

update-deployment

Updates the current deployment.

```
update-deployment
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-a value | --area=value>
[-c | --clear]
[-r value | --remove-packages=value]
[-v value | --version=value]
[-d value | --description=value]
[-f | --force-update]
[deployment files]
```

Overview

Use this command to add a new deployment or to update the current deployment in a given area.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the file `bootstrap.xml`. If the tool password is omitted, the command prompts the user for it in the console. Refer to Bootstrap.xml file.
`-a value --area=value`	Required	none	The deployment area that should be updated.
`-c --clear`	Optional	none	Indicates that all existing packages should be removed before any new files are added. If no files are provided to add to the deployment, the deployment area is empty.
`-r value --remove-packages=value`	Optional	none	A comma-separated list of IDs of packages that should be removed from the deployment. The IDs can be determined using the show-deployment command. Should not be specified together with the `--clear` argument
`-v value --version=value`	Optional	none	The version of the new deployment. If no value is given, it is taken from the current deployment, or from the last added distribution if one is added.
`-d value -- description=value`	Optional	none	The description of the new deployment. If no value is given it is taken from the current deployment, or from the last added distribution if one is added.
`-f --force-update`	Optional	none	Indicates that users connecting to the server should be forced to update their clients.
`[deployment files]`	Optional	none	A comma-separated list of files (packages and distributions) that should be added to the deployment. Note that the paths cannot contain spaces.

update-ldap-config

Updates LDAP configurations.

```
update-ldap-config
[-c value | --configuration=value]
[-b value | --bootstrap-config=value]
<--id=value>
[-t value | --type=value]
[-s value | --servers=value]
```

```
[--clear-context-names]
[-n value | --context-names=value]
[-u value | --username=value]
[-p value | --password=value]
[--schedules=value]
[--clear-schedules]
[--user-search-filter=value]
[--user-name-attribute=value]
[--authentication-attribute=value]
[--security-authentication=value]
[--referral-mode=value]
[--referral-mode-root-dse=value]
[--request-control=value]
[--page-size=value]
[--import-limit=value]
[--user-display-name-attribute=value]
[--group-display-name-attribute=value]
{-Ckey=value}
{-Rvalue}
{-Svalue}
[--connection-timeout=value]
[--read-timeout=value]
```

Overview

Use this command to update LDAP configurations.

Options

Option	Optional or Required	Default Value	Description
`-c value --configuration=value`	Optional	configuration.xml	The path to the server configuration file.
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`--id=value`	Required	none	Specifies the identifier for the LDAP configuration to be updated.
`-t value --type=value`	Optional	none	The type of LDAP server. The following names are valid types: • `ActiveDirectory` • `SunOne` • `SunJavaSystem` • `Custom` When you specify any of the first three types, a type-specific configuration template is automatically applied in runtime so that the most fundamental configuration options are configured automatically. When you specify a Custom LDAP server type, there is no such configuration template and all those configuration options must be specified explicitly. When a custom LDAP configuration is to be used for authentication or with the user directory LDAP provider, the `--user-search-filter` and `--user-name-attribute` arguments must be specified. For such an LDAP configuration to be used for group synchronization, additional parameters must also be specified when running the config-ldap-group-sync command. See the help topic for that command for more information.
`-s value --servers=value`	Optional	none	Specifies a whitespace-separated list of LDAP server URLs. An LDAP server URL has the format **<protocol>://<server>[:<port>]**: • <protocol>: Either LDAP or LDAPS • <server>: The fully qualified DNS name of the LDAP server. • <port>: (Optional) Number indicating the port number the LDAP service is listening on. When using the LDAP protocol, the port number defaults to 389. When using the LDAPS protocol, the port number defaults to 636. Active Directory LDAP servers also provide a Global Catalog containing forest-wide information, instead of domain-wide information only. The Global Catalog LDAP service

Option	Optional or Required	Default Value	Description
			by default listens on port number 3268 (LDAP) or 3269 (LDAPS). Spotfire Server does not expect any search base, scope, filter or other additional parameters after the port number in the LDAP server URLs. Such properties are specified using other configuration options for this command. Examples of LDAP server URLs: – LDAP://myserver.example.com – LDAPS://myserver.example.com – LDAP://myserver.example.com:389 – LDAPS://myserver.example.com:636 – LDAP://myserver.example.com:3268 – LDAPS://myserver.example.com:3269
`--clear-context-names`	Optional	none	Clears context names from the LDAP configuration. This argument can be used together with the **`--context-names`** argument to remove all old context names before adding the new.
`-n value` `--context-names=value`	Optional	none	A list of distinguished names (DNs) of containers holding LDAP accounts to be visible within Spotfire Server. When specifying more than one DN, the DNs must be separated by pipe-characters (|). The specified context names are added to the context names that are already configured. To set the context names from scratch, use the `--clear-context-names` argument with the `--context-names`. If the specified containers contain a large number of users, of which only a few should be visible in Spotfire Server, a custom user search filter can be specified to include only the designated users (see the `--user-search-filter` argument). Examples: • CN=users,DC=example,DC=com • OU=project-x,DC=research,DC=example,DC=com
`-u value` `--username=value`	Optional	none	The name of the LDAP service account to be used when searching for users (and optionally also groups) in the LDAP server. This service account does not need to have any write permissions, but it needs to have read permissions for all configured context names (LDAP containers). For most LDAP servers, the account name is the account's distinguished name (DN). For Active Directory, the account name can also be specified in the forms **ntdomain\name** and **name@dnsdomain**. Examples: CN=spotsvc,OU=services,DC=research,DC=example,dc=COM RESEARCH\spotsvc (Active Directory only)

Option	Optional or Required	Default Value	Description
			spotsvc@research.example.com (Active Directory only)
`--password=value`	Optional	none	The password for the LDAP service account.
`--schedules=value`	Optional	none	A comma-separated list of schedules for when the LDAP synchronization should be performed. The schedules are given in a cron-compatible format, where each schedule consists of either five fields or one shorthand label. Make sure to enclose the value in double quotes. The specified schedules are added to the schedules that are already configured. To set the schedules from scratch, use the `--clear-schedules` argument with the `--schedules`. The five fields are, from left to right, with their valid ranges: minute (0-59), hour (0-23), day of month (1-31), month (1-12) and day of week (0-7, where both 0 and 7 indicate Sunday). A field can also be configured with the wildcard character *, indicating that any moment in time matches this field. A group synchronization is triggered when all fields match the current time. If both day of month and day of week have non-wildcard values, then only one of them has to match. There are also the following shorthand labels that can be used instead of the full cron expressions: • `@yearly` or `@annually`: run once a year (equivalent to `0 0 1 1 *`) • `@monthly`: run once a month (equivalent to `0 0 1 * *`) • `@weekly`: run once a week (equivalent to `0 0 * * 0`) • `@daily` or `@midnight`: run once a day (equivalent to `0 0 * * *`) • `@hourly`: run once an hour (equivalent to `0 * * * *`) • `@minutely`: run once a minute (equivalent to `* * * * *`) • `@reboot` or `@restart`: run every time Spotfire Server is started Refer to the Wikipedia overview article on the cron scheduler.
`--clear-schedules`	Optional	none	Clears from the LDAP configuration the LDAP synchronization schedules. This argument can be used together with the `--schedules` argument to remove all old schedules before adding the new.
`--user-search-filter=value`	Optional; must be specified for custom LDAP configurations either when running this command or the create-	For Active Directory servers, the parameter value defaults to '(&(objectClass=user)(!(objectClass=computer)))'. For any version of the Sun Directory Servers, it defaults to `objectClass=person`.	Specifies an LDAP search expression filter to be used when searching for users. If only a subset of all the users in the specified LDAP containers should be allowed access to Spotfire Server, a more detailed user search filter can be used. The search expression can, for example, be expanded so that it also puts restrictions on which groups the users belong to, or which roles they have. • For Active Directory servers, access can be restricted to only those users belonging to a certain group by using a search expression with the pattern

Option	Optional or Required	Default Value	Description
	ldap-config command. (The parameter is required for all custom configurations.)		`&(objectClass=user)(memberOf=<groupDN>)`, where `<groupDN>` is replaced by the real DN of the group to which the users must belong. If the users are divided among multiple groups, use the pattern `&(objectClass=user)(|(memberOf=<firstDN> )(memberOf=<secondDN>))`. Add extra `(memberOf=<groupDN>)` sub-expressions as needed. Active Directory example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)` • For a Sun Java System Directory Server version 6 and later, the same effect can be achieved by using a search expression with the pattern `&(objectClass=person)(isMemberOf=<groupDN>)`. If the users are divided among multiple groups, use the pattern `&(objectClass=person)(|(isMemberOf=<firstDN>)(isMemberOf=<secondDN>))`. Add extra `(isMemberOf=<groupDN>)` sub-expressions as needed. Sun Java System Directory Server example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)` • For Sun ONE Directory Servers as well as the newer Sun Java System Directory Servers or the older iPlanet Directory Server, access can be restricted to only those users having certain specific roles. The search expression for role filtering must match the pattern `&(objectClass=person)(nsRole=<roleDN>)`. If multiple roles are of interest, use the pattern `&(objectClass=person)(|(nsRole=<firstDN>)(nsRole=<secondDN>)`. Add extra (nsRole=<roleDN>) sub-expressions as needed. Sun ONE Directory Servers example: `&(objectClass=person)(isMemberOf=cn=project-x,dc=example,dc=com)` The syntax of LDAP search expression filters is specified by the RFC 4515 document. Consult this documentation for information about more advanced filters.
`--user-name-attribute=value`	Optional; must be specified for custom LDAP configurations either when running this command or the create-ldap-	For Active Directory servers the value defaults to sAMAccountName. For a Sun Java System Directory Server (or any older Sun ONE Directory Server or iPlanet Directory Server) with a default configuration, it defaults to UID.	Specifies the name of the LDAP attribute containing the user account names.

Option	Optional or Required	Default Value	Description
	config command.		
`--authentication-attribute=value`	Optional; should be used only for advanced setups. It is not set by default.	none	Specifies the name of the LDAP attribute containing a user identity that can be used for binding (authenticating) to the LDAP server. This attribute fills no purpose in most common LDAP configurations, but can be useful in more advanced setups, where the distinguished name (DN) does not work for authentication, or where users should be able to log in using a username that does not map directly to an actual LDAP account. When setting up SASL with DIGEST-MD5 in an Active Directory environment, the DN does not work for authentication and the userPrincipalName attribute must be used instead. The `--authentication-attribute` argument should then be set to `userPrincipalName` and the `--user-name-attribute` argument should be set to `sAMAccountName` (the latter value also happens to be the default value for an Active Directory LDAP configuration, so there's no need to set it explicitly). See also the `--security-authentication` argument. When setting up SASL with GSSAPI in an Active Directory environment, the DN does not work for authentication and the sAMAccountName or userPrincipalName attribute must be used instead. The `--authentication-attribute` argument should then be set to `sAMAccountName` or `userPrincipalName` and the `--user-name-attribute` argument should be set to `sAMAccountName` (the latter value also happens to be the default value for an Active Directory LDAP configuration, so there is no need to set it explicitly). See also the `--security-authentication` argument. Example: By setting the `--user-name-attribute` argument to `cn` and the `--authentication-attribute` argument to `userPrincipalName` in an Active Directory environment, the users can log in to Spotfire Server using their CN attribute values, but underneath the hood, Spotfire Server actually uses the userPrincipalName attribute value of the LDAP account with the matching CN for the actual authentication.
`--security-authentication=value`	Optional; should be used only in advanced setups.	simple	This parameter specifies the security level to use when binding to the LDAP server. • To enable anonymous binding, it should be set to `none`. • To enable plain username/password authentication, it should be set to `simple`. • To enable SASL authentication, it should be set to the name of the SASL mechanism to be used, for example DIGEST-MD5 or GSSAPI. Use multiple -C arguments to set the additional JNDI environment properties that the SASL authentication mechanism typically requires. When setting up SASL with DIGEST-MD5 in an Active Directory environment, all accounts must use reversible encryption for their passwords. This is typically not the default setting for the domain controller. The `--`

Option	Optional or Required	Default Value	Description
			`authentication-attribute` argument must also be used to specify the userPrincipalName attribute for the actual authentication to work correctly. When setting up SASL with GSSAPI in an Active Directory environment, the `--authentication-attribute` argument must be used to specify either the sAMAccountName or the userPrincipalName attribute and the custom property `kerberos.login.context.name` must be mapped to the JAAS application configuration SpotfireGSSAPI. This in turn requires a fully working Kerberos configuration file at `<server installation dir>/tomcat/spotfire-config/krb5.conf`.
`--referral-mode=value`	Optional	follow	Specifies how LDAP referrals should be handled. Valid arguments are `follow` (automatically follow any referrals), `ignore` (ignore referrals), and `throw` (fail with an error).
`[--referral-mode-root-dse=value]`	Optional	If not explicitly set, the value for `--referral-mode` is used.	Specifies how LDAP referrals should be handled when looking up the RootDSE. Valid arguments are: • follow (automatically follow any referrals) • ignore (ignore referrals) • throw (fail with an error)
`--request-control=value`	Optional	probe	Determines the type of LDAP controls to be used when executing search queries to the LDAP server. The default behavior is to probe the LDAP server for the best supported request control. The paged results control is always preferred, because it provides the most efficient way of retrieving the query result set. The virtual list view control can also be used for the same purpose if the paged results control is not supported. The virtual list view control is automatically used together with a sort control. Both the paged results control and the virtual list view control supports a configurable page size, set by the `--page-size` argument. • To explicitly configure the server for probing, set the argument value to `probe`. • To configure the server for the paged results control, set the argument value to `PagedResultsControl`. • To request the virtual list view control, set the argument value to `VirtualListViewControl`. • To completely disable request controls, set the argument value to `none`.
`--page-size=value`	Optional	The page size value defaults to 2000 for both the paged results control and the virtual list view control.	Specifies the page size to be used with the paged results control or the virtual list view control when performing search queries to the LDAP server
`--import-limit=value`	Optional	unlimited	Specifies a threshold that limits the number of users that can be imported from an LDAP server to Spotfire Server in one query. This can be used to prevent accidental flooding of the Spotfire Server user directory

Option	Optional or Required	Default Value	Description
			when integrating with an LDAP server with tens or even hundreds of thousands of users. By setting an import limit, the administrator can be sure that an unexpected high number of users does not affect the server performance. By default, there is no import limit. To explicitly request unlimited import, set the parameter value to -1. All positive numbers are treated as an import limit. In most cases, it is recommended to leave this parameter untouched.
`--user-display-name-attribute=value`	Optional	none	Specifies the name of the LDAP attribute containing the user display names.
`--group-display-name-attribute=value`	Optional	none	Specifies the name of the LDAP attribute containing the group display names.
`-Ckey=value`	Optional	none	Specifies additional JNDI environment properties to be used when connecting to the LDAP server. Note that it does not add to the previously configured custom properties; it replaces them completely. If you want to keep any of the old custom properties, make sure to specify them once again when adding new ones. This option can be specified multiple times with different keys. Example: The equivalent of specifying the `--security-authentication=DIGEST-MD5` argument is `-Cjava.naming.security.authentication=DIGEST-MD5 .` Example: Updating the context names `update-ldap-config --id="ldap1" --context-names="OU=project-x,DC=research,DC=example,DC=com|OU=phbs,DC=management,DC=example,DC=com"`
`-Rvalue`	Optional and may be specified multiple times with different values.	If this argument is not specified, the Java defaults are used.	Specifies the protocols to be used for LDAPS when connecting to the LDAP server. Example: To enable only TLSv1.2 `-RTLSv1.2`

`-Svalue`	Optional and may be specified multiple times with different values.	If this argument is not specified, the Java defaults are used.	Specifies the cipher suites to be used for LDAPS when connecting to the LDAP server.
			Example: To enable only these two cipher suites
			`-STLS_DHE_RSA_WITH_AES_128_GCM_SHA256`
			`-STLS_DHE_RSA_WITH_AES_256_GCM_SHA384`

`--connection-timeout=value`	Optional	no timeout (see description)	Specifies the connection timeout. The value must be a non-negative integer representing the timeout in milliseconds. A value less than or equal to zero results in

Option	Optional or Required	Default Value	Description
			no timeout, effectively waiting until the connection times out on TCP network level.
`--read-timeout=value`	Optional	no timeout (see description)	Specifies the read timeout. The value must be a non-negative integer representing the timeout in milliseconds. A value less than or equal to zero results in no timeout, effectively waiting until the connection times out on TCP network level.

update-site

Updates a site.

```
update-site
[-b value | --bootstrap-config=value]
[-t value | --tool-password=value]
<-s value | --site-name=value>
[-d value | --display-name=value]
[--clear-display-name]
```

Overview

Use this command to update the properties of a site. See also the create-site and set-public-address commands.

Options

Option	Optional or Required	Default Value	Description
`-b value --bootstrap-config=value`	Optional	none	The path to the bootstrap configuration file. See Bootstrap.xml file for more information about this file.
`-t value --tool-password=value`	Optional	none	The configuration tool password used to decrypt the database password in the `bootstrap.xml` file. If the tool password is omitted, the command will prompt the end user for it on the console. See Bootstrap.xml file for more information.
`-s value --site-name=value`	Required	none	The name of the site that should be updated. The list-sites command can be used to find the names of all sites.
`-d value --display-name=value`	Optional	none	The display name of the site. May help users quickly identify which server to connect to (in a environment with multiple Spotfire systems).
`--clear-display-name`	Optional	none	When this flag is specified, any existing display name will be removed. Cannot be specified together with the **`--display-name`** argument.

version

Displays the current version of the server.

```
version
```

Overview

Use this command to display the current version of the server.

Glossary

Deployments & Packages

deployment area
Deployment areas, which are set up by the Spotfire administrator, make it possible to give different users access to different versions of the Spotfire client, while still using a single Spotfire Server.

distribution
A collection of one or more software packages. The contents of a distribution are distributed to each end user's desktop using the deployment mechanism. A distribution is deployed to a deployment area.

Nodes & Services

node manager
The node manager is the networked software agent that is responsible for managing a set of services on a specific physical or virtual host. This software makes it possible to execute remote commands from the Spotfire Server.

node
All the services and instances that are run by a particular *node manager*.

service
An application that runs on a node manager and provides a particular capability; in the current version of Spotfire Server, Spotfire Web Player, Spotfire Automation Services, TERR service, and Spotfire Service for Python are the available services. A service is not available to end users until a *service instance* is running.

service instance
A specific realization of a service that is available to Spotfire end users. For example, when a user opens an analysis in the Spotfire Web Player, the user is accessing a particular instance of the Web Player service. (This distinction is invisible to the user.)

resource pool
A set of specific Spotfire Web Player *service instances* (or a single instance) that can be used in a routing rule to define where a given file, or a file requested by a specific user, should preferably open. For example, a rule can specify that company VIPs always view analyses in a particular resource pool.

Scheduling & Routing

rules
There are three types of rules: **File**, **Group**, and **User**.

The Spotfire administrator creates rules to do one of the following:

- Schedule updates to analyses (type of rule = **File**).
- Specify resource pools on which to open analyses that are requested by specific users or members of specific groups (type of rule = **User** or **Group**).
- Specify resource pools on which to open specific analyses (type of rule = **File**).

scheduled update
A rule that sets a schedule for automatically adding fresh data to an existing analysis. The rule also indicates the resource pool on which the analysis should open (Type of rule=**File**).

routing rule
A rule that specifies the resource pool on which an analysis should preferably open.

Users & Groups

users
All Spotfire users are registered in the Spotfire database. Administrators assign users to groups so that users have access to the Spotfire features that are enabled in the groups that they are members of.

group
A group is a "container" that can contain members and licenses. There are system-created groups, groups that are synchronized with an external user directory, and administrator-created groups. Groups can contain other groups as well as individual users.

primary group
The primary group is the group that determines which licenses and settings apply for a user who belongs to two or more groups. This setting is applied when a user's parent group settings are in conflict.

group member
A user or group is considered a member of the groups that they belong to explicitly, as well as the groups above these groups in the group hierarchy. In the second case, the membership is "inherited".

license
Licenses determine which features a user has access to when working in Spotfire. Administrators set licenses at the group level.

license feature
Most licenses contain a group of features that can be individually enabled and disabled. This gives administrators increased control over what users are permitted to do in Spotfire.

role
The access rights that a user acquires from the groups that the user is a member of.

Automation Services

Automation Services job
An automated procedure that carries out a multi-step task. Automation Services jobs are created in the Spotfire Job Builder, which is available in Spotfire Analyst. The jobs are saved as XML files.

scheduled Automation Services job
An Automation Services job that was scheduled, through the Automation Services area of the Spotfire administration interface, to run periodically.

Miscellaneous

information link
An information link is a structured request for data. Users can create information links to connect to external JDBC databases and thereby access and load data into Spotfire analysis files. Information links and the elements they are created from are stored in the Spotfire database.

post-authentication filter
The Spotfire Server filter that can either block all users who try to log in but are not already present in the user directory, or automatically create a new account in the user directory for any user who logs in to the server for the first time. It is also possible to use the Spotfire Server api to create a custom post-authentication filter.

preferences

Preferences are default settings for the way that people work, and the analyses they create. Preferences include a wide range of properties, from which toolbars are visible when the user starts Spotfire to the look of tables in visualizations. Administrators set preferences at the group level, using the Administration Manager in Spotfire Analyst.