

TIBCO ActiveSpaces®

Installation

*Release 2.1.2 Add-On
January 2014*

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.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIB, TIBCO, TIBCO Adapter, Predictive Business, Information Bus, The Power of Now, TIBCO ActiveMatrix BusinessEvents, and TIBCO ActiveSpaces are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

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. SEE THE README 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.

Copyright © 1999-2014 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Preface	v
Related Documentation	vi
TIBCO ActiveSpaces Documentation	vi
Typographical Conventions	vii
Connecting with TIBCO Resources	x
How to Join TIBCOCommunity	x
How to Access All TIBCO Documentation	x
How to Contact TIBCO Support	x
Chapter 1 Introduction	1
Installation Overview	2
Installation Modes	2
Installation Types	2
Installer Account	2
Installer Log File	3
Installation Requirements	4
Disk Space Requirements	4
System Memory Requirements	4
Software Requirements	5
Chapter 2 Installation Steps	7
Installing ActiveSpaces	8
Installing in GUI Mode	8
Setting Environment Variables	12
TIBCO ActiveSpaces bin and lib Directories	12
TIBCO Rendezvous bin and lib Directories	13
JRE or JDK bin and lib Directories	13
New Environment Variables Required	14
Upgrading to Release 2.1.1	15
Upgrade Procedures	15
Upgrade Paths	15
Running the as-convert Utility	17
Post-Installation Tasks	20
Enabling Processing of Large Files with Shared-Nothing Persistence	20
Testing Your ActiveSpaces Installation	22

Uninstalling ActiveSpaces 24

Installing TIBCO ActiveSpaces Monitoring and Management 27

 Required Environment Variables 27

 Running TIBCO ActiveSpaces Monitoring and Management 27

 Disabling the ASMM Space Browser 27

Index 29

Preface

TIBCO ActiveSpaces® is a distributed peer-to-peer in-memory data grid, a form of virtual shared memory that leverages a distributed hash table with configurable replication.

TIBCO ActiveSpaces® combines the features and performance of databases, caching systems, and messaging software to support large, highly volatile data sets and event-driven applications. It lets you off-load transaction-heavy systems and allows developers to concentrate on business logic rather than the complexities of developing distributed fault-tolerance.

TIBCO ActiveSpaces is available in three versions:

- **TIBCO ActiveSpaces® Enterprise Edition**—Provides C, Java, and .NET API sets and enables full cluster functionality. To enable remote clients, you must purchase licenses for the TIBCO ActiveSpaces Remote Client Edition.
- **TIBCO ActiveSpaces® Remote Client Edition**—Can be purchased in addition to the Enterprise Edition. Allows you to set up remote clients. Applications running on the remote clients can access the data grid and perform most ActiveSpaces operations.
- **TIBCO ActiveSpaces® Developer Edition**—A developer version of the product. This version is downloadable from TIBCO Developer Network at <http://tap.tibco.com>.

Topics

- [Related Documentation, page vi](#)
- [Typographical Conventions, page vii](#)

Related Documentation

This section lists documentation resources you may find useful.

TIBCO ActiveSpaces Documentation

The following documents form the TIBCO ActiveSpaces documentation set:

- *TIBCO ActiveSpaces Installation* Read this manual for instructions on site preparation and installation.
- *TIBCO ActiveSpaces Administration* Read this manual to gain an understanding of the product that you can apply to the various tasks you may undertake.
- *TIBCO ActiveSpaces Developer's Guide* Read this manual for instructions on using the product to develop an application that manages data grids.
- *TIBCO ActiveSpaces C API Reference* Read this manual for instructions on using the product to develop an application written in the C programming language that communicates with a data grid.
- *TIBCO ActiveSpaces Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.
- *Java API Reference* Read the Java API documentation for information on the Java classes and methods used in the Java API set. The Java documentation is provided online only in the `AS_HOME/docs/html` directory.
- *.NET API Reference* Read the .NET API documentation for information on the functions used in the .NET API set. The .NET documentation is provided online only in the `AS_HOME/docs/html` directory.

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>TIBCO_HOME</i> <i>ENV_HOME</i> <i>AS_HOME</i>	<p>Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as <i>TIBCO_HOME</i>. The default value of <i>TIBCO_HOME</i> depends on the operating system. For example, on Windows systems, the default value is C:\tibco.</p> <p>Other TIBCO products are installed into an <i>installation environment</i>. Incompatible products and multiple instances of the same product are installed into different installation environments. An environment home directory is referenced in the documentation as <i>ENV_HOME</i>. The default value of <i>ENV_HOME</i> depends on the operating system. For example, on Windows systems the default value is C:\tibco.</p> <p>TIBCO ActiveSpaces installs into a directory within <i>TIBCO_HOME</i>. This directory is referenced in the documentation as <i>AS_HOME</i>. The default value of <i>AS_HOME</i> depends on the operating system. For example, on Windows systems, the default value is C:\tibco\as\2.0.</p>
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use MyCommand to start the foo process.</p>
bold code font	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none"> • In procedures, to indicate what a user types. For example: Type admin. • In large code samples, to indicate the parts of the sample that are of particular interest. • In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [enable disable]

Table 1 General Typographical Conventions (Cont'd)




Convention	Use
<i>italic font</i>	<p>Italic font is used in the following ways:</p> <ul style="list-style-type: none">• To indicate a document title. For example: See <i>TIBCO ActiveMatrix BusinessWorks Concepts</i>.• To introduce new terms For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal.• To indicate a variable in a command or code syntax that you must replace. For example: <code>MyCommand <i>PathName</i></code>
Key combinations	<p>Key name separated by a plus sign indicate keys pressed simultaneously. For example: <code>Ctrl+C</code>.</p> <p>Key names separated by a comma and space indicate keys pressed one after the other. For example: <code>Esc, Ctrl+Q</code>.</p>
	<p>The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.</p>
	<p>The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.</p>
	<p>The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.</p>

Table 2 Syntax Typographical Conventions

Convention	Use
[]	<p>An optional item in a command or code syntax.</p> <p>For example:</p> <pre>MyCommand [optional_parameter] required_parameter</pre>
	<p>A logical OR that separates multiple items of which only one may be chosen.</p> <p>For example, you can select only one of the following parameters:</p> <pre>MyCommand para1 param2 param3</pre>

Table 2 Syntax Typographical Conventions

Convention	Use
{ }	<p>A logical group of items in a command. Other syntax notations may appear within each logical group.</p> <p>For example, the following command requires two parameters, which can be either the pair param1 and param2, or the pair param3 and param4.</p> <pre>MyCommand {param1 param2} {param3 param4}</pre> <p>In the next example, the command requires two parameters. The first parameter can be either param1 or param2 and the second can be either param3 or param4:</p> <pre>MyCommand {param1 param2} {param3 param4}</pre> <p>In the next example, the command can accept either two or three parameters. The first parameter must be param1. You can optionally include param2 as the second parameter. And the last parameter is either param3 or param4.</p> <pre>MyCommand param1 [param2] {param3 param4}</pre>

Connecting with TIBCO Resources

How to Join TIBCOCommunity

TIBCOCommunity is an online destination for TIBCO customers, partners, and resident experts, a place to share and access the collective experience of the TIBCO community. TIBCOCommunity offers forums, blogs, and access to a variety of resources. To register, go to <http://www.tibcommunity.com>.

How to Access All TIBCO Documentation

After you join TIBCOCommunity, you can access the documentation for all supported product versions here:

<http://docs.tibco.com/TibcoDoc>

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, please contact TIBCO Support as follows.

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

<http://www.tibco.com/services/support>

- If you already have a valid maintenance or support contract, visit this site:

<https://support.tibco.com>

Entry to this site requires a user name and password. If you do not have a user name, you can request one.

Chapter 1 **Introduction**

This chapter explains installation modes, requirements, and other options you should be aware of before starting the installation.

Topics

- [Installation Overview, page 2](#)
- [Installation Requirements, page 4](#)

Installation Overview

This section gives an overview of the installer.

Installation Modes

Three installation modes are available: GUI, console, and silent.

GUI Mode

In the GUI mode, the installer presents panels that allow you to make choices about product selection, product location, and so on. To invoke the installer in GUI mode, double-click the executable.

Console Mode

Console mode allows you to run the installer from the command prompt or terminal window. This is useful if your machine does not have a GUI environment.

Silent Mode

Silent mode installs the product using either default or custom settings that are saved in a response file. Silent mode installs the product without prompting you for information.

Installation Types

Two installation types are available: Typical or Custom.

- The Typical installation type installs all the software for a specific profile.
- The Custom installation type allows you to select components.

Installer Account

Microsoft Windows

You must have administrator privileges for the machine on which this product is installed. If you do not have administrator privileges, the installer exits. You must then log out of the system and log in as a user with the required privileges, or request your system administrator to assign the privileges to your account.

If you intend to install the product on a network drive, ensure that the account used for installation has permission to access the network drive.

UNIX

Any user can install this product.

- Regular (non-root) user
- Super-user (root).

While installing this product on UNIX platforms, ensure that the same installer account is used to install all TIBCO <Family> products.

A graphic environment such as CDE or X Windows is required to run the installer in GUI mode.

Installer Log File

The installer log file, `tibco_universal_installer.timestamp.username_install.log`, is written to the `USER_HOME/.TIBCO/install_timestamp` folder of the user's home directory. To change the location of the installer log file, specify the option `-v logFile="myLogFile"` when you run the installer.

The installer log file captures the following information:

- Installation environment details such as user that invoked the installer, host name, Java home in the environment, operating system details, and so on.
- List of assemblies installed.

Installation Requirements

This section describes the disk space requirements, system memory requirements, software requirements, and supported platforms for this product.

Disk Space Requirements

Before installing this product, extract the contents of the installation archive to a temporary directory. The installer files consume up to 1 MB of disk space.

Temporary Disk Space Required by the Installer

The installer requires at least **70 MB** of free space in the temporary directory. On Microsoft Windows, the temporary directory typically is:

`USER_HOME\Local Settings\Temp.`

If your system does not have sufficient disk space in the default temporary area, you can run the installer with a different temporary directory by using the following option when starting the installer:

`-is:tempdir /new_temp`

where `/new_temp` has sufficient free disk space.

The installer calculates the disk space required in the product home location for the selected components. The calculation is done before the actual installation (copying of files to the system) begins. The installer will proceed only if sufficient free disk space is available in the product home location.

However, if disk space is consumed by another process while the installer is copying the files, and if the required disk space is thereby reduced, the installer may fail and will then give a failure message.

While performing installation, avoid running other processes that consume disk space in the product home location.

Disk Space After Installation

This product can consume 1 GB of free space under `TIBCO_HOME`.

System Memory Requirements

System memory requirements depend on the size of the data grid that will be replicated on the device that is running TIBCO ActiveSpaces. A minimum of 100 MB of RAM is recommended.

Software Requirements

[Table 3](#) lists the required and optional software products.

Table 3 Software Requirements

Software	Description
TIBCO Software The software products below are distributed and installed separately from this product. See the readme file for the supported versions.	
TIBCO Rendezvous	<p>TIBCO Rendezvous is required if you wish to use the Rendezvous discovery transport (using a discovery URL that begins with tibrv:). Rendezvous must be installed on each machine where you install ActiveSpaces.</p> <p>If you will only make use of the embedded PGM discovery transport (using a discovery URL that begins with tibpgm:), TIBCO Rendezvous is not required.</p> <p>TIBCO ActiveSpaces Enterprise Edition includes limited license support for Rendezvous. The license is limited to use for local area communication (RVD). Separate purchase of TIBCO Rendezvous software is required for routing between multiple subnets. The TIBCO Rendezvous component included with ActiveSpaces should only be used with ActiveSpaces.</p> <p>For more information about discovery transport in ActiveSpaces, see Connecting to the Metaspace in the <i>TIBCO ActiveSpaces Developer's Guide</i>.</p>
Third-party Software See the readme file for the supported versions.	
Java JDK or JRE	<p>Installation of the Java JDK or JRE, version 1.6.0_26 or above, on a machine running ActiveSpaces enhances the functionality of ActiveSpaces as follows:</p> <ul style="list-style-type: none"> • The JDK allows use of the Java API and Administration Command Line Interface (Admin CLI). • The JRE alone will allow use of the Admin CLI and TIBCO Administrator, but not the Java API. <p>The JDK or JRE is not required on machines where the Java API or Admin CLI will not be used.</p>

Table 3 Software Requirements

Software	Description
DBMS Software	If you are implementing shared all persistence on the device where ActiveSpaces is installed, you must install a database management system of your choice and integrate it with the ActiveSpaces application that you develop.
Web Browsers	Required to run the TIBCO ActiveSpaces Monitoring and Management (ASMM) GUI.
Apache Ant	Required to compile the sample Java code.
C compiler	<p>Required to compile the sample C code.</p> <p>ActiveSpaces supports the following compilers:</p> <ul style="list-style-type: none">• Solaris, SPARC and X86 Sun Studio 12• IBM XL C, 11.01.0000.0010• HP aCC, HP C/aC++, B3910B A.06.25• Linux/OS X Latest GCC / GCC version 4.1.2 20070115• Windows cl / VS 12 <p>For the Windows platform, TIBCO ActiveSpaces provides a Makefile that works with Microsoft Visual C++.</p>

Chapter 2 **Installation Steps**



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 file for the availability of this software version on a specific operating system platform.

This chapter explains the procedures for installing TIBCO ActiveSpaces on the supported Windows and UNIX platforms.

It explains the standard TIBCO installation for the TIBCO ActiveSpaces core product and the installation of the TIBCO Administrator component.

Topics

- [Installing ActiveSpaces, page 8](#)
- [Setting Environment Variables, page 12](#)
- [Upgrading to Release 2.1.1, page 15](#)
- [Post-Installation Tasks, page 20](#)
- [Testing Your ActiveSpaces Installation, page 22](#)
- [Uninstalling ActiveSpaces, page 24](#)
- [Installing TIBCO ActiveSpaces Monitoring and Management, page 27](#)

Installing ActiveSpaces

This section describes the installation procedure for ActiveSpaces using the **TIBCO Universal Installer**.

Installing in GUI Mode

To install ActiveSpaces using GUI mode, perform the following steps.

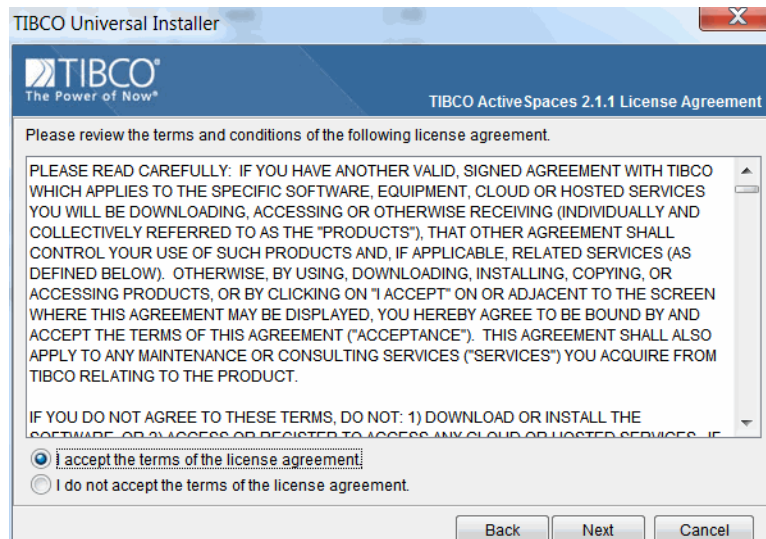
1. Open the physical media or download the ActiveSpaces product package.
2. Extract the ActiveSpaces product archive file to a temporary directory.
3. Navigate to the temporary directory that contains the universal installer.
4. Run **TIBCO Universal Installer**.

The Welcome screen appears.

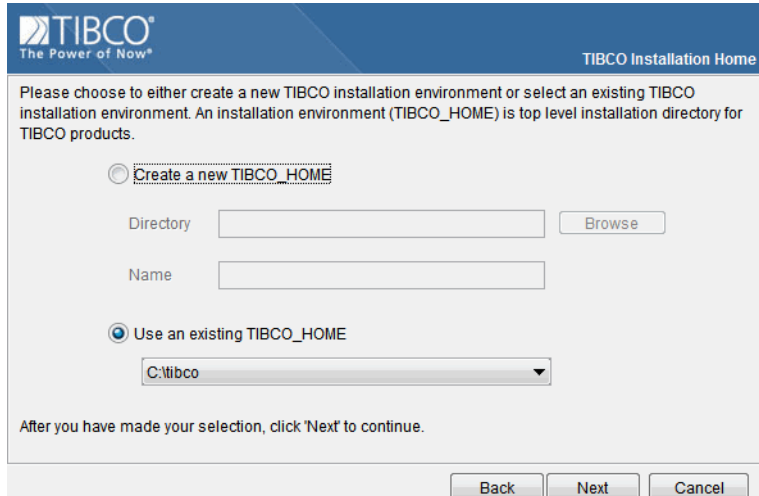
5. Click **Next**.

The License Agreement screen appears.

6. After reading through the license text, click **I accept the terms of the license agreement** and then click **Next**.



The TIBCO Installation Home window appears:



TIBCO
The Power of Now®

TIBCO Installation Home

Please choose to either create a new TIBCO installation environment or select an existing TIBCO installation environment. An installation environment (TIBCO_HOME) is top level installation directory for TIBCO products.

☐ Create a new TIBCO_HOME

Directory

Name

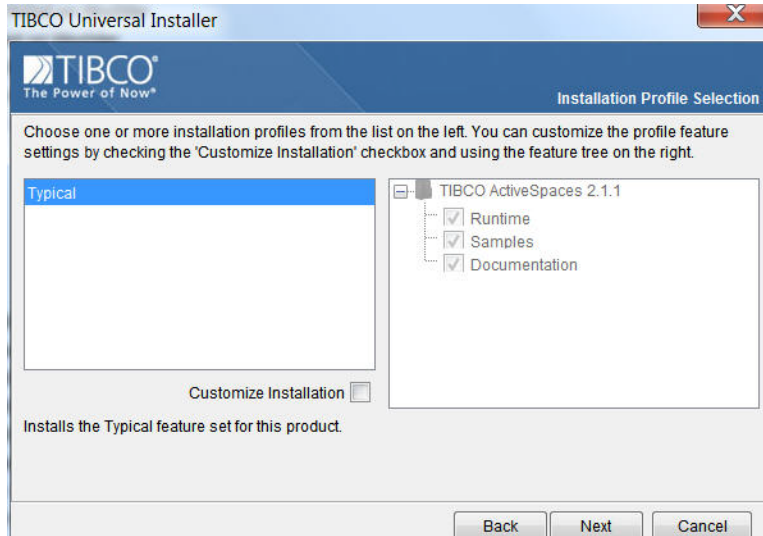
☒ Use an existing TIBCO_HOME

C:\tibco

After you have made your selection, click 'Next' to continue.

7. To use an existing TIBCO_HOME directory, leave the checkbox for **Use an existing TIBCO_HOME** checked. If you want to install in another directory, check the **Create a new TIBCO_HOME** check box and specify a new TIBCO_HOME directory.
8. Click **Next**.

The Installation Profile Selection window appears.



TIBCO Universal Installer

TIBCO
The Power of Now®

Installation Profile Selection

Choose one or more installation profiles from the list on the left. You can customize the profile feature settings by checking the 'Customize Installation' checkbox and using the feature tree on the right.

Typical

Customize Installation ☐

Installs the Typical feature set for this product.

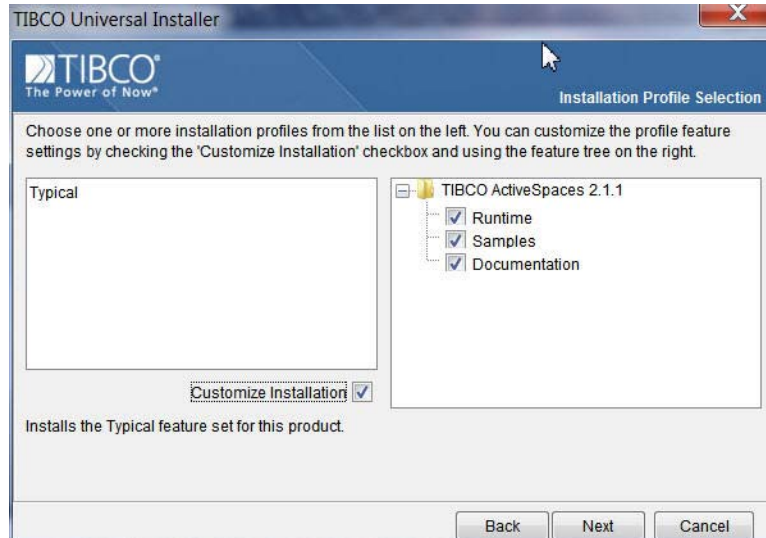
TIBCO ActiveSpaces 2.1.1

- ☒ Runtime
- ☒ Samples
- ☒ Documentation

9. To install all features, click **Typical** or to choose the features to install, check the **Customize Installation** check box.

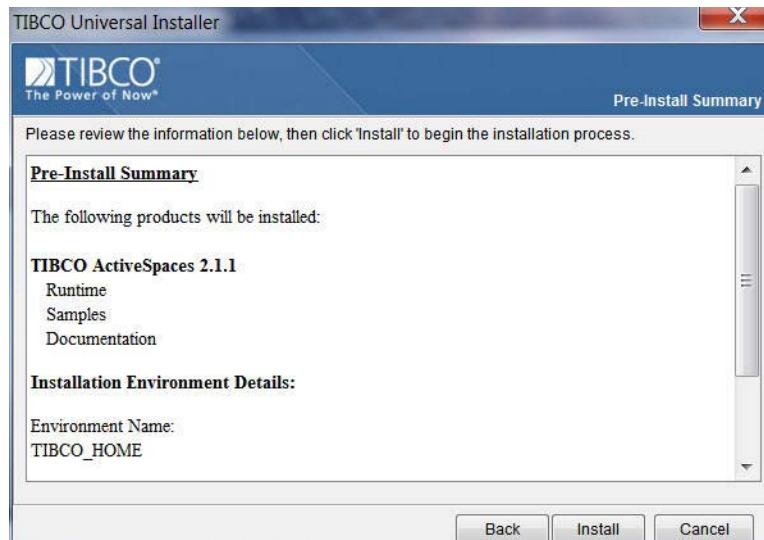
10. Click **Next**.

If you select **Customize Installation**, a screen similar to the following appears. Deselect the check box next to the features you do not want to install, and click **Next**.

11. Click **Next**.

12. If there is an existing installation, you are prompted to overwrite it. If you are sure that you want to proceed, click **Next**.

A pre-install summary screen appears.



13. Click **Install**.

A post-install report summarizes the installation results. Click **Finish** to close the installer window.

Setting Environment Variables

This section describes the environment variables that must be set after the ActiveSpaces installer is run.

TIBCO ActiveSpaces bin and lib Directories

For the operating system to find the ActiveSpaces libraries when an ActiveSpaces application is run, the following paths must be added to the PATH (Windows and UNIX) or LD_LIBRARY_PATH (UNIX) or DYLD_LIBRARY_PATH (OSX) environment variables on machines where ActiveSpaces is installed:

- Add the ActiveSpaces `bin` directory to the PATH on Windows and UNIX
- Add the ActiveSpaces `lib` directory to the PATH on Windows, or the LD_LIBRARY_PATH on UNIX, or the DYLD_LIBRARY_PATH on OSX

Windows example

If ActiveSpaces is installed in `C:\tibco\as\2.1`:

Add `C:\tibco\as\2.1\lib` and `C:\tibco\as\2.1\bin` to the PATH environment variable.

The following Windows command script sets the required environment variables for starting the Admin CLI in a command window.

Example 1 Windows Command File for Setting Environment Variables for the Admin CLI

```
@echo off

set AS_HOME=c:\tibco\as\2.1
set PATH=%AS_HOME%\lib;%AS_HOME%\bin;%PATH%

set JAVA_HOME=%DRIVE_C%\Program Files\Java\jdk1.6.0_24
set PATH=%PATH%;%JAVA_HOME%\bin;%JAVA_HOME%\lib
```



Make sure that you prepend the `AS_HOME\lib` PATH setting and the `AS_HOME\bin` PATH setting to the PATH setting; for example:

```
set PATH=%AS_HOME%\lib;%AS_HOME%\bin;%PATH%
```

UNIX example

If ActiveSpaces is installed in `/usr/tibco/as/2.1`:

Add `/usr/tibco/as/2.1/lib` to the `LD_LIBRARY_PATH` and `/usr/tibco/as/2.1/bin` to the `PATH` environment variable.

OSX example

If ActiveSpaces is installed in `/usr/tibco/as/2.1`:

Add `/usr/tibco/as/2.1/lib` to the `DYLD_LIBRARY_PATH` and `/usr/tibco/as/2.1/bin` to the `PATH` environment variable.

TIBCO Rendezvous bin and lib Directories

If you want to use TIBCO Rendezvous as the transport for ActiveSpaces, Rendezvous must be installed on each machine running ActiveSpaces, and these paths must be added to the `PATH` (Windows) or `LD_LIBRARY_PATH` (UNIX). For example, add:

- The Rendezvous `bin` directory: to the `PATH` on Windows and UNIX
- The Rendezvous `lib` directory: to the `PATH` on Windows, or the `LD_LIBRARY_PATH` on UNIX

Windows example

If TIBCO Rendezvous is installed in `C:\tibco\tibrv`:

Add `C:\tibco\tibrv\lib` and `C:\tibco\tibrv\bin` to the `PATH` environment variable

UNIX example

If TIBCO Rendezvous is installed in `/usr/tibco/tibrv`:

Add `/usr/tibco/tibrv/lib` to the `LD_LIBRARY_PATH` and `/usr/tibco/tibrv/bin` to the `PATH` environment variable

JRE or JDK bin and lib Directories

In order for the operating system to find the Java class libraries when an ActiveSpaces application is run, the `bin` and `lib` directories for the JDK or JRE must be included in the `PATH` environment variable, for example, `C:\Program Files\Java\jre6\bin` and `C:\Program Files\Java\jre6\lib`.

New Environment Variables Required

CLASSPATH environment variable

In order for Java ActiveSpaces applications to run, the CLASSPATH variable should include the following JAR files:

- `as-common.jar`

All applications require this JAR file to be in the CLASSPATH.

- `as-admin.jar` and `antlr-3.2.jar`

All applications making use of the Admin object require these JAR files to be in the CLASSPATH.

AS_HOME environment variable

Create a System environment variable called `AS_HOME` that points to the directory where ActiveSpaces is installed, for instance, `C:\tibco\as\2.1` on Windows or `/usr/tibco/as/2.1` on UNIX.

JAVA_HOME environment variable

The `JAVA_HOME` environment variable must be created if it does not already exist. It should point to the top level directory of the Java JDK or JRE, for example, `C:\Program Files\Java\jre6`.

Upgrading to Release 2.1.1

Upgrade Procedures

Two upgrade methods are available for TIBCO ActiveSpaces:

- **Rolling Upgrade** You can use this method when upgrading from a minor release to a major release. With a rolling upgrade, you bring one node down and upgrade it, restart it, and then upgrade additional nodes, one node at a time.



The rolling upgrade feature is available only for upgrade between minor releases; for example, upgrading from one service pack to another service pack release.



You cannot use the rolling upgrade method for nodes that are running through as-agent processes. For these nodes, you must use the offline upgrade method.

- **Offline Upgrade** You perform an offline upgrade by shutting down each instance of ActiveSpaces, upgrading the node to ActiveSpaces 2.1.1, and then restarting the node. You can use this method for any of the upgrade paths to ActiveSpaces 2.1.1.

If you are upgrading from TIBCO ActiveSpaces Release 2.0.0 to Release 2.0.x or from release 2.0.1 to release 2.0.2, and you have implemented shared-nothing persistence, then after upgrading to the new release version, you must convert any shared-nothing persistence data files from your 2.0.0 release to the newer format that is used with Release 2.0.2.

An off-line conversion utility, `as-convert`, is provided to perform this data conversion. The utility automatically detects the release that created the data files and converts them to the release 2.1.1 format.

For information on the `as-convert` utility, see [Running the as-convert Utility, page 17](#).

Upgrade Paths

The following upgrade paths are available for ActiveSpaces 2.1.1:

- **Release 2.0 to 2.1.1** Requires an offline upgrade. Convert any shared-nothing persistence data files to the ActiveSpaces 2.1.1 format by running the `as-convert` utility.

- **Release 2.0.1 to Release 2.1.1** Requires an offline upgrade. Convert any shared-nothing persistence data files to the ActiveSpaces 2.1.1 format by running the `as-convert` utility.
- **Release 2.0.1 to 2.0.2** Requires an offline upgrade. Convert any shared-nothing persistence data files to the ActiveSpaces 2.1.1 format by running the `as-convert` utility.
- **Release 2.0.2 to Release 2.1.1** Requires an offline upgrade. You do not need to convert shared-nothing persistence data files for this upgrade.
- **Release 2.1.1 to Future 2.1.x Releases** Depending on your situation, you can use the rolling upgrade method or the offline upgrade method.

Performing a Rolling Upgrade

To perform a rolling upgrade:

1. Run the TIBCO Universal Installer and install the new version of ActiveSpaces into a new `TIBCO_HOME` directory.



If there is no previous installation on this machine, you do not need to use a different `TIBCO_HOME` directory.

2. Start one instance (for example, an `as-agent`) from the new `TIBCO_HOME` directory, and wait for redistribution to complete.
3. Then, bring down gracefully (using the `metaspace.closeAll` method) a node that was started from the old `TIBCO_HOME` directory or kill the node (using `CTRL-c` or `kill -9`), and wait for redistribution to complete.
4. Repeat steps 2-3 until all nodes from the old `TIBCO_HOME` are removed from the cluster.
5. Repeat steps 1-4 on all hosts.



Review the Release Notes the new TIBCO ActiveSpaces version for changes to the API that might affect your applications.

You will need to upgrade your applications to use the new API calls.

Performing an Offline Upgrade

On each AS node:

1. Stop all ActiveSpaces processes.
2. If you have implemented shared-nothing persistence, run the `as-convert` utility to convert shared-nothing persistence data files to the release 2.1.1 format.

See [Running the as-convert Utility, page 17](#).

3. Run the TIBCO Universal Installer and install AS 2.1.1.

See [Installing ActiveSpaces, page 8](#).

4. Restart the ActiveSpaces processes.



Review the Release Notes for the new TIBCO ActiveSpaces version for changes to the API that might affect your applications.

You will need to upgrade your applications to use the new API calls.

Running the as-convert Utility

The `as-convert.exe` file is located in the following directory:

`AS_HOME/bin`



Before you run `as-convert`, stop all `as-agents` and `seeders`. The **`as-convert`** utility must be run off-line.

Command Syntax

```
as-convert -data_store <directory_path> -metaspace <metaspace_name>  
-space <space_name> -name <membername> -file <file_name> -compact  
-dry_run -verbose -log <log_file> -debug <log_level>
```

[Table 4](#) describes the parameters for `as-convert`.

Table 4 Parameters for as-convert

Parameter	Usage
<code>-data_store</code>	<i>directory_path</i> specifies the path to the data store to convert.

Table 4 Parameters for *as-convert*

Parameter	Usage
-metaspace	To specify conversion of the data files for all of the spaces defined for a metaspace, specify a metaspace name.
-space	To specify conversion of the data files for a specific space within the metaspace, specify the space name with the -space parameter and the metaspace name with the -metaspace parameter.
-name	To specify conversion of the data files for a specific space member, specify the member name with the -name parameter, the space name with the -space parameter, and the metaspace name with the -metaspace parameter.
-file	To specify conversion of a specific file, specify the filename with the -file parameter, the data store path with the -data_store parameter, and the metaspace name with the -metaspace parameter.
-compact	Deletes any white spaces in converted file. White spaces can be added to the data file as a result of processing Takes, which delete data.
-dry_run	To run the utility without actually converting the data store and output informational messages, include the -dry_run parameter
-help	Outputs a summary of the command syntax for as-convert .
-verbose	Causes output of more information.
-log	Specifies the name of a log file to which to write log information
-debug	Specifies the log level for messages output by the utility.

Usage Notes

The *as-convert* utility converts ActiveSpaces shared-nothing files from one format (or one version) to another (usually higher). The utility does the following:

- If a file name is provided, processes that file and prints out the result.
- If a space name is provided along with the member name and metaspace name, converts all files that belong to that space.
- If a member name is provided along with the space name, converts files for that member only.
- If a metaspace name is provided, the utility does the above for each space that is part of the metaspace.
- If no argument is provided, processes the entire data store — reads each subdirectory and converts all files.
- the `dry_run` option will just touch the files and identify which are older than current version. This option is good for estimating how many files need conversion.

Post-Installation Tasks

This section describes tasks that you might have to perform after completing the TIBCO ActiveSpaces installation.

Enabling Processing of Large Files with Shared-Nothing Persistence

This section discusses considerations for handling large files when using shared-nothing persistence.

Setting Resource Limits (ulimit) for the Current Shell

On some systems, the administrator might have set small `ulimit` values for resources that affect execution of ActiveSpaces processes, such as data segment size, stack size, and virtual memory. It is advisable to set these values to `unlimited` and let processes receive maximum available memory from the available resources.

You can use the following commands to change resource values to `unlimited`:

```
ulimit -d unlimited  
ulimit -s unlimited  
ulimit -v unlimited
```

Procedure for Enabling Processing of Large Files

For shared-nothing persistence, ActiveSpaces is enabled to work with files larger than 2 GB.

If you are implementing shared-nothing persistence on a UNIX or Linux platform, complete these steps before starting seeders:

1. At the shell command prompt, enter the following:
ulimit -a
2. If the `ulimit` command does not return `unlimited`, enter the following to set the user resources (including file length) to `unlimited`:
ulimit -f unlimited

3. Make sure that the file system where ActiveSpaces will write shared-nothing persistence files is enabled for large files.
 - a. To set the directory to which shared-nothing persistence files are written, do one of the following:
 - The default directory is the home directory of the user running the ActiveSpaces application. If that is where you want the shared nothing files written, you do not need to change the directory.
 - Set the *AS_DATA_STORE* environment variable to point to the directory where the shared-nothing persistence files are to be stored.
 - Run *as-agent* with the *-data_store* parameter to specify the data store directory:
as-agent -data_store <path>
where *path* specifies the directory path for the data store directory.
 - b. Consult with your system administrator on how to verify and enable the file system for large files.

Testing Your ActiveSpaces Installation

When the installation is completed and the required environment variables are set, you can test your ActiveSpaces installation to ensure that it is functioning properly. A simple way of testing the installation is described here.

- 1. In a command prompt or shell window launch the Admin Command Line Interface (CLI) by navigating to the ActiveSpaces lib directory, then invoking `java -jar as-admin.jar`.
- 2. At the Admin CLI prompt type `connect`.
- 3. When connected, type `show members`. The output of this command should be the ID of a single member of the default metaspace, `ms` (that instance of the Admin CLI).

Example:

```
C:\Documents and Settings\username>cd C:\tibco\as\2.1\lib

C:\tibco\as\2.1\lib>java -jar as-admin.jar
as-admin> connect

ip_address=101.98.201.47 port=50000
member joined: 101.98.201.47:50000
ConnConnected to metaspace name [ms] discovery [tibpgm] listen
[tcp://101.98.201.47:50000]e
cas-admin> ted metaspace name=[ms],
listen=[tcp://101.98.201.47:50000], discovery=[tibpgm],
name=[a62c92f-c350]

as-admin> show members

Cluster members:
-----
Member id                Member role                | Member name
-----
a62c92f-c350-4f10dc89-3b7  manager                    | a62c92f-c350

Remote members:
No remote members
as-admin>
```

- 4. In a second command prompt (or shell) window, launch another instance of the Admin CLI,
- 5. Type `connect` again.
- 6. Once connected, type `show members` again. The output should now show the IDs of *two* members (both instances of the Admin CLI).

Example:

```
C:\Documents and Settings\username>cd C:\tibco\as\2.1\lib
```

```
C:\tibco\as\2.1\lib>java -jar as-admin.jar
as-admin> connect
```

```
ip_address=10.98.201.47 port=50001
Connected to metaspace name [ms] discovery [tibpgm] listen
[tcp://10.98.201.47:50001]
as-admin> ted metaspace name=[ms],
listen=[tcp://10.98.201.47:50001], discovery=[tibpgm],
name=[a62c92f-c351]
member joined: 10.98.201.47:50001
member joined: 10.98.201.47:50000
as-admin> show members
```

```
Cluster members:
```

```
-----
Member id | Member role | Member name
-----
a62c92f-c350-4f10dc89-3b7 | manager | a62c92f-c350
a62c92f-c351-4f10df44-1b | member | a62c92f-c351
```

```
Remote members:
No remote members
as-admin>
```

The output of the **show members** command will also indicate for each process whether it is the **MANAGER** or is simply a **MEMBER** of the metaspace.

For more information about the Admin CLI and the commands **connect** and **show members**, see the *TIBCO ActiveSpaces Administration* document. You can also type **?** at any time to see a list of Admin CLI commands and their parameters.

Uninstalling ActiveSpaces

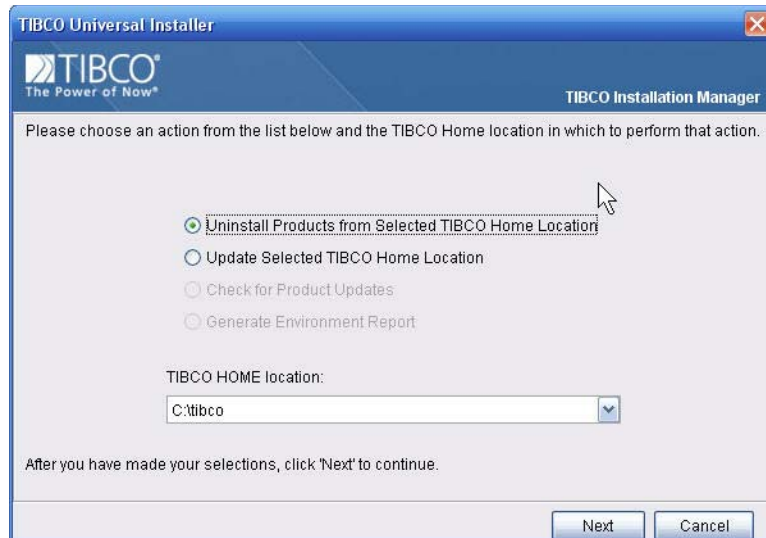
Use GUI mode or Console mode to uninstall the software.

GUI Mode

To uninstall this product:

1. Shut down all running TIBCO <Family> applications.
2. Navigate to *TIBCO_HOME/tools/universal_installer* and run *TIBCOUniversalInstaller*.

The TIBCO Installation Manager screen appears.



3. On the TIBCO Installation Manager page, perform the following steps:
 - Select the **Uninstall Products From Selected TIBCO Home Location** radio button.
 - Select the *TIBCO_HOME* location from the TIBCO Home Location drop-down list.

The Welcome dialog appears

4. Click the **Next** button.

5. Choose an uninstallation option. The wizard provides two uninstallation options:
 - **Custom Uninstall** You can select the products to be removed.
 - **Typical Uninstall** The universal uninstaller removes all the products in this *TIBCO_HOME*.
6. Click the **Next** button.
7. If you selected the **Custom Uninstall (Select The Products To Be Removed)** radio button, select the check boxes for products to uninstall, and then click the **Uninstall** button.
8. Review the Pre-Uninstall Summary and click the **Uninstall** button to start the uninstallation process.
9. Review the Post-Uninstall Summary and click the **Finish** button to exit the uninstall wizard.
10. Click the **Uninstall** button to remove the selected products.
11. To complete the procedure, clean up the environment variables that you modified after the ActiveSpaces installation.

Console Mode

To uninstall this product in Console mode, complete the following steps:

1. Using a command window, navigate to the *TIBCO_HOME\tools\universal_installer* directory.
2. Type the following command at the command prompt:
TIBCOUniversalInstaller.exe -console
3. Complete the installation by responding to the console window prompts.

Uninstalling on Solaris SPARC Systems

When uninstalling TIBCO ActiveSpaces on Solaris SPARC systems, you might see an error message indicating that the uninstall failed due to a locked file condition for the *as-common.jar* file.

To avoid this problem:

1. Enter the following command:
cd \$TIBCO_HOME/ tools/universal_installer \$
2. Start the TIBCO Universal Uninstaller as follows:

```
$TIBCO_HOME/ tools/universal_installer  
$./TIBCOUniversalInstaller-sol-sparc.bin -V  
disableLockChecks="true"
```

Installing TIBCO ActiveSpaces Monitoring and Management

TIBCO ActiveSpaces Monitoring and Management (ASMM) is a GUI that allows you to perform many of the functions provided by the ActiveSpaces admin CLI, and which also allows you to view statistics about space members and browse space data.

The ASMM software is automatically installed when you install TIBCO ActiveSpaces. By default, ASMM is installed in the following directory:

`C:\tibco\as\2.1\asmm`

Required Environment Variables

TIBCO ASMM requires setting the path to the Java lib directory.

Running TIBCO ActiveSpaces Monitoring and Management

For information on starting and using TIBCO ActiveSpaces Monitoring and Management, see the *TIBCO ActiveSpaces Administration* document.

Disabling the ASMM Space Browser

By default, the Utilities menu and the space browser are enabled for ASMM. If you want to disable the ASMM Utilities menu and the space browser, do the following.

1. On the device where ActiveSpaces is installed, navigate to the following directory:

```
%AS_HOME%\2.1\asmm\war>
```

2. Locate the `clientconfig.xml` file.

The `clientconfig.xml` file contains the following XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<asmmconfig>
  <!-- How much time (ms) between client update requests -->
  <ClientUpdateInterval>5000</ClientUpdateInterval>

  <!-- Set of Metaspaces to connect to upon opening
ActiveSpaces Monitoring and Management console -->
  <StartupMetaspaces>
  </StartupMetaspaces>

  <!-- Enable/Disable space browser utilities -->
  <EnableSpaceBrowser>true</EnableSpaceBrowser>
```

3. Edit the file and locate the line that reads:

```
<EnableSpaceBrowser>true</EnableSpaceBrowser>
```

4. Change true to false.

```
<EnableSpaceBrowser>false</EnableSpaceBrowser>
```

5. Save the file.

6. Start ASMM.

The Utilities menu and the space browser are now disabled.

Index

A

AS_HOME environment variable [14](#)
as-convert [15](#)

C

CLASSPATH environment variable [14](#)
Console Mode [2](#), [25](#)
customer support [x](#)

D

Disk Space Requirements [4](#)

G

GUI Mode [2](#)

I

Installation Modes
 Console Mode [2](#)
 GUI Mode [2](#)
 Silent Mode [2](#)
Installation Requirements
 Disk Space Requirements [4](#)
 Software Requirements [5](#)
 System Memory Requirements [4](#)
Installation Type
 Custom [2](#)

Installer Account [2](#)
Installer Log File [3](#)
installing ActiveSpaces [8](#)
installing in GUI mode [8](#)

J

JAVA_HOME environment variable [14](#)
JRE or JDK bin and lib directories [13](#)

N

new environment variables required [14](#)

O

optional supporting software [8](#)

S

setting environment variables [12](#)
Silent Mode [2](#)
Software Requirements [5](#)
support, contacting [x](#)
supported platforms [5](#)
System Memory Requirements [4](#)

T

- technical support [x](#)
- testing your installation [22](#)
- TIBCO ActiveSpaces bin and lib directories [12](#)
- TIBCO Rendezvous [5](#)
- TIBCO Rendezvous bin and lib directories [13](#)
- TIBCO_HOME [vii](#)

U

- Uninstallation
 - Console Mode [25](#)
- uninstalling ActiveSpaces [24](#)
- upgrading from Release 2.0.1 or 2..0.x to Release 2.1.1 [15](#)