TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM

User's Guide

Software Release 1.4
June 2016

Two-Second Advantage®

Important Information

Contents

Figures

Preface

Topics

Related Documentation

This section lists useful documentation resources.

TIBCO Silver Fabric Enabler for ActiveMatrix BPM Documentation

The following documents form the TIBCO Silver Fabric Enabler for ActiveMatrix BPM documentation set:

- *TIBCO Silver Fabric Enabler for ActiveMatrix BPM Installation*
  Read this manual for instructions on site preparation and installation.
- *TIBCO Silver Fabric Enabler for ActiveMatrix BPM User's Guide*
  Read this manual for instructions on using the product.
- *TIBCO Silver Fabric Enabler for ActiveMatrix BPM 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.

Other TIBCO Product Documentation

TIBCO Silver Fabric Enabler for ActiveMatrix BPM uses TIBCO Silver® Fabric private cloud infrastructure to configure and install a TIBCO ActiveMatrix® BPM environment on the cloud.

It is essential that you read documentation for the following TIBCO products:

- TIBCO Silver® Fabric
- TIBCO ActiveMatrix® BPM
- TIBCO Enterprise Message Service™
- TIBCO Business Studio™

Your business applications may leverage services provided by other TIBCO products as well.

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
TIBCO_HOME *ENV_HOME* *SFBP_HOME* *SILVERFABRIC_HOME*	Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as *TIBCO_HOME*. The default value of *TIBCO_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco`. Other TIBCO products are installed into an *installation environment*. Incompatible products and multiple instances of the same product are installed into different installation environments. An environment home directory is referenced in documentation as *ENV_HOME*. The default value of *ENV_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco`. TIBCO Silver Fabric Enabler for ActiveMatrix BPM is installed into a directory that is referenced in documentation as *SFBP_HOME*. The value of *SFBP_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco\sfbp`. TIBCO Silver Fabric is installed into a directory that is referenced in documentation as *SILVERFABRIC_HOME*. The value of *SILVERFABRIC_HOME* depends on the operating system. For example, on Windows systems, the default value can be `C:\fabric`.
`code font`	Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example: Use `MyCommand` to start the foo process.
`bold code font`	Bold code font is used in the following ways: • 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
italic font	Italic font is used in the following ways: • To indicate a document title. For example: See *TIBCO ActiveMatrix BPM Concepts*. • To introduce new terms. For example: A portal page may contain several portlets. *Portlets* are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: `MyCommand` *PathName*
Key combinations	Key names separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C. Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	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.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

Connecting with TIBCO Resources

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts, a place to share and access the collective experience of the TIBCO community. TIBCOmmunity 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 TIBCOmmunity, you can access the documentation for all supported product versions here:

http://docs.tibco.com

It is especially important to gather expertise from the TIBCO ActiveMatrix BPM documentation to get the most from your TIBCO ActiveMatrix BPM installation on TIBCO Silver Fabric.

All TIBCO product documentation is available at https://docs.tibco.com

TIBCO Silver Fabric Enabler for ActiveMatrix BPM documentation is here:

https://docs.tibco.com/products/tibco-silver-fabric-enabler-for-activematrix-bpm-1-4-0

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 introduces TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM within the context of TIBCO Silver® Fabric and TIBCO ActiveMatrix® BPM.

Topics

Product Overview

TIBCO Silver Fabric Enabler for ActiveMatrix BPM is a complementary software to use with TIBCO Silver® Fabric. It simplifies configuration and publishing of TIBCO ActiveMatrix BPM Server, TIBCO ActiveMatrix BPM Runtime Nodes, and associated dependencies to run as an integrated ActiveMatrix BPM system environment on the TIBCO Silver Fabric Cloud.

The software includes two Silver Fabric BPM Component wizards that guide you through the configuration pages necessary to create new ActiveMatrix BPM Server and ActiveMatrix BPM Runtime components. Create Silver Fabric ActiveMatrix BPM Stacks with those components and their component dependencies to get a complete TIBCO ActiveMatrix® BPM environment on the cloud.

To build and run a TIBCO ActiveMatrix BPM environment complete these tasks:

- Prerequisite: Create a supported database and a supported messaging server to provide support for the ActiveMatrix BPM environment. These can be components started as dependencies within the ActiveMatrix BPM Stack or they can be stand-alone servers loosely coupled by JDBC drivers and specified connection protocols.
- Create and publish a TIBCO ActiveMatrix BPM Server component. Refer to Creating a TIBCO ActiveMatrix BPM Server Component on page 7.
- Create and publish a TIBCO ActiveMatrix BPM Runtime component. Refer to Creating a TIBCO ActiveMatrix BPM Runtime Component on page 48.
- Using the previously mentioned components, define a TIBCO Silver Fabric Stack with dependencies on the Database component and EMS component (if applicable) to create an ActiveMatrix BPM environment.

Main Functionalities

TIBCO Silver Fabric Enabler for ActiveMatrix BPM provides the following main functionalities:

- You can quickly set up and install TIBCO ActiveMatrix BPM Administrator and TIBCO ActiveMatrix Runtime environments on TIBCO Silver Fabric machines.
- TIBCO ActiveMatrix BPM Administrator can publish BPM projects onto the ActiveMatrix BPM Runtime environment using traditional tools, such as TIBCO ActiveMatrix BPM Administrator user interface or its command-line tools.

- It provides a simplified configuration interface for publishing a TIBCO ActiveMatrix BPM Administrator or Runtime component instance to the cloud.

Components

TIBCO Silver Fabric Enabler for ActiveMatrix BPM facilitates creation and configuration of the following components:

- TIBCO ActiveMatrix BPM server component

  Configure and publish an instance of a TIBCO ActiveMatrix BPM server component.

- TIBCO ActiveMatrix BPM Runtime Component

  The TIBCO ActiveMatrix BPM Runtime Component is configured to publish and start an instance of the TIBCO Host.

Distribution

The TIBCO ActiveMatrix® BPM Distribution for TIBCO Silver® Fabric is the expanded installation of TIBCO ActiveMatrix® BPM. It is downloaded separately from the product download area and it is published by the Enabler to the Silver Fabric Engines by the components with their respective configurations.

Stacks

The Stack is a set of components used to quickly create an ActiveMatrix BPM enterprise environment. An ActiveMatrix BPM Stack can have two component types: one ActiveMatrix BPM Server Component and one or more ActiveMatrix BPM Runtime Components configured to be published as any of the various BPM node types.

The Stack has one-to-one mapping to a single ActiveMatrix BPM enterprise environment.

Chapter 2 Creating a TIBCO Silver Fabric Enabler for ActiveMatrix BPM Stack

This chapter explains how to configure and publish TIBCO Silver Fabric Enabler for ActiveMatrix BPM Stack.

Topics

Prerequisites

Configuration of an ActiveMatrix BPM component requires specific knowledge about external dependencies. In most cases you also need files like a JDBC driver, certificates, and trust stores to create secure connections with the databases.

Component configurations depend on environment implementation details to establish connections with external services hosted somewhere beyond the local host. Using external services can be optional for development systems, but most production systems require a more complex architecture with externally hosted services and dependencies.

In particular, the database and the messaging service must be in place or ready to be deployed just prior to publishing and deploying the rest of your BPM Stack because the database will be initialized with a BPM specific schema when ActiveMatrix BPM Server is published and run on TIBCO Silver Fabric.

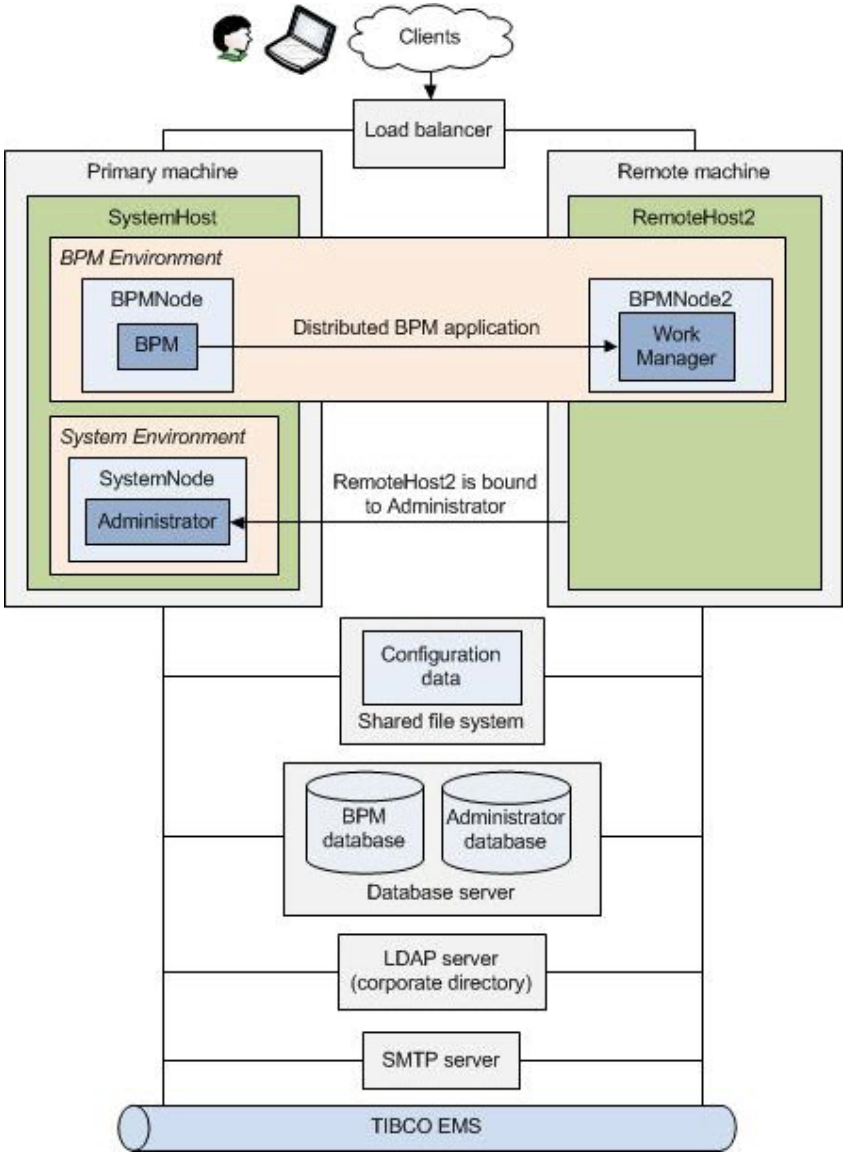
Additionally, you might have several services that you might plan on hosting separately from TIBCO ActiveMatrix BPM Server. If the following services are going to be hosted externally with respect to TIBCO ActiveMatrix BPM, you should install and run them, prior to component configuration so you can specify the URL and connection profile values:

- External EMS for ActiveMatrix Administrator
- External Authentication Database Server or external LDAP Server
- External Monitoring Service Database
- External Log Service Database
- Monitoring Service connected with an external EMS
- ActiveMatrix Administrator Log Service
- ActiveMatrix Administrator Payload Service
- External CMIS Repository

In many cases, the native capabilities of TIBCO ActiveMatrix BPM support these services for development cases making configuration much easier. Production environment requirements and performance loads might make these optional configurations necessary.

The TIBCO Silver Fabric Enabler for ActiveMatrix BPM explicitly supports a distributed ActiveMatrix BPM System. TIBCO Silver Fabric features a virtual router that supports load balancing. Distributed application architecture is supported by as many BPM run time nodes as can be hosted on Silver Fabric engines. Distributed, horizontally scalable architecture of TIBCO ActiveMatrix BPM is illustrated below.

Figure 1 TIBCO ActiveMatrix BPM - distributed architecture for horizontal scalability.



You can have more help to decide on optimizing your environment from the TIBCO ActiveMatrix BPM documentation on docs.tibco.com.

Creating a TIBCO ActiveMatrix BPM Server Component

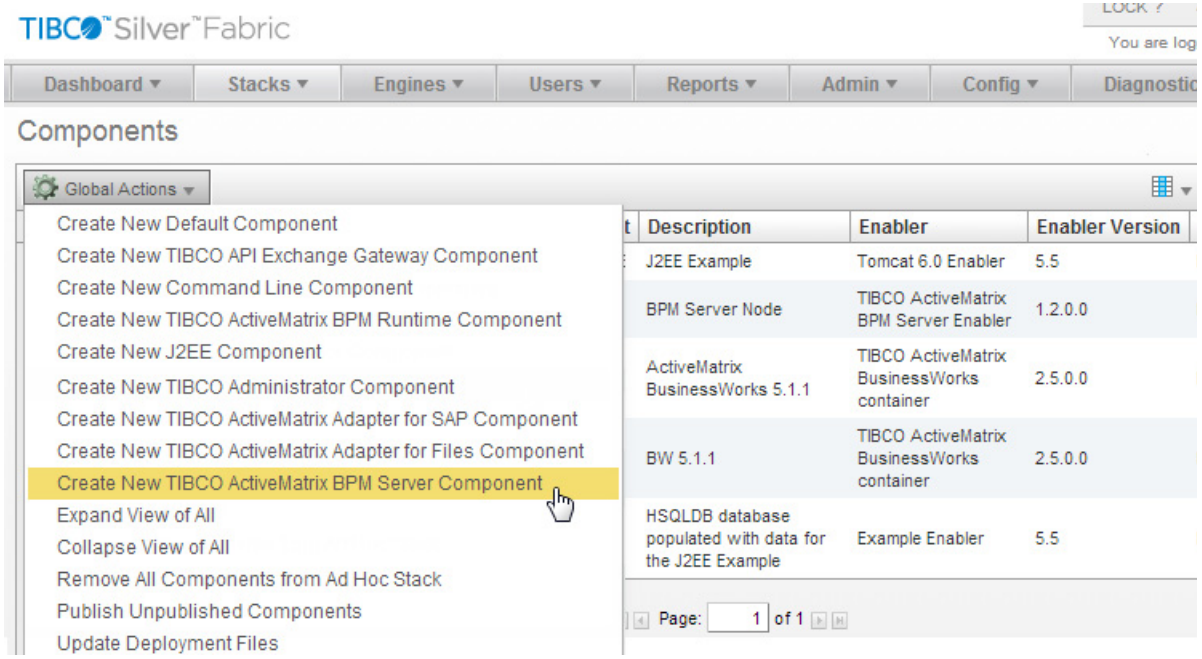
To create and configure a TIBCO ActiveMatrix® BPM Server Component, also known as an ActiveMatrix BPM Server Component or simply the component, use the TIBCO Silver Fabric Component Wizard described in the following pages.

Create and Finish a TIBCO ActiveMatrix BPM Server Component

A TIBCO ActiveMatrix BPM server component can create different run mode configurations to publish and deploy an ActiveMatrix Administrator, a BPM Server, or both. Regardless of which you would like to configure, you must create it and finish it in the same way.

1. Using TIBCO Silver Fabric Administrator, select **Stacks > Components**.
2. On the Components page, select **Create New TIBCO ActiveMatrix BPM Server Component** from the **Global Actions** list depicted here in Figure 2.

Figure 2 Creating a new TIBCO ActiveMatrix BPM Server Component



3. Provide a unique name and a description (optional) for your new component.

Figure 3 Configuring General Properties

4. Choose the TIBCO ActiveMatrix® BPM product distribution version.

   Choose the LGPL Hibernate distribution that was downloaded and installed.

   You must select the latest versions of any installed distributions shown to benefit from the latest fixes and features.

5. Click **Next** and step through the pages of the Component Wizard making configuration settings appropriate for your implementation. You must step through the Component Wizard until the **Finish** button is active.

   Many other Component Wizard pages and settings are required to complete creation of this enabler, and some are optional. Those pages are described in separate topics.

6. Click **Finish** to save your component settings. Exiting the Component Wizard prior to clicking **Finish** results in loss of any settings made.

If you come to a point where you must step away from the Component Wizard for an extended length of time (>60 minutes - default maximum), you should proceed until you can click **Finish**. Click **Finish** to save changes you have made.

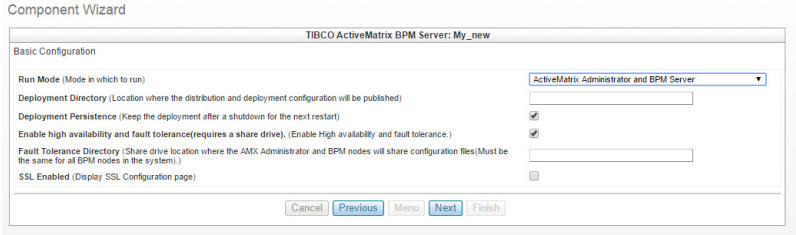
If TIBCO Silver Fabric Administrator times out, then any settings or configuration changes you have made are lost if you have not clicked **Finish**.

Set the ActiveMatrix BPM Server Run Mode - Basic Configuration Page

A TIBCO ActiveMatrix BPM server component can create different run mode configurations to publish and deploy an ActiveMatrix Administrator, a BPM Server, or both. Use the Basic Configuration page to set the Deployment Directory, Deployment Persistence, and to display the SSL Configuration page.

1. **Run Mode** - On the Component Wizard > Basic Configuration page, use the pull-down menu to select a Run Mode:
   - ActiveMatrix Administrator and BPM Server
   - ActiveMatrix Administrator Only
   - BPM Server Only

Figure 4 Basic Configuration



The Component Wizard displays pages according to your run mode selection.

2. **Deployment Directory** - Enter an absolute path to specify where ActiveMatrix BPM is published on the TIBCO Silver Fabric engine. When you specify either a Windows or a UNIX-based absolute path, you might want to define a resource preference in the stack for an OS platform that matches your component deployment directory setting. If that directory is not present, it is created during the deployment initialization.

If you use a directory on a shared drive, then the component instance can be more quickly migrated or run on different engine instances without having to go through the full re-initialization of the node (which takes longer).

If no Deployment Directory is specified, the default is:
ENGINE_HOME`/work/[`*`component_name`*`]`

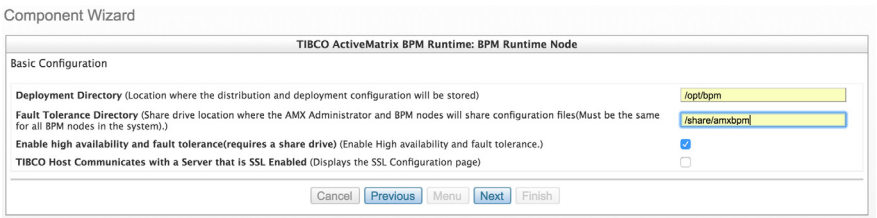
To prevent any operating system path length limitations from becoming an issue, use a **Deployment Directory** with as few characters as possible. For example, you could use "`C:\AMX`" instead of "`C:\ActiveMatrix`" or for a Linux based engine "`/dd`" will create the deployment directory at the root `/`.

Fault Tolerance (FT) Directory: is the location of the shared drive, where the configuration files that are shared by the replicated ActiveMatrix Administrator or BPM nodes are placed. The location must be same for all BPM nodes in the system. This field gets enabled if you select the **Enable high availability and fault tolerance** checkbox**.** This field is mandatory if you enable high availability and fault tolerance feature.

The two scenarios are:

- When using **ActiveMatrix Administrator Only** mode, after selecting the **Replicate AMX Administrator Instance** checkbox, the **Fault Tolerance Directory** field is displayed. You need to specify a shared drive to store the shared information between primary and replicated administrators. The **ActiveMatrix Administrator Only** component with **Replicate AMX Administrator Instance** enabled, requires the component to be instantiated twice (Minimum or Maximum engines should be 2). For more information, refer to the section `Configuring a Stack to Replicate an ActiveMatrix Administrator`.
- When using **ActiveMatrix Administrator and BPM Server** or **BPM Server Only** modes, after selecting the **Enable high availability and fault tolerance** checkbox, the **Fault Tolerance Directory** field is displayed. You must specify a shared drive to store the shared information among the BPM nodes. Set this shared drive with a different location from that of the **ActiveMatrix Administrator** replication mode. For more information, refer to the section `Configuring a Stack to Setup BPM Fault Tolerance`.

Figure 5 ActiveMatrix Administrator Only mode and BPM Server



3. **Deployment Persistence**: select the Deployment Persistence check box. Deployment persistence ensures that restart of an ActiveMatrix BPM Server Component maintains deployment configuration continuity for the TIBCO ActiveMatrix BPM Server Component and the runtime component instances of TIBCO ActiveMatrix BPM. Deployment persistence preserves the deployment directory, files, and the configuration, so that a restart reestablishes the same connections.

   When Deployment Persistence is selected, the ActiveMatrix BPM Admin URL is preserved. It does not mean that the IP address of the TIBCO Runtime host instance name is preserved unless you define a resource allocation preference to make sure the component is attached to a fixed machine.

   ActiveMatrix BPM Server components that run an ActiveMatrix Administrator, run modes "ActiveMatrix Administrator Only" or "ActiveMatrix Administrator and BPM Server", still require a dedicated host

name. ActiveMatrix Administrator must be run from a dedicated SystemHost or SystemNode and moving to a new machine is not currently supported.

.

When Deployment Persistence is enabled and the TIBCO ActiveMatrix BPM Server Component is published, subsequent changes to the Component configuration will not be applied to the existing deployment. Deployed Component settings are protected in a persistent instance that can be restarted on any engine on the same machine.

EXCEPTION: When Deployment Persistence is enabled the following settings can be changed and published to existing published components:

- SSL Enabled (and most all SSL configuration settings)
- Use Internal LDAP Server (and all external LDAP settings)
- External Admin HTTP Base Port
- Enable Metrics Collection Runtime (MCR)
- Enable Common Logging

If you wish to change another component setting that was deployed as persistent, you have to start a new component on a new engine. You must copy the component and change that copy prior to publishing it within a stack to a clean Silver Fabric Engine instance.

When Deployment Persistence is selected and an ActiveMatrix BPM Admin or ActiveMatrix BPM Runtime component are stopped, they can be restarted on any Silver Fabric engines on the same machine.

When Deployment Persistence is **not** selected, then an orderly shutdown of the component will cleanup and remove the deployment directory and the associated database tables.

When you no longer want a persistent deployment on a published instance, the component Deployment Persistence check box must be cleared and the component must be restarted. On next orderly shutdown after restart the deployment directory will be cleared.

New component instances can be started on any cleaned TIBCO Silver Fabric Engine.

4. **Enable high availability and fault tolerance:** select this checkbox if you want to have high availability and fault tolerance, in the **ActiveMatrix Administrator and BPM Server mode** or **BPM Server Only** mode. The **Fault Tolerance Directory** field becomes available after selecting this checkbox. Similarly for Administrator only mode, if you select the check box **Replicate AMX**

Administrator Instance then the the **Fault Tolerance Directory** field becomes available.

5. **SSL Enabled**: communication channels between ActiveMatrix BPM nodes, runtime applications, database, LDAP, and TIBCO EMS may be secured with SSL.

   Use the **SSL Enabled** check box, when you want to use SSL/HTTPS. This check box adds a page to the BPM Server Component Creation wizard to prompt you to enter SSL Configuration details for the Admin and EMS connections. The keystore types, aliases, and passwords can be set for use of the respective certificates.

6. Click **Finish** or proceed until you can click **Finish** to save your changes.

You must click **Next** or **Finish** to confirm any changes you make on any Component Wizard configuration page. If you click **Cancel** or **Menu** after making a change it will not be recorded. Additionally, you must click **Finish** to properly save any changes made in the Component Wizard or the Silver Fabric Administrator will revert any changes to the last completed version.

Upload JDBC Drivers

TIBCO ActiveMatrix BPM running on TIBCO Silver Fabric engines supports use of a subset of a third-party databases supported by the classic product. Refer to the readme file for the official list of supported databases compatible with this release.

When using Oracle 11g R2, upload the JDBC driver file named `ojdbc6.jar`. Because of distribution constraints the file name, `ojdbc6.jar`, is required.

When using Microsoft SQL Server upload the Microsoft JDBC driver for the connection.

When using the Administrator Only run mode for development purposes, you can use the in-process HSQL database, but most enterprise implementations on the cloud depends upon a more robust database. Even when configuring an "Admin Only" component, you must upload the appropriate JDBC driver for the ActiveMatrix BPM Servers that will connect to the ActiveMatrix Administrator, even if the Administrator is just going to be using the default in-process HSQL database.

The Component Wizard will prompt you to enter the database connection parameters on other pages for other dependent parts like the ActiveMatrix Administrator Database Configuration page.

ActiveMatrix Administrator Basic Configuration

Define the ActiveMatrix Administrator Basic Configuration parameters.

Enterprise Name

Enter an ActiveMatrix Administrator Enterprise Name. The Enterprise Name must be unique because only one ActiveMatrix Administrator may run on an Enterprise.

To prevent any operating system path length limitations from becoming an issue in Windows hosts, the following best practices can be used in object naming.

- Use an **Enterprise Name** that is as short as possible, e.g. "N1" or "N2" etc.… instead of "testDevNode"
- Use a **Server Name** that is as short as possible, e.g. "`i1`" instead of "`instanceOne`"
- Use a **Deployment Directory** with as few characters as possible: "`C:\AMX`" instead of "`C:\ActiveMatrix`"
- Install the TICBO Silver Fabric Engine in a directory with a short path and file name, e.g. "`C:\SF`" instead of "`C:\Engine`"
- Choose host names so they will be as short as possible: 8 characters or less should be fine.

Server Name

Enter a server name that is unique within the enterprise. The server name is also known as the server instance name and the ActiveMatrix BPM server host name. Multiple ActiveMatrix BPM server host instances may make reference to a single ActiveMatrix Administrator instance.

It is best to minimize the length of the server names.

Enable Metrics Collection Runtime (MCR)

The TIBCO ActiveMatrix BPM Metrics Collection Runtime (MCR) is a monitoring service used to collect runtime object performance statistics for diagnosis and performance tuning using the TIBCO ActiveMatrix BPM Administrator Infrastructure Dashboard. MCR may prove very useful for governance and development performance tuning.

For more information on using the monitoring service, refer to the chapters on *Monitoring and Governance in the TIBCO ActiveMatrix BPM documentation*.

Enable Common Logging

Select the Enable Common Logging check box during development and testing phases of your implementation. Common logging can help you to debug and to verify proper and expected functionality.

When using common logging, be sure to enable Metrics Collection Runtime so those measures may be logged.

Clear the Enable Common Logging check box to disable the logging service and to provide an incremental advantage in processing speed.

Port Settings

All the port settings initially point to the proper default values. Normally the port settings should be left alone unless your implementation environment has port conflicts that require changes. For more detailed descriptions of these and other settings refer to the *TIBCO Silver Fabric Users Guide* and *TIBCO ActiveMatrix BPM Administration Guide.*

External Admin HTTP Base Port

By default the HTTP and HTTPS port used by TIBCO ActiveMatrix BPM Administrator and CLI is 8120, but that port may be changed here if there is a port conflict.

For all Base Port configuration parameters the actual port number at runtime will be the value of the [`Base Port`] + [`Engine Instance Number`] on which the component is running.

Internal Admin HTTP Base Port

The ActiveMatrix BPM Administrator internal HTTP port (default 19767) is used for communication between ActiveMatrix BPM Administrator and other ActiveMatrix BPM Server hosts and BPM Runtime nodes. In a Distributed BPM configuration, a load balancer must be set to manage HTTP requests between the primary and remote machines. For more information on a Distributed BPM configuration refer to the *TIBCO ActiveMatrix BPM Installation Guide*.

System Host Management Base Port

The System Host communicates with the TIBCO ActiveMatrix BPM Administrator Server using port 6051 by default. Port conflicts may force you to change this.

System Node Management Base Port

The management port of the SystemNode is used to run the ActiveMatrix BPM Administrator server. The default port is 6021.

Admin Credential Server Base Port (Credential Server port)

Specifies the Management port number for the TIBCO Credential Server which communicates with the SystemNode, hosts, and the Administrator Server.

The default port is 6041.

Admin Credential Server User Name and Password

User identifier and password for TIBCO Credential Server clients.

ActiveMatrix Administrator EMS Configuration

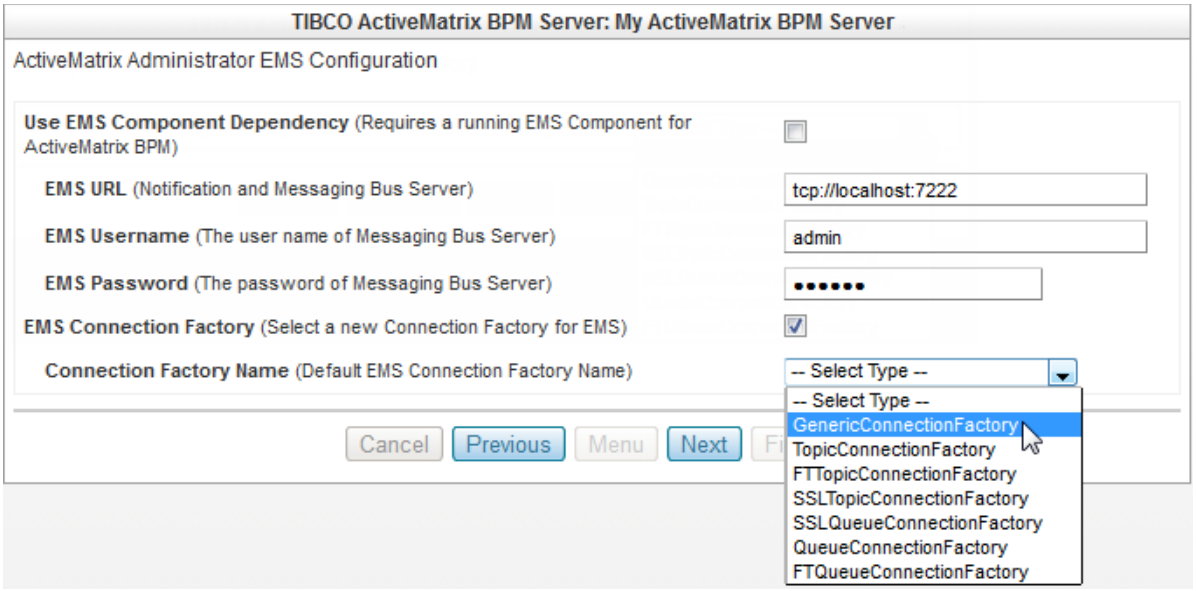
Use EMS Component Dependency

Check this box to require the use of a TIBCO Enterprise Messaging Service (EMS) instance published using the same stack using TIBCO Silver Fabric Enabler for TIBCO EMS. Using a TIBCO EMS component dependency can simplify and speed up your deployment because the connections between the TIBCO ActiveMatrix BPM components and TIBCO EMS component are automated even though you do not necessarily know during development which engine will receive the TIBCO EMS instance unless you have already installed it or explicitly defined where it will be.

When creating your stack for an ActiveMatrix BPM environment, you must also define an ActiveMatrix BPM server component dependency on the EMS component you have created for the stack. Refer to Creating an ActiveMatrix BPM Stack on page 62 for more information. The connection between the ActiveMatrix BPM Server and the TIBCO EMS instances will be handled automatically.

If you want to use an external EMS provider, clear the **Use EMS Component Dependency** check box and then you can specify how the ActiveMatrix BPM server will connect with an external EMS provider.

Figure 6 ActiveMatrix Administrator EMS Configuration



EMS Connection Factory

The EMS Connection Factory check box is used to select a new TIBCO EMS Connection Factory that can be used by the TIBCO ActiveMatrix BPM Server instance.

Connection Factory Name

After the EMS Connection Factory check box is selected, the Connection Factory Name menu is used to select the default TIBCO EMS connection factory:

- Generic Connection Factory
- Topic Connection Factory
- Fault Tolerant Topic Connection Factory
- SSL Topic Connection Factory
- SSL Queue Connection Factory
- Queue Connection Factory
- Fault Tolerant Queue Connection Factory

Descriptions of the Connection Factory and Queue Connection Factory types can

be found in the TIBCO Enterprise Messaging Service documentation.

Configure the ActiveMatrix Administrator Database

The ActiveMatrix Administrator Database Configuration page is used to specify if your component will use the in-process database or an enterprise ready third-party database.

For components that are going to be used for only a development implementation you can leave the **Use Default In-Process Database** check box selected to use the embedded in-process ActiveMatrix Administrator database.

For most other use cases clear the check box to reveal the database configuration parameters to use for establishing a connection between TIBCO ActiveMatrix Administrator and the third-party database.

Database Type

If you will use an external database, select either an Oracle 11g or Microsoft SQL Server to use with your ActiveMatrix Administrator. Refer to the readme for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM Stack you must have the database allocated and ready to support a database system administrator connection for automated schema creation by the BPM Server Component.

For whichever database your implementation will use, you must upload the JDBC database driver for it.

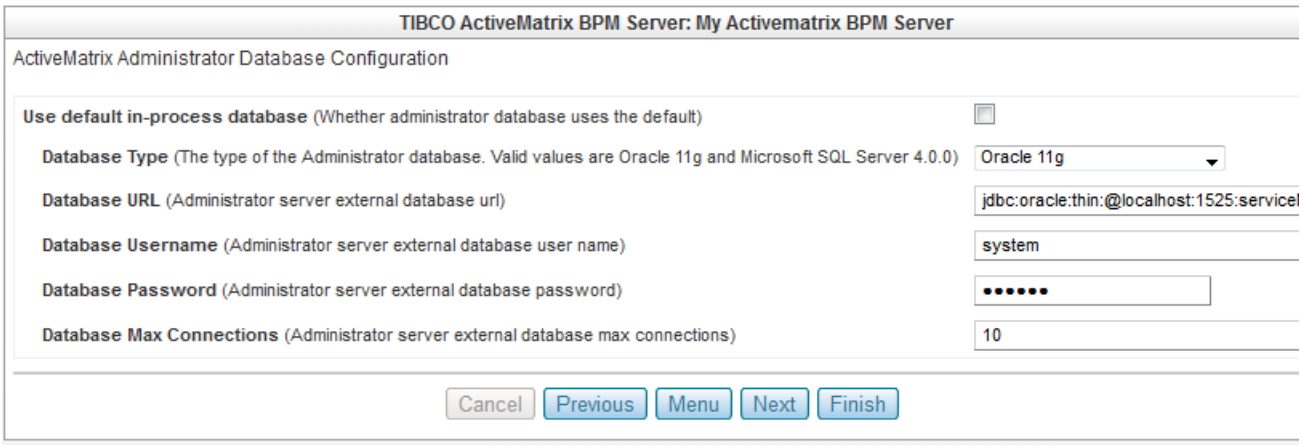
Select the supported database to use for your implementation.

a. Oracle 11g

b. Microsoft SQL Server 4.0.0

Whichever database you select, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12

Figure 7 BPM Database Administrator Configuration for Oracle 11g



Database URL

Enter the Database URL. ActiveMatrix Administrator will use the URL to connect with an external database.

For an Oracle Database instance the format of the string can be either:
`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:
`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

Database User Name and Password

Enter the Database user name and password. The user profile used must have system administrator privileges.

Database Max Connections

Set the maximum connections that may be made between ActiveMatrix Administrator and the external database.

ActiveMatrix Administrator Authentication Realm Configuration

ActiveMatrix Administrator may authenticate using either a database (internal or external) or an LDAP server.

The next Component Wizard page lets you configure the database connection or the LDAP authentication realm details depending on your authentication realm type selection.

For a database connection refer to ActiveMatrix Administrator Database Authentication Realm Configuration on page 19.

For LDAP refer to ActiveMatrix Administrator LDAP Authentication Realm Details on page 20.

User Name and Password: These fields apply to an LDAP Authentication Realm. If you are using an LDAP Authentication Realm make sure the ActiveMatrix Administrator user name and password exists in your LDAP server.

ActiveMatrix Administrator Database Authentication Realm Configuration

If you want to use the configured ActiveMatrix Administrator database for the authentication realm, simply leave the check box for **Database Authentication Realm: Use ActiveMatrix Administrator database** selected and you will not have to configure the database connection for the authentication realm.

Otherwise clear the check box for **Database Authentication Realm: Use ActiveMatrix Administrator database** to reveal the database configuration parameters.

Database Type

If you use a separate, external database, select either an Oracle 11g or Microsoft SQL Server to use with your ActiveMatrix Administrator. Refer to the readme file for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM component you will need to have the database allocated and ready to support a database system administrator connection. This is required for automated schema creation by the BPM Server Component.

For whichever database your implementation will use, you must upload the JDBC database driver compatible for it.

Select the supported database to use for your implementation.

a. Oracle 11g

b. Microsoft SQL Server 4.0.0

Whichever database you select, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12

Database URL

Enter the Database URL ActiveMatrix Administrator will use to connect with an external database.

For an Oracle Database instance the format of the string can be either:
`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:
`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

Database User Name and Password

Enter the DBA user name and password. The user profile used must have system administrator privileges.

Database Max Connections

Set the maximum connections that may be made between ActiveMatrix Administrator and the external database.

ActiveMatrix Administrator LDAP Authentication Realm Details

When you select **LDAP** as the ActiveMatrix Administrator authentication realm type, a connection must be configured to connect with the LDAP provider. The ActiveMatrix Administrator LDAP Authentication Realm Details page lets you specify the configuration parameters needed.

The TIBCO ActiveMatrix Administrator user name profile must exist on that LDAP provider.

The Administrator LDAP Authentication Realm Details page is displayed after you click **Next** on the Basic Configuration page. Using the LDAP Authentication Realm Details page, you can specify many settings, which are described in the *TIBCO ActiveMatrix BPM Installation and Configuration* guide in the section: Administrator Server Configuration: LDAP Authentication Realm Details. The parameter descriptions are duplicated here for your convenience.

Machine Name Port List

Comma-separated list of URLs for an LDAP server. To achieve fault tolerance, you can specify multiple URLs. For example:

`ldap://server1.example.com:686,ldap://server2.example.com:1686`

Bind DN Name

The superuser's distinguished name or superuser's name to be used to connect to the server.

Default: `uid=Manager,ou=people,dc=example,dc=com`

Password

LDAP server password.

Context Factory

The factory object that provides the starting point for resolution of names within the LDAP server.

Default: `com.sun.jndi.ldap.LdapCtxFactory`

User Search Base DN

Base distinguished name from which the search starts.

Default: `ou=people,ou=na,dc=example,dc=org`

User Search Expression

The expression used for searching a user. For example: `(CN=%U)`. `"%U"` is replaced by the user name being searched for. You can define any complex filter such as `(&(cn=%U)(objectClass=account))`.

Default: `(&(uid={0})(objectclass=person))`

User Attribute with User Name

The name of the attribute in the user object that contains the user's name.

Default: uid

Search Timeout

The time (milliseconds) to wait for a response from the LDAP directory server. The default is: 30000

Follow Referrals

Whether to follow LDAP referrals. When selected, requests to LDAP can be redirected to another server.

Select this field to indicate that the LDAP information might be available at another location, or possibly at another server or servers.

Group Indication

Specifies how a user's group memberships are found. Group information is used by ActiveMatrix Administrator when a user, once authenticated, performs other activities in the system.

Options:

- **Group has users** List of users that belong to the group. When selected, the Group Attribute with User Names field is enabled.
- **User has groups** List of groups to which the user belongs. When selected, the User Attribute with Group Names field is enabled.

Default: Group has users

Group Search Base DN

Base distinguished name from which the search for the group starts.

Default: `ou=groups,ou=na,dc=example,dc=org`

Group Search Expression

Search by matching this expression against potential groups.

Default: `cn={0}`

Group Attribute with User Names

Name of the attribute in the group object containing its users.

Example: uniqueMember (OpenLDAP) or member (ActiveDirectory).

Default: uniqueMember

Group Attribute with Group Name

Name of the attribute in the group object that contains the name of the group.

Example: cn (OpenLDAP) or sAMAccountName (ActiveDirectory).

Default: cn

Group Attribute with Subgroup Names

Name of the attribute in the group object that contains its subgroups.

Example: uniqueMember (OpenLDAP) or member (ActiveDirectory).

Default: uniqueMember

User Attribute with Group Names

Name of the attribute in the user object that lists the groups to which the user belongs.

Default: None

Group Search Scope Subtree

When searching the group, indicate whether to traverse into the subtree or to search only under the group base distinguished name.

Default: Selected

User Search Scope Subtree

Whether to search the entire subtree starting at the base DN, or search only the nodes one level below the base DN.

Default: Selected

Security Authentication

Value of Simple Authentication and Security Layer (SASL) authentication protocol to use. Values are implementation-dependent. Some possible values are:

Simple - User name and password are required.

None - No log-in is performed, any other settings are ignored, and the LDAP client is anonymous.

Default: Simple

LDAP Server is SSL-Enabled

Indicate that the LDAP server is enabled for SSL. When checked, the SSL Trust Store Configuration fields are enabled.

Trust Store Location

Upload the Trust Store file (as either a JKS or JCEKS file) to be used by the ActiveMatrix Administrator to establish a secured connection with the LDAP server.

Trust Keystore Type

The type of the uploaded trust store: JKS or JCEKS.

Trust Store Password

Either the password specified in the Create a Trust Store wizard, or the password required to access the trust store that is to be uploaded.

ActiveMatrix Administrator Monitoring Service Configuration

The ActiveMatrix Administrator Monitoring Service can be set to use the internal database and messaging systems or it can be configured to use external providers.

Monitoring Service Database

Specify whether the monitoring service should use the same configured database as installed with the ActiveMatrix Administrator server. If you clear the check box you will need to configure the monitoring service to connect to an external database.

Database Type

If you use an external database, select either an Oracle 11g or Microsoft SQL Server to use with ActiveMatrix Administrator. Refer to the readme file for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM component you will need to have the database allocated and ready to support a database system administrator connection for automated schema creation by the BPM Server Component.

For whichever database your implementation use, you must upload the JDBC database driver compatible for it on the previous

Select the supported database to use for your implementation.

a. Oracle 11g

b. Microsoft SQL Server 4.0.

Whichever database you select, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12

Database URL

Enter the Database URL ActiveMatrix Administrator will use to connect with an external database.

For an Oracle Database instance the format of the string can be either:

`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:

`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

Database User Name and Password

Enter the DBA user name and password. The user profile used must have system administrator privileges.

Database Max Connections

Set the maximum connections that may be made for this service between ActiveMatrix Administrator and the external database.

Monitoring Service Messaging

Specify whether the monitoring service should use the same messaging service as the ActiveMatrix Administrator server. If you are using the same messaging service there is nothing else you need to configure. If you clear the Monitoring Service check box you will need to configure the monitoring service to connect to an external Enterprise Messaging Service.

EMS URL - EMS uses TCP and the default port used for messaging is 7222. The format for the EMS URL will be: `tcp://[`*HostName*`]:[`*Port*`]`

The actual port value used is implementation dependent. EMS can be hosted locally or it may be hosted by an external installation.

The EMS user name and password should be an EMS administrator profile to ensure full functionality of the component.

If TIBCO ActiveMatrix BPM Administrator is going to be dependent on a external TIBCO Enterprise Message Service instance, then TIBCO EMS must be running before the TIBCO ActiveMatrix BPM server component is started.

To ensure that happens you must either Adding Allocation Rules with a component dependency for the TIBCO ActiveMatrix BPM Server component so the EMS component is running before it is started, or set a Stack dependency for BPM Server component depending on the EMS component.

ActiveMatrix Administrator Log Service Configuration

The ActiveMatrix Administrator Log Service can be set to use the internal database or it can be configured to use external providers.

Log Service Database

Specify whether the log service should use the same configured database as with the ActiveMatrix Administrator server. If you are using the same database there is nothing else you need to configure. If you clear the check box you will need to configure the log service to connect to an external database.

Database Type

If you use an external database, select either an Oracle 11g or Microsoft SQL Server to use with ActiveMatrix Administrator. Refer to the readme file for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM component you must have the database allocated and ready to support a database system administrator connection for automated schema creation by the BPM Server Component.

For whichever database your implementation will use, you must upload the JDBC database driver compatible for it on the previous

Select the supported database to use for your implementation.

a. Oracle 11g

b. Microsoft SQL Server 4.0.0

Whichever database you select, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12

Database URL

Enter the Database URL ActiveMatrix Administrator will use to connect with an external database.

For an Oracle Database instance the format of the string can be either:
`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:
`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

Database User Name and Password

Enter the DBA user name and password. The user profile used must have system administrator privileges.

Database Max Connections

Set the maximum connections that may be made for this service between ActiveMatrix Administrator and the external database.

ActiveMatrix Administrator Payload Service Configuration

The ActiveMatrix Administrator Payload Service can be set to use the same database as the Administrator or it can be configured to use external providers.

Payload Service Database

Specify whether the payload service should use the same database as the ActiveMatrix Administrator server. If you are using the same database there is nothing else you need to configure. If you clear the check box you will need to configure the payload service to connect to an external database.

Database Type

If you use an external database, select either an Oracle 11g or Microsoft SQL Server to use with ActiveMatrix Administrator. Refer to the readme file for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM component you must have the database allocated and ready to support a database system administrator connection for automated schema creation by the BPM Server Component.

For whichever database your implementation uses, you must upload the JDBC database driver compatible for it.

Select the supported database you to use for your implementation.

a. Oracle 11g

b. Microsoft SQL Server 4.0.0

Whichever database you select, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12.

Database URL

Enter the Database URL ActiveMatrix Administrator will use to connect with an external database.

For an Oracle Database instance the format of the string can be either:

`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:

`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

Database User Name and Password

Enter the DBA user name and password. The user profile used must have system administrator privileges.

Database Max Connections

Set the maximum connections that may be made for this service between ActiveMatrix Administrator and the external database.

ActiveMatrix BPM Database Administrator Configuration

The ActiveMatrix BPM database will be created automatically.

Database Driver

Select either an Oracle 11g or Microsoft SQL Server driver to use with your ActiveMatrix BPM Database Administrator. Refer to the readme for specific ActiveMatrix Administrator versions supported for third-party database.

Prior to publishing and deploying your BPM component you must have the database allocated and ready to support a database system administrator connection for automated schema creation by the BPM Server Component.

For whichever database your implementation will use, you must also upload the JDBC database driver compatible for it.

If you are configuring a "BPM Server Only" component, you must make sure that the "Admin Only" or "Admin+BPM" node also has the appropriate driver

Whether you select Oracle 11g or MS SQL Server 4.0.0, you must also make sure that you have uploaded the appropriate JDBC driver to support the connection configurations specified here. The component wizard has a separate page for the JDBC driver upload. Refer to Upload JDBC Drivers, page 12.

URL

Enter the Database URL. ActiveMatrix Administrator will use the URL to connect with an external database.

For an Oracle Database instance the format of the string can be either:

`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:
`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor`

DBA User Name and DBA Password

Enter the DBA user name and DBA password. The user profile used must have system administrator privileges.

Tablespace Name

Name of the Oracle tablespace where the BPM database will be created.

If the tablespace does not already exist, it will be created.

The Administrator database and the BPM database must each use a separate tablespace.

If multiple BPM databases are hosted on this server, each subsequent BPM database must also use a separate tablespace.

If the BPM database will be hosted on Microsoft SQL Server this field is ignored.

Default: `bpm_tablespace`

The build.properties entry: `bpm.db.tablespace`

ActiveMatrix BPM Runtime Database Configuration

URL

The Activematrix BPM Runtime Database URL will be used by all the nodes in the environment.

Enter the JDBC connection string that identifies the ActiveMatrix BPM database to be used by this BPM environment.

Enter the Database URL. ActiveMatrix BPM Runtime will use the URL to connect with the ActiveMatrix BPM database.

For an Oracle Database instance the format of the string can be either:
`jdbc:oracle:thin:@[HostName]:[Port]:[ServiceID]` or
`jdbc:oracle:thin:@//[HostName]:[Port]/[ServiceName]`

For a Microsoft SQL Server instance the URL should be in the format:
`jdbc:sqlserver://[HostName]:[Port];DatabaseName=[DBName];SelectMethod=cursor)`

ActiveMatrix BPM will use this string to connect to this ActiveMatrix BPM database.

User Name and Password

Enter the user name and password. The user profile used must not already exist in the database. It is added during component activation.

This is the database server user account that ActiveMatrix BPM uses to connect to this ActiveMatrix BPM database.

The specified user name must be unique to this BPM system. You cannot use an existing user account that is used by a different BPM database on this server.

Default: *bpmuser*

The BPM Runtime Database user corresponds with the build.properties entry: `bpm.db.username`

Confirm the password for this user.

ActiveMatrix BPM BDS Database Configuration

Specify whether the ActiveMatrix BPM Business Data Services (BDS) for releases 3.0 and later (not 2.2.x) should use its own database separate from other ActiveMatrix BPM components. If it does not matter to your implementation whether the BDS database is separated from other component databases, then you can leave the check box cleared and you need not configure a separate tablespace.

Select the **Use own database for BDS** check box to configure the connection that ActiveMatrix BPM should use to connect to the ActiveMatrix BPM BDS database.

URL

Enter the JDBC connection string where you want to create the separate ActiveMatrix BPM BDS database.

Enter the Database URL. ActiveMatrix BPM Runtime will use the URL to connect with the ActiveMatrix BPM database.

For an Oracle Database instance the format of the string can be either:

`jdbc:oracle:thin:@[`*`HostName`*`]:[`*`Port`*`]:[`*`ServiceID`*`]` or

`jdbc:oracle:thin:@//[`*`HostName`*`]:[`*`Port`*`]/[`*`ServiceName`*`]`

For a Microsoft SQL Server instance the URL should be in the format:

`jdbc:sqlserver://[`*`HostName`*`]:[`*`Port`*`];DatabaseName=[`*`DBName`*`];SelectMethod=cursor`

ActiveMatrix BPM will use this string to connect to this ActiveMatrix BPM database.

BDS User Name and BDS Password

Database server user account that ActiveMatrix BPM will use to connect to the ActiveMatrix BPM BDS database.

The BDS user name corresponds with the build.properties entry: `bpm.bds.username`

Tablespace Name

Name the new and separate ActiveMatrix BPM BDS database tablespace to be created for this component.

ActiveMatrix BPM Application and Client Configuration

BPM Application Name

Name of this particular ActiveMatrix BPM application instance. Each BPM system in the same ActiveMatrix environment must have a unique application name.

The default BPM application name: `${COMPONENT_NAME}` uses the component name as the BPM application name.

The BPM application name corresponds to the build.properties entry: `bpm.app.name`

HTTP Work Presentation Host

Name used to externally identify the machine that hosts this BPM system. This name is used when Work Manager distributes work using the push distribution model. When a user receives an email notification of a work item, the specified name is embedded in the URL in the email body that the user clicks to access the work item. (See "*Work Management > Distributing Work to Users" in TIBCO ActiveMatrix BPM Concepts* for more information.)

If this BPM system is distributed across different machines (to provide high availability and scalability), this should be the name of the load balancer. See Load Balancer.

Default: localhost

The HTTP Work Presentation Host name corresponds with the build.properties entry: `bpm.http.host`

HTTP Port

HTTP port number is used by this BPM node for communication with external clients (Openspace, Workspace and custom client applications).

This port must not be in use by any other ActiveMatrix node on the same machine.

If this BPM system is distributed across different machines (to provide high availability and/or scalability), this port should be configured for load balancing across all required machines. See Load Balancer.

Default: 8080

build.properties entry: bpm.http.port

SOAP Binding Version (BPM 3.0 and higher)

Version of the SOAP specification that will be applied to all SOAP bindings:

- on services exposed by this instance of the ActiveMatrix BPM application.
- on services and references exposed by process applications deployed to this BPM system.

SOAP versions 1.1 and 1.2 are supported.

Default: 1.1

build.properties entry: bpm.soap.version

SMTP Host

Name of the machine that hosts the SMTP server that is used by this BPM system to send pushed email work notifications and to execute email service tasks.

If the machine is not the one on which TIBCO ActiveMatrix BPM is running, its name must be specified using a fully qualified domain name or IP address.

Default: localhost

build.properties entry: bpm.smtp.host

SMTP Port

Port number on which the SMTP server is running.

Default: 25

build.properties entry: bpm.smtp.port

ActiveMatrix BPM Sizing Details Configuration

You can customize the size of many aspects of your TIBCO ActiveMatrix BPM Server installation. Specifically you can set the following items:

- Database Connection Pool Size
- Database Statement Cache Size
- Number of Process Engine Threads
- Maximum Java PermGen
- Number of Message Core Threads
- Number of Message Maximum Threads
- Java Heap Minimum Size
- Java Heap Maximum Size

It makes sense to customize these settings with the **Environment** setting even if you do not want to mess around with testing for optimal settings given your hardware implementation environment and project needs. With the **Environment** setting you can set a rough template of good default values with just the selection of either the Development or Production options.

Select the **Customize BPM Size** check box to reveal the following configuration parameters:

Environment

Choose an environment option either **Development** or **Production**. All the other parameters on this page will get default values for the environment type selected.

You can change any listed parameter or you can change nothing, but if you toggle the Environment again it will re-insert default values according to the Environment type selected.

Database Connection Pool Size

Number of database connections that are available for each of the ActiveMatrix BPM DataSource resource templates.

Database Statement Cache Size

Number of prepared statements that can be cached by each connection in the DataSource resource template pool. Caching prepared statements improves the performance but it uses more memory.

Number of Process Engine Threads

Number of threads in the Process Engine background thread pool.

Max Java PermGen Size

Maximum size for Java PermGen.

Development default: 256

Production default: 1024

build.properties entry: `bpm.sizing.jvm.maxpermsize`

Number of Message Threads

Range for the message threads. The development default: Core: 5 Maximum: 10

The production default: Core: 10 Maximum: 20

build.properties entries:

- (Core) bpm.sizing.core.msg.threads
- (Maximum) bpm.sizing.max.msg.threads

Java Heap Minimum and Java Heap Maximum Size

Range for the Java heap size.

Development default: Minimum: 512 Maximum: 1536

Production default: Minimum: 1024 Maximum: 4096

build.properties entries:

- (Minimum) bpm.sizing.jvm.xms
- (Maximum) bpm.sizing.jvm.xmx

The value for Java Heap Size is a starting point. You should monitor the system under load and adjust this value if necessary. For more information, refer to TIBCO ActiveMatrix Performance Tuning.

ActiveMatrix BPM LDAP Configuration

TIBCO ActiveMatrix BPM LDAP can be set to use an internal LDAP server or it can be configured to use an external LDAP server.

All settings on the ActiveMatrix BPM LDAP Configuration page may be changed and published to running instances of the BPM Server Component even when Deployment Persistence is enabled.

Use Internal LDAP Server

Specify if TIBCO ActiveMatrix BPM Server uses an internal LDAP to authenticate with ActiveMatrix BPM nodes. If you are using the internal LDAP server the only other configuration setting you must set is the internal LDAP port. If you clear the check box you must configure some LDAP connection parameters to connect with the external LDAP server.

URL

When an external LDAP server is used, enter that location and port here to establish the connection. The default value is: `ldap://localhost:10389`

User Search Base DN

Base distinguished name from which an LDAP search starts.

Default: `ou=system`

User Name and Password

Distinguished name and password of the LDAP user who will be used as the TIBCO-admin user.

ActiveMatrix BPM CMIS Configuration (BPM 3.1 or later only)

ActiveMatrix BPM can configure a Content Management Interoperability Services (CMIS) repository. Select the **Use CMIS Repository** check box and set the following configuration parameters.

URL

Specify the URL used to connect to the external CMS's CMIS interface.

Default: `http://localhost:10802/opencms/cmisatom`

`build.properties entry: bpm.cmis.url`

CMIS Binding Type

Type of binding that is used against the CMIS URL. Currently, only atompub is supported.

Default: atompub

build.properties entry: bpm.cmis.bindingtype

Repository ID

ID of the main repository within the CMS that is to be connected to.

Default: `cmis-online`

properties entry: bpm.cmis.repositoryid

AMX BPM Folder Path

Path of an existing folder in the main repository that is used to store case folder data. You must either specify a folder that already exists or create a new folder manually and specify that folder.

Default: `/amx-bpm`

build.properties entry: bpm.cmis.amxbpmfolderid

Admin User and Admin User Password

Name of the administrator user for authenticating against the CMIS URL.

Default: Admin

The value is stored in ECMCredentialsKeystore, and is used when connecting to the CMS server.

build.properties entry: bpm.cmis.adminuser

Password of the administrator user for authenticating against the CMIS URL.

Default: admin

The value is stored in ECMCredentialsKeystore, and is used when connecting to the CMS server.

build.properties entry: bpm.cmis.adminpassword

ActiveMatrix BPM JMS Configuration

This page allows you to choose whether to use the same EMS Server as the Administrator for ActiveMatrix BPM messaging, or to set up a distinct EMS server to keep the messaging separate.

Use Admin EMS Configuration

This check box option is only available when the run mode is set to have both ActiveMatrix Administrator and ActiveMatrix BPM Server running together from the same component and a TIBCO EMS instance is already set up for Administrator messaging. If that is the case you can leave the **Use Admin EMS Configuration** check box selected and the TIBCO ActiveMatrix BPM instance will use the same EMS server as the ActiveMatrix Administrator, making it easy to skip this configuration page.

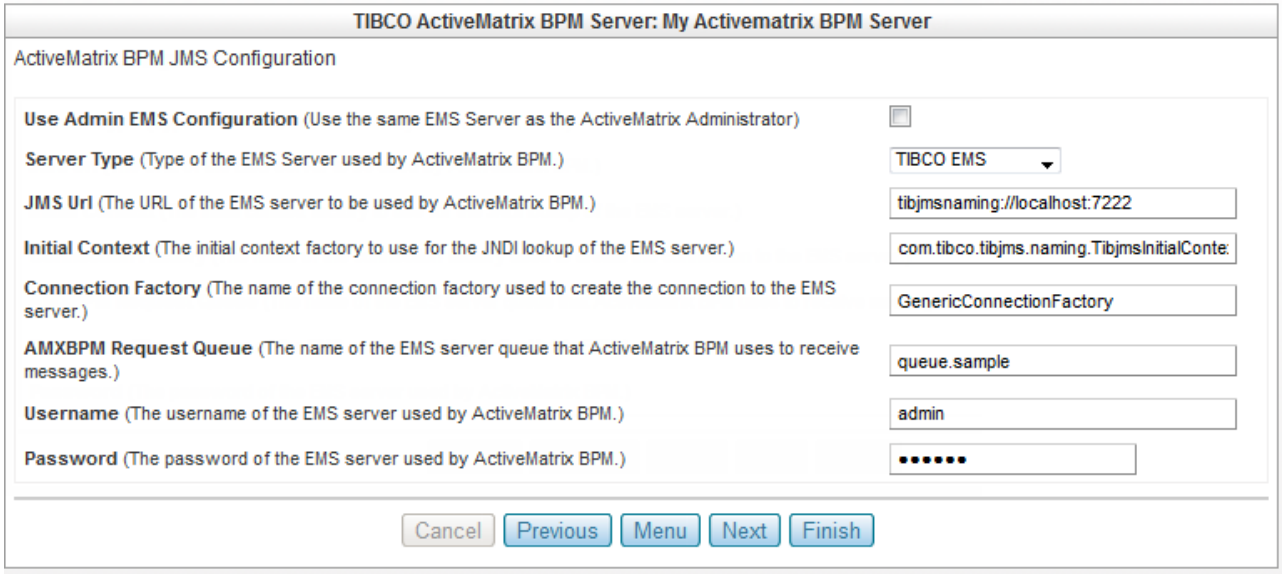
Clear this check box to use an external JMS server instance separate from the one used by the ActiveMatrix Administrator. After clearing the check box, the Component Wizard displays the BPM JMS Configuration page so you can configure the connection to an external TIBCO EMS for ActiveMatrix BPM Server messaging.

Use EMS Component Dependency (BPM Server Only run mode)

This check box option is only available when the ActiveMatrix Administrator EMS is not available, but instead you have a TIBCO EMS Component on which you can set a component dependency for messaging. If you will create a TIBCO EMS Component to run in the same stack as this component, you can set a dependency on that component and leave this check box selected.

You must clear this check box and configure a connection with a TIBCO EMS instance if you don't have a TIBCO EMS Component on which you can set a dependency.

Figure 8 Activematrix BPM JMS Configuration



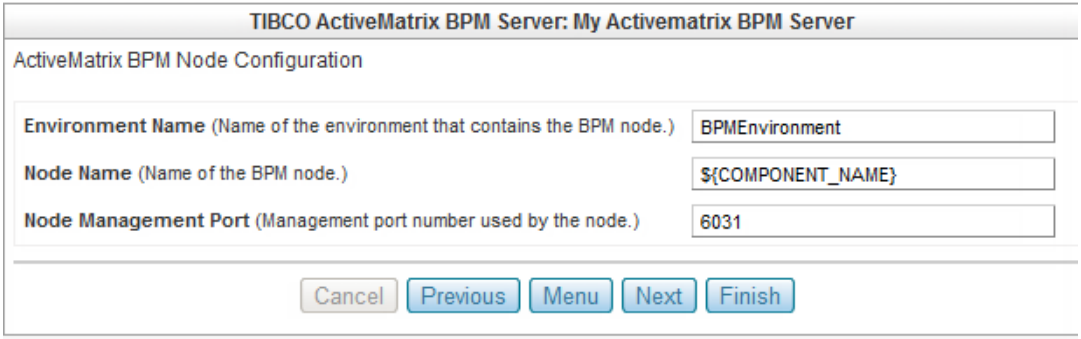
For more information on configuring a connection with an external JMS provider refer to the ActiveMatrix BPM: JMS Configuration page in the TIBCO ActiveMatrix BPM Installation Guide.

Or you can refer to the TIBCO Enterprise Message Service documentation available at http://docs.tibco.com.

ActiveMatrix BPM Node Configuration

The ActiveMatrix BPM Node Configuration page enables specification of the Environment Name, Node Name, and Node Management Port.

Figure 9 ActiveMatrix BPM Node Configuration



The Environment name must be unique for each TIBCO ActiveMatrix BPM Server. Multiple runtime nodes may be launched in each Environment, but TIBCO Silver Fabric Enabler for ActiveMatrix BPM supports only one ActiveMatrix BPM Server and one Environment per component.

Node Name

`${COMPONENT_NAME}` is replaced at run time with the name of the BPM server component.

Node Management Port

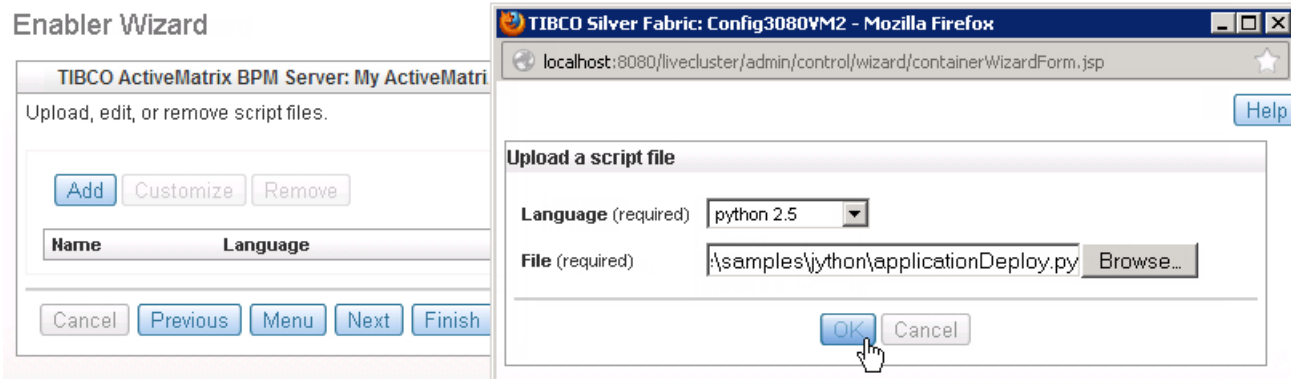
The **Node Management Port** is the port on which the debug node receives life cycle messages. Management port is used by the node.

The Create TIBCO ActiveMatrix BPM Server Wizard uses the default port: 6031

Using Scripts with the Enabler

The next diagram shows how a script may be uploaded. Scripts can be executed at various points in the engine life cycle and uploaded to use with the component or enabler. Python, Jython, and JavaScript are supported. Refer to the *TIBCO Silver Fabric Developers Guide* for more information on the methods that may be called.

Figure 10 Upload, Edit, or Remove Scripts



Scripts can be written to accomplish many tasks. Scripts may be used to automate publishing ActiveMatrix BPM application archives instead of manually publishing them.

TIBCO Silver Fabric has generic support for adding any JSR-223 compliant script engines that support interfaces that can be invoked and compiled. Support for Jython is included and the JDK includes JavaScript. Refer to https://developer.mozilla.org/en-US/docs/Rhino/Overview for more information.

Refer to the *TIBCO Silver Fabric Developer's Guide* on enabler and component scripting and methods for a more complete description of what can be done with scripts.

Add, edit, or remove script-provided statistics (optional)

If you have implemented the `ScriptStatisticsProvider` class added to this component by a script or an archive, then the **Add, edit, or remove script-provided statistics** page allows for the addition of archives from that statistics provider.

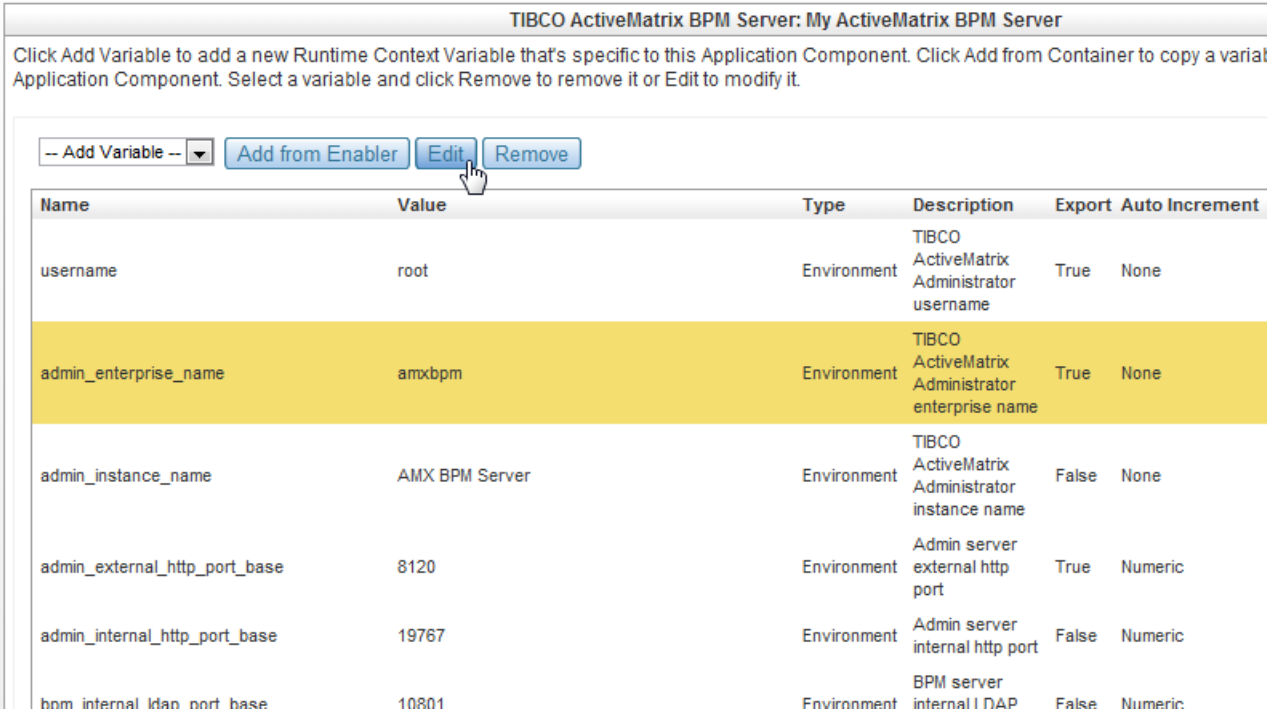
Refer to the *TIBCO Silver Fabric Developer's Guide* for information on implementing script-provided or archive-provided statistics.

Adding or Editing Runtime Context Variables

String, **Environment**, **System**, or **Encrypted** variables may be added and edited to the ActiveMatrix BPM Server component to define and set runtime specific context variables.

Select a variable type from the **Add Variable** pull-down list or **Add from Enabler** button to use a variable from a selected container.

Figure 11 Adding a Runtime Context Variable



Variable values from an Enabler may be added to the runtime as well. Use the **Add from Enabler** button to add container-specific context variables.

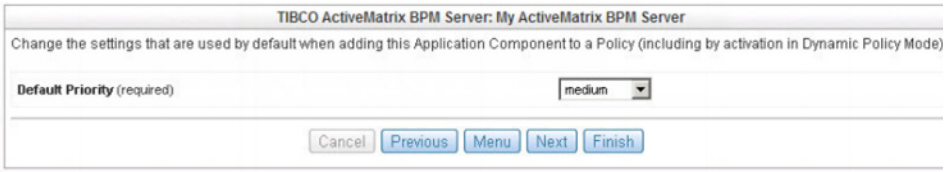
Runtime context variables may be changed by selecting the variable (selected row is highlighted) and clicking **Edit** to change its attributes. Selected rows may also be removed.

Scroll all the way to bottom to find the Component Wizard navigation buttons.

Setting Priority for the Component

The TIBCO ActiveMatrix BPM server component should be added at medium priority or higher to ensure that it is launched early in the stack.

Figure 12 Default Engine Allocation Settings



Adding Allocation Rules

Five different allocation rule types may be used to configure component behavior. Refer to the *Silver Fabric Help or the Silver Fabric User's Guide* for definitive information on allocation rules and controlling the component behavior.

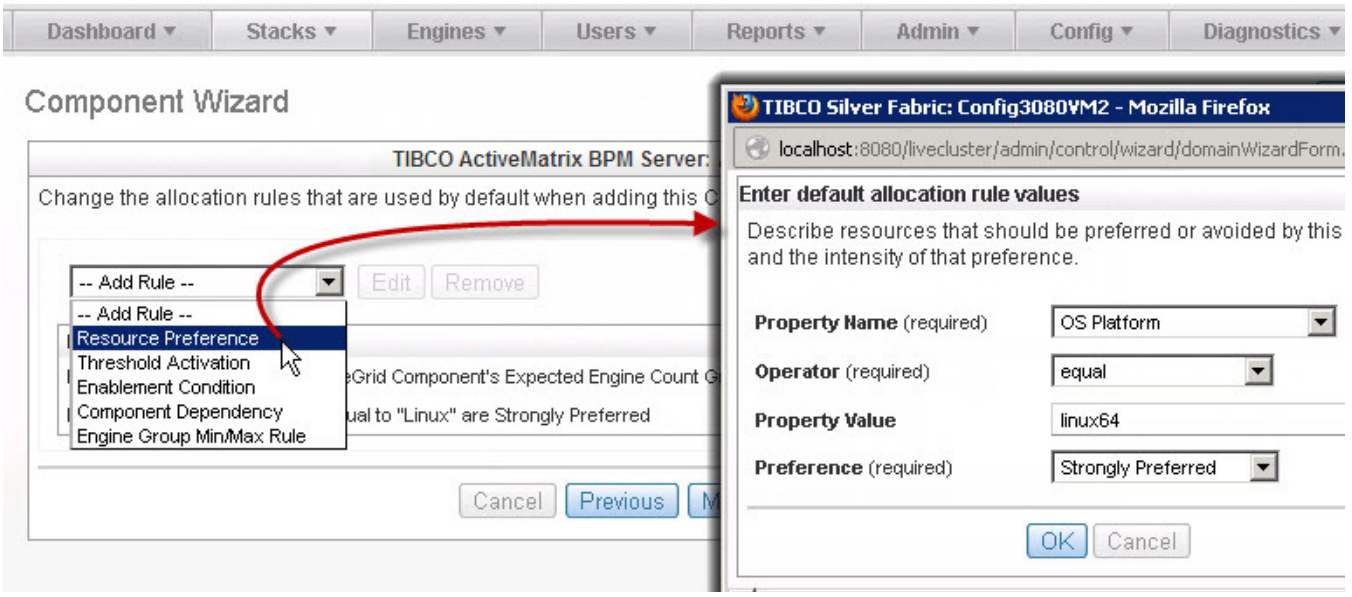
Add rules to specify and set component behavior. You can add rules to do the following tasks:

- Specify Resource Preferences,
- Set Thresholds for Activation,
- Set Enablement Conditions,
- Specify Component Dependency, or
- Set Engine Group Minimums or Maximums

Each rule selection will bring up a slightly different dialog window that allows property selection of a tracked engine or component archive statistic to be evaluated according to a logical operator and a value you specify to define an action. In some cases the component archive statistics may be selected.

As an example, Resource Preference, which is shown, lets you select a property, an operator, and a property value to specify a preference or requirement for how engines are allocated to a component based on the property condition defined.

Figure 13 Adding Resource Preferences for Engine Allocation



More information on using allocation rules is available in the *TIBCO Silver® Fabric Cloud Administration Guide*.

Patching at the Component Level

Patches can be added at the component level, providing an additional level of control over what will be patched and when.

Do not apply an ActiveMatrix BPM patch to an ActiveMatrix BPM Server Component running in **ActiveMatrix Administrator Only** run mode.

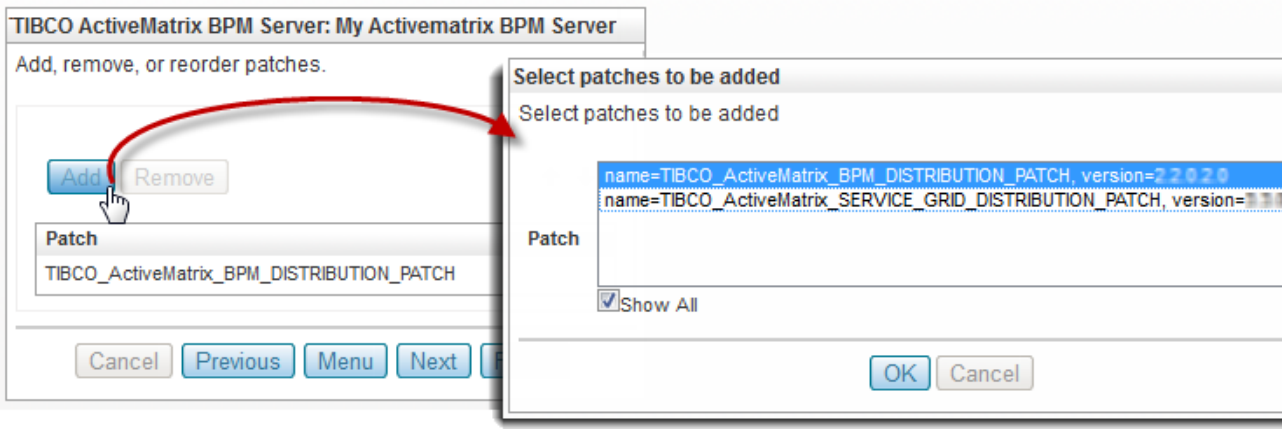
Patches can be installed on component activation either during an initial activation or during a restart, instead of just the "Apply Patch" action while running.

Simply place the zipped archive hotfix file in the appropriate TIBCO Silver Fabric Broker gridlib directory. The default broker gridlib directory is:

`SILVERFABRIC_HOME/webapps/livecluster/deploy/resources/gridlib/`

When configuring your component, the **Add, remove, or reorder patches** page will allow you to select and add the patch or patches as appropriate and applicable to your specific component. Some notes, conditions, and cautions do apply.

Figure 14 Selecting a patch in the Add, remove, or reorder patches page



For ActiveMatrix Patches

ActiveMatrix patches must be applied to both the ActiveMatrix BPM Server (regardless of run mode) and the ActiveMatrix BPM Runtime.

Make sure that on initial startup, the ActiveMatrix BPM Runtime node has the same patch level as the ActiveMatrix Administrator to which it will connect. After initial startup, the patch level of the different components can vary.

ActiveMatrix patches can be applied during the initial activation, or they can be applied later using the "Apply Patch" or "Restart Component" action.

For BPM Patches

ActiveMatrix BPM patches should only be applied to the ActiveMatrix BPM Server when the **Run Mode** is **ActiveMatrix Administrator and BPM Server** or **BPM Server Only**.

ActiveMatrix BPM patches should **NOT** be applied to a BPM Server when **Run Mode** is **ActiveMatrix Administrator Only**.

ActiveMatrix BPM patches should not be applied to component instances of the ActiveMatrix BPM Runtime because applying the patch to the ActiveMatrix BPM Server automatically upgrades the runtime nodes also when they are running during the upgrade. Any new ActiveMatrix BPM Runtime instances created after the patch was applied to the ActiveMatrix BPM Server will automatically start up with the hotfix patch already applied.

There is one caveat, any configured BPM runtime nodes must run for the upgrade to succeed.

This means the proper procedure for patching is one of the following:

Patching an existing system with BPM Server and BPM Runtime both Running

1. Add the patch to the ActiveMatrix BPM Server enabler or component.
2. Apply the patch using the **Apply Patch** action.

Alternatively, you can apply the patch using the **Restart Component** action but the warning applies.

Applying the patch with the **Restart Component** action will not work if the stack has a component dependency rule with a **Shutdown dependency** flag selected. Attempting to patch the ActiveMatrix BPM Server with the shutdown dependency configured will cause an error condition that will cycle the server continuously until the patch is removed from the enabler and component.

Patching BPM Server when running without BPM Runtime components

When patching ActiveMatrix BPM Server in this way you must not have any ActiveMatrix BPM Runtime components that have been activated with that ActiveMatrix BPM Server, especially BPM Runtime nodes with a persistent directory, even if they are currently shut down.

1. Add the patch to the ActiveMatrix BPM Server component.

2. Apply the patch to the ActiveMatrix BPM Server instance using either **Apply Patch** or **Restart Component**.
3. Start any new ActiveMatrix BPM Runtime component instances and they will be automatically updated with the patch that was applied to the ActiveMatrix BPM Server.

Component Wizard Generic Pages

Several of the Component Wizard pages are generic for all Silver Fabric enablers. Help buttons on these pages expose page specific documentation. Configurations for these generic settings are generally optional for the ActiveMatrix BPM Runtime component.

Refer to the page help and to *TIBCO Silver Fabric User's Guide and Administration Guide* for additional information on these configuration pages.

Finishing and Publishing a Component

You must click **Finish** to save a new component or to save any new changes to an existing component.

After you click the **Finish** button, make sure that the component is published to make it available when creating a Stack.

To do this, select **Publish Component** in the Actions drop-down list located on the same row as the Component.

Creating a TIBCO ActiveMatrix BPM Runtime Component

The TIBCO ActiveMatrix BPM Runtime is the runtime component used to start additional TIBCO ActiveMatrix BPM nodes.

Each TIBCO ActiveMatrix BPM Runtime Component requires a component dependency on a TIBCO ActiveMatrix BPM Server Component.

This configuration step is required. If the dependency is not set, the TIBCO ActiveMatrix BPM Runtime Component will fail to start.

The ActiveMatrix BPM Runtime Component will automate the following tasks:

- Create a TIBCO Host and add it to TIBCO ActiveMatrix BPM Enterprise.
- Configure a node of the specified type (BPM, ProcessManager, WorkManager, or WebComponents) to run on the TIBCO Host.
- Start and monitor the TIBCO Host and node instances.

Each ActiveMatrix BPM Runtime Component can start only one runtime instance per Silver Fabric engine host.

Create a TIBCO ActiveMatrix BPM Runtime Component

1. In TIBCO Silver Fabric Administrator GUI, select **Stacks > Components**.
2. Select **"Create New TIBCO ActiveMatrix BPM Runtime Component"** from the **Global Actions** menu.
3. Provide a name and description for the new ActiveMatrix BPM Runtime Component and click Next.
4. Select the TIBCO ActiveMatrix Product Distribution version you want to publish to the cloud. The TIBCO ActiveMatrix BPM Distribution versions available for selection are those you installed in TIBCO Silver Fabric *SILVERFABRIC_HOME* gridlib directory. Your version may be more recent than what is pictured here. Distributions that are installed and compatible with this enabler are displayed. All versions of the distributions are compatible. Click

Next and continue through the Component Wizard pages and configurations until you can click **Finish** to save your component configurations.

You must click **Next** or **Finish** to confirm any changes you make on any Component Wizard configuration page. If you click **Cancel** or **Menu** after making a change it will not be recorded. Additionally, you must click **Finish** to properly save any changes made in the Component Wizard or the Silver Fabric Administrator will revert any changes to the last completed version.

Define Basic Configuration

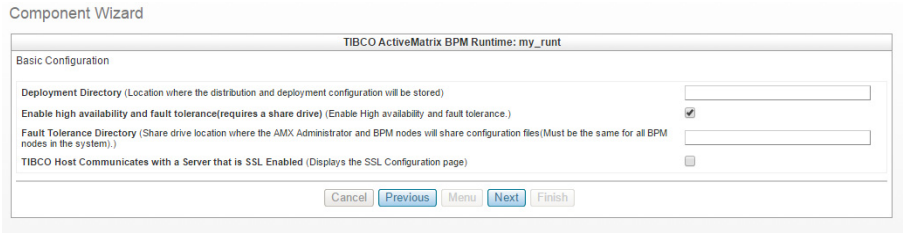
Deployment Directory - Defines the relative directory where the TIBCO ActiveMatrix Runtime Host will be installed on the Silver Fabric Engine.

For stacks that will include multiple ActiveMatrix BPM Runtime Components, a different and unique Deployment Directory path must be defined for each runtime component intended for deployment on the same machine. The same requirement applies for deployments of components on different machines if they use a shared file system.

Multiple instances of the same component will be assigned an instance ID number appended to the component name.

Default deployment directory is:
ENGINE_HOME`/work/`*[COMPONENT_NAME]+{INSTANCE_ID]*

Figure 15 Basic Configuration



Enable high availability and fault tolerance (requires a share drive) (Enable High availability and fault tolerance.): When this check box is selected, a new configuration field is displayed: **Fault Tolerance Directory (Share drive location where the AMX Administrator and BPM nodes shares the configuration files .** Note that this location must be the same for all BPM nodes in the system. The same value must be set for all BPM nodes in the system (including BPM server component and any other BPM runtime component in the same BPM system)

TIBCO Host Communicates with a Server that is SSL Enabled

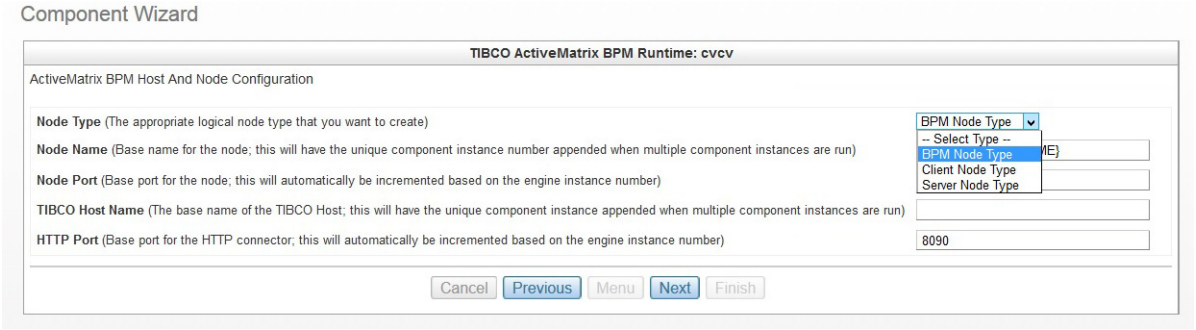
Select this box if you will run this ActiveMatrix Runtime component with an ActiveMatrix Server that is SSL certificate enabled.

If you have selected the check box **TIBCO Host Communicates with a Server that is SSL Enabled** then the SSL Configuration page will be displayed later in the Component Wizard to allow for upload of the trust store file and for other configurations.

Configuring ActiveMatrix BPM Host And Node

Every ActiveMatrix BPM Runtime Component creates a host and node environment ready to run applications. The **ActiveMatrix BPM Runtime Host and Node Configuration** page gives you control over the type of node you create. Your runtime component can be a BPM Node Type, a Processor Manager, a Work Manager, or a Web Components Node Type.

Figure 16 ActiveMatrix BPM Host and Node Configuration



Node Type - The ActiveMatrix BPM documentation perfectly describes different node types and how they can be used to your advantage. Refer to TIBCO ActiveMatrix BPM Architecture for an overview of node contents.

Node Name - **Node Port** - give your node a unique name and a port on which to communicate. Avoid assigning the same port to another runtime component. Identical applications messaging on the same port might cause conflicts.

Multiple ActiveMatrix runtime instances created from the same component will use port numbers that are incremented

TIBCO Host Name- Define a unique name for the individual ActiveMatrix BPM Runtime Host, so that the ActiveMatrix BPM environments can have even more logical separation when your components are running side by side. If the TIBCO Host Name field is left blank, the component name is used.

HTTP Port - Define a unique port for each Runtime Node to avoid conflicts.

ActiveMatrix BPM Sizing Configuration

Customize these TIBCO ActiveMatrix BPM Runtime settings with the **Environment** setting even if you are not taking the time to test your hardware implementation environment to determine optimal settings. With the **Environment** setting you can set a rough template of good default values by selecting either the Development or Production options.

Select the **Customize JVM Size** check box to expose these parameters to configuration.

Environment

Choose an environment option, either **Development** or **Production**. All other parameters on this page will get default values that are a good starting point for the environment type selected.

You can change any listed parameter or you can change nothing, but if you toggle the Environment again it will re-insert default values according to the Environment type selected.

Max Java PermGen Size

Maximum size for Java PermGen.

Development default: 256

Production default: 1024

build.properties entry: `bpm.sizing.jvm.maxpermsize`

Java Heap Minimum and Java Heap Maximum Size

Range for the Java heap size.

Development default: Minimum: 512 Maximum: 1536

Production default: Minimum: 1024 Maximum: 4096

build.properties entries:

- (Minimum) bpm.sizing.jvm.xms
- (Maximum) bpm.sizing.jvm.xmx

The value for Java Heap Size is a starting point. You must monitor the system under load and adjust this value if necessary. For more information, refer to TIBCO ActiveMatrix Performance Tuning.

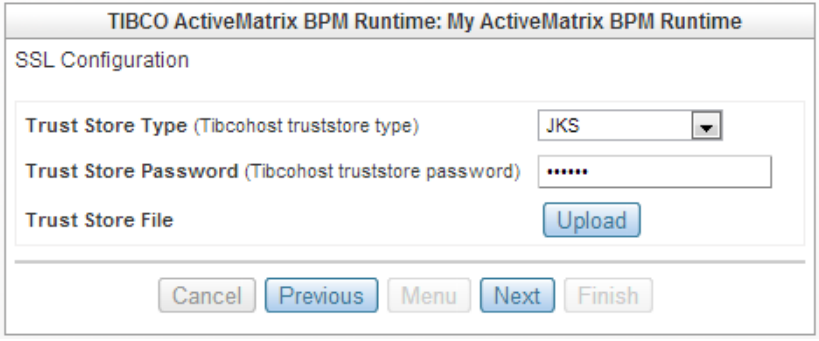
SSL Configuration (optional)

Set the **SSL Configuration** for a Runtime Server. This SSL Configuration page appears if you have selected the check box to use SSL with all TIBCO Host communications.

Trust Store Type

The trust store file can be either a JKS or JCEKS file. This trust store file will be used by the TIBCO ActiveMatrix BPM Runtime node to establish a secured connection with the TIBCO ActiveMatrix BPM Server.

Figure 17 SSL Configuration



SSL configuration is required if you have selected the use SSL check box. The Trust Store file must be uploaded for use with a valid password or the component will fail to establish a proper connection.

If the TIBCO ActiveMatrix BPM Server is using a certificate to secure communications then that certificate has to be added to the Runtime Server's Trust Store.

Custom Keystore Configuration

To establish secured communications over HTTP, LDAP, and WSS you must upload the keystore and truststore files for the communication channels you wish to secure.

Adding, Editing, and Removing Scripts from ActiveMatrix BPM Runtime

Script usage is the same for either the TIBCO ActiveMatrix BPM Server or Runtime components. Refer to Using Scripts with the Enabler on page 41 for a description of the functionality.

Adding, Editing and Removing Script-Provided Statistics

Script-provided statistics usage is the same for either the TIBCO ActiveMatrix BPM Server or Runtime components. Refer to Add, edit, or remove script-provided statistics (optional) on page 41 for information on this feature.

Adding or Removing Log File Patterns

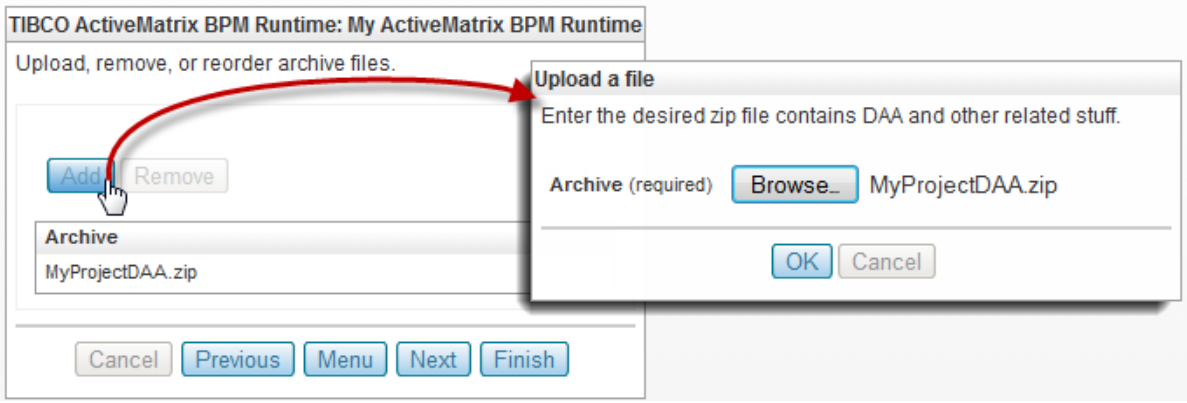
Adding and removing log file patterns usage is the same for either the TIBCO ActiveMatrix BPM Server or Runtime components. Refer to `Add or remove log file patterns`.

Adding and Removing Archive Files

Upload ActiveMatrix BPM Distributed Application Archive (DAA) files to publish and run ActiveMatrix BPM projects and applications. Application archives that are uploaded using the enabler are published with the component at run time. DAA files may be uploaded one at a time as follows:

1. Click the **Add** button in the Upload, Remove, or Recorder Archive Files panel, as shown in Figure 18.

Figure 18 Uploading Archives



2. Click the **Browse** button in the "Upload A File" panel to navigate to a compressed archive file that contains your DAA.

Your compressed file must contain the DAA file and optionally an XML properties file. The XML file can contain and define all the published configurations, path name, global variables, java heap size, and their respective values.

3. Click **OK**, and follow these steps:
   a. Repeat the steps to add more application archives to deploy and run, or
   b. Click **Next** to proceed with further configuration of the component, or
   c. Click **Finish** to complete and save configurations to the component.

Adding or Editing ActiveMatrix BPM Runtime Context Variables

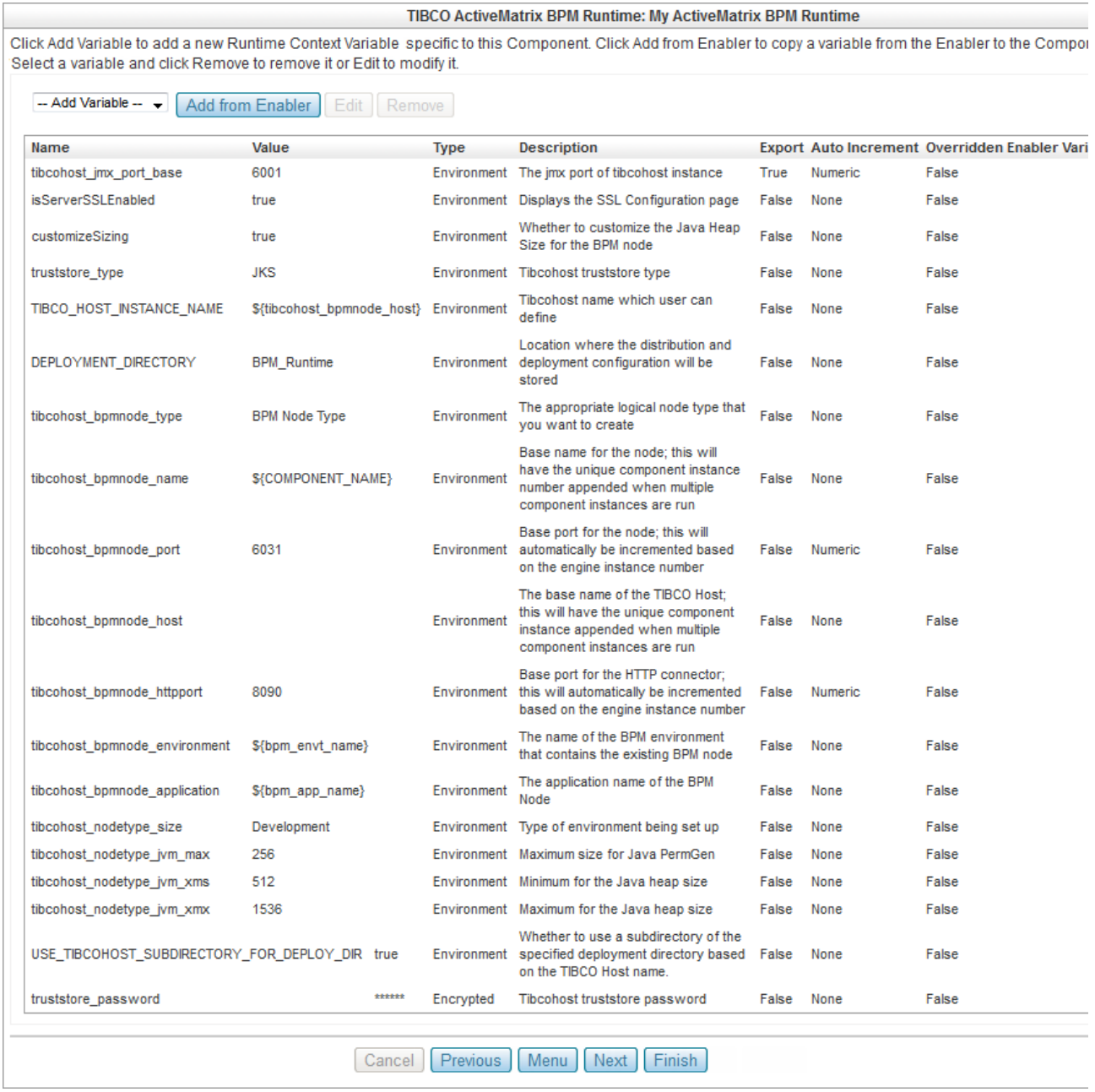
String, **Environment**, **System**, or **Encrypted** variables may be added and edited to the ActiveMatrix BPM Runtime component to define and set runtime specific context variables.

Select a variable type from the **Add Variable** pull-down list or **Add from Enabler** button to use a variable from a selected container.

Runtime context variables may be changed by selecting the variable (selected row is highlighted) and clicking **Edit** to change its attributes. Selected rows may also be removed.

Scroll all the way to bottom to find the Component Wizard navigation buttons.

Figure 19 Runtime Context Variables in ActiveMatrix BPM Runtime



TIBCO ActiveMatrix BPM Runtime: My ActiveMatrix BPM Runtime

Click Add Variable to add a new Runtime Context Variable specific to this Component. Click Add from Enabler to copy a variable from the Enabler to the Compo
Select a variable and click Remove to remove it or Edit to modify it.

-- Add Variable -- | Add from Enabler | Edit | Remove

Name	Value	Type	Description	Export	Auto Increment	Overridden Enabler Vari
tibcohost_jmx_port_base	6001	Environment	The jmx port of tibcohost instance	True	Numeric	False
isServerSSLEnabled	true	Environment	Displays the SSL Configuration page	False	None	False
customizeSizing	true	Environment	Whether to customize the Java Heap Size for the BPM node	False	None	False
truststore_type	JKS	Environment	Tibcohost truststore type	False	None	False
TIBCO_HOST_INSTANCE_NAME	${tibcohost_bpmnode_host}	Environment	Tibcohost name which user can define	False	None	False
DEPLOYMENT_DIRECTORY	BPM_Runtime	Environment	Location where the distribution and deployment configuration will be stored	False	None	False
tibcohost_bpmnode_type	BPM Node Type	Environment	The appropriate logical node type that you want to create	False	None	False
tibcohost_bpmnode_name	${COMPONENT_NAME}	Environment	Base name for the node; this will have the unique component instance number appended when multiple component instances are run	False	None	False
tibcohost_bpmnode_port	6031	Environment	Base port for the node; this will automatically be incremented based on the engine instance number	False	Numeric	False
tibcohost_bpmnode_host		Environment	The base name of the TIBCO Host; this will have the unique component instance appended when multiple component instances are run	False	None	False
tibcohost_bpmnode_httpport	8090	Environment	Base port for the HTTP connector; this will automatically be incremented based on the engine instance number	False	Numeric	False
tibcohost_bpmnode_environment	${bpm_envt_name}	Environment	The name of the BPM environment that contains the existing BPM node	False	None	False
tibcohost_bpmnode_application	${bpm_app_name}	Environment	The application name of the BPM Node	False	None	False
tibcohost_nodetype_size	Development	Environment	Type of environment being set up	False	None	False
tibcohost_nodetype_jvm_max	256	Environment	Maximum size for Java PermGen	False	None	False
tibcohost_nodetype_jvm_xms	512	Environment	Minimum for the Java heap size	False	None	False
tibcohost_nodetype_jvm_xmx	1536	Environment	Maximum for the Java heap size	False	None	False
USE_TIBCOHOST_SUBDIRECTORY_FOR_DEPLOY_DIR	true	Environment	Whether to use a subdirectory of the specified deployment directory based on the TIBCO Host name.	False	None	False
truststore_password	******	Encrypted	Tibcohost truststore password	False	None	False

Cancel | Previous | Menu | Next | Finish

Adding Allocation Rules

Adding, editing, and removing allocation rules is the same for both the TIBCO ActiveMatrix BPM Server and Runtime components. Refer to Add or remove log file patterns on page 60 for a description of the functionality.

Component Dependency Requirements

One allocation rule that is important for TIBCO ActiveMatrix BPM Runtime is the dependency on TIBCO ActiveMatrix BPM Server Component. The ActiveMatrix BPM Server component is also dependent on an instance of EMS and the database. These dependencies must be set. A component dependency may be set from the Stack definition, see After initially defining a Stack, you can still update it by adding or removing ActiveMatrix BPM Components and configuring other properties of the Stack on page 64, or you can set it directly on the component definition as is described here.

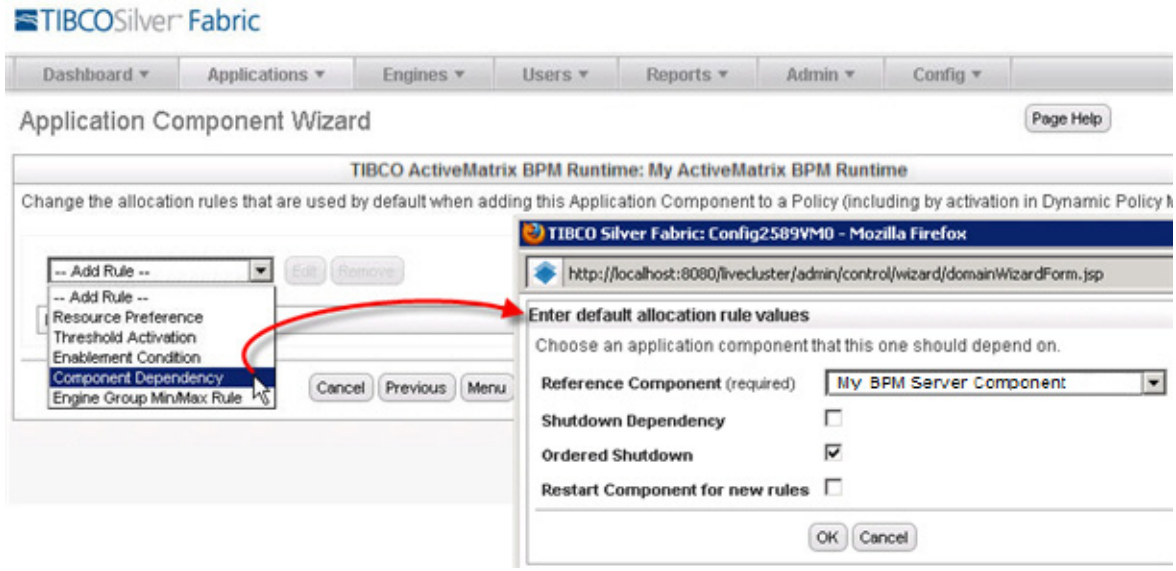
To set the component dependency:

1. Edit the ActiveMatrix BPM Runtime Component. Use the **Component Wizard** Menu button to show the page list. Select the "Add/edit default rule settings"

   Or instead of using the Menu page list, you could step through the Component Wizard until you reach the page that allows you to "Change the allocation rules that are used by default when adding this component to a policy (including by activation in Dynamic Policy Mode)." selection link.

2. In the **Add Rule** drop-down list select "Component Dependency".

Figure 20 Setting a Component Dependency



3. In the **Reference Component** drop-down list, select the name of the Component on which the dependence will be set to run inside your Stack.

The Reference Component name must match the value for Name specified in Configuring General Properties on page 8.

4. **Shutdown Dependency** - Clear the **Shutdown Dependency** check box when you want the TIBCO Hosts to continue running even if contact with the ActiveMatrix BPM Administrator is lost**.**
5. **Ordered Shutdown** should always be checked to provide for proper clean-up of allocated resources.
6. **Restart Component for new rules** should use the default setting.

Adding ActiveMatrix BPM Runtime Patches

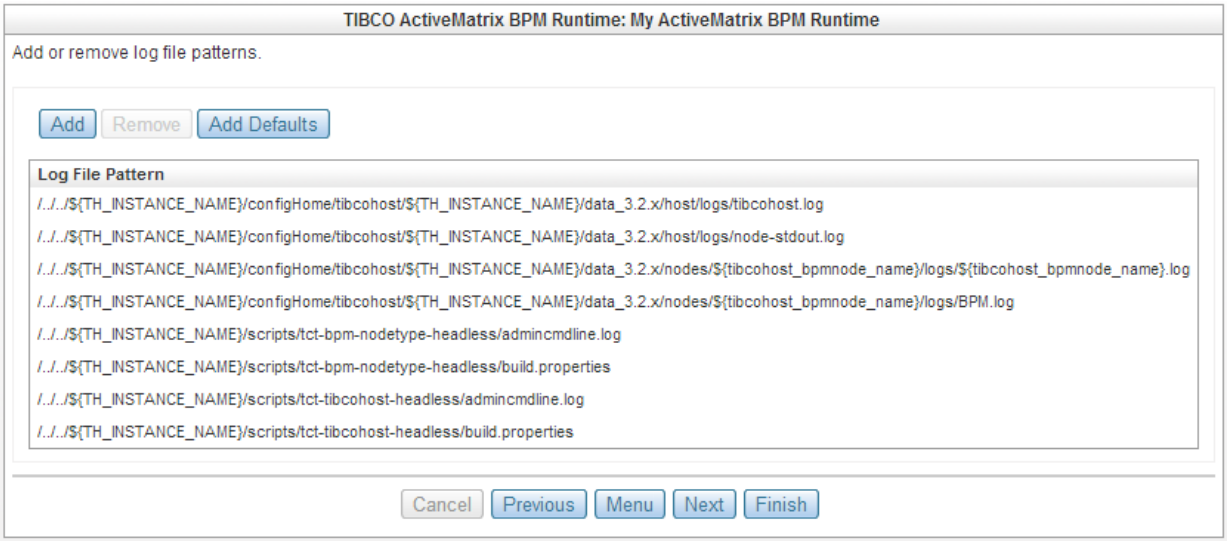
Patching TIBCO ActiveMatrix BPM Runtime is usually handled by patching the ActiveMatrix BPM Server instance. Some special conditions can apply. Refer to the section: Patching at the Component Level on page 45.

Patching can also occur at the Enabler container level. Refer to Applying a Service Pack or Hotfix to an existing BPM Server on page 76

Add or remove log file patterns

Log output directories may be added with paths that are relative to the engine instance work directory (work/host-instance). Path names can contain references to runtime context variables in the form `${variable_name}`.

Figure 21 Adding a Log File Pattern



The filename can be a regular expression, using conventions defined in javadoc for `java.util.regex.Pattern`. The rest of the path cannot be a regular expression.

Use a forward slash (/) for the directory separator character.

Log file patterns may also be removed by selecting the directory pattern first and then clicking the Remove button.

If you remove log file patterns, they will not be restored by the **Add Defaults** button. You can add them back manually at any time.

Component Wizard Generic Pages

Several of the Component Wizard pages are generic for all Silver Fabric enablers. Help buttons on these pages expose page specific documentation. Configurations for these generic settings are generally optional for the ActiveMatrix BPM Runtime component.

Refer to the page help and to *TIBCO Silver Fabric User's Guide and Administration Guide* for additional information on these configuration pages.

Finishing and Publishing a Component

You must click **Finish** to save a new component or to save any new changes to an existing component.

After you click the **Finish** button, make sure that the component is published to make it available when creating a Stack.

To do this, select **Publish Component** in the Actions drop-down list located on the same row as the component.

Creating an ActiveMatrix BPM Stack

Stacks are groups of components that are published to the Cloud together. Any number of components could be included in your Stack depending on your implementation, but minimally a TIBCO ActiveMatrix BPM Stack would have at least a TIBCO ActiveMatrix BPM Server Component and an instance of the TIBCO ActiveMatrix BPM Runtime Component. The TIBCO ActiveMatrix BPM Runtime Component depends upon an instance of the TIBCO ActiveMatrix Server Component.

Stack Dependency Requirements

Each ActiveMatrix BPM Runtime Component must have a component dependency set on one TIBCO ActiveMatrix BPM Server Component. You must set this component dependency for each of your ActiveMatrix BPM Components or those components without which it will not have enough information for proper publishing.

These steps are required. If you do not set the dependency for the ActiveMatrix BPM Runtime Component to depend on the ActiveMatrix BPM Server Component to be up and running before it starts, then the Runtime Component will fail to start.

Likewise the ActiveMatrix BPM Server Component requires an EMS and a Database to be up and running before it is started. Whether those dependencies are external resources or components made for direct allocation on TIBCO Silver Fabric dictates how the Stacks are put together for your implementation.

Components in a Stack can have:

- Component dependencies,
- Enablement conditions,
- Resource preferences,
- Threshold activations based on metrics,
- Engine group maximums and minimums,
- And many other configurable features.

Stacks and components can be published and unpublished programmatically.

The TIBCO Silver Fabric documentation describes these features and many more ways in which components and stacks work together. Refer to the TIBCO Silver Fabric documentation for more information.

Creating an ActiveMatrix BPM Stack with a Component Dependency

After you have created the components, you can create the stack so they can be published on Silver Fabric engines as a unit.

After setting the dependency and publishing this Stack, this ActiveMatrix BPM Runtime component will start after TIBCO ActiveMatrix BPM Server and Administrator are up and running.

1. In the TIBCO Silver® Fabric Administration Tool, select **Stacks > Stacks**.
2. Click the **Create New Stack** button.
3. Enter a Stack name.
4. In the components area, add one TIBCO ActiveMatrix BPM Server Component and one or more TIBCO ActiveMatrix BPM Runtime Components from the **Available Components** list to the **Selected Components** list.
5. In the Policies area, expand the component you just added to view the details of the component.
6. **Add a rule** by choosing a rule type on the left side of the ActiveMatrix BPM Runtime Component that you added. Choose "**Component Dependency**".
7. Using the **Depend on** drop down selector pick the ActiveMatrix BPM Server Component.
8. Select the **Ordered Shutdown** check box.

Shutdown Dependency - Because TIBCO ActiveMatrix BPM Administrator does not currently support the Fault Tolerant mode this setting will not apply.

Ordered Shutdown provides for a logical, sequential shutdown so that dependent components are shut down first. Ordered shutdown is especially important when the domain is hosted on a dependent database. When you have an ActiveMatrix Administrative Component that uses an external database, order of shutdown is less important.

Restart Component for new rules - If new rules are defined for a component that has already been published, it must be restarted for the new changes to be applied. If you wish to manually restart components later to propagate changes leave this box cleared.

If the ActiveMatrix BPM Server Component was configured to "*Use dependent EMS server*" then that dependency must be set here on the ActiveMatrix BPM Server Component also.

Pack by Host - Select this to specify that dependent components must run on the same host.

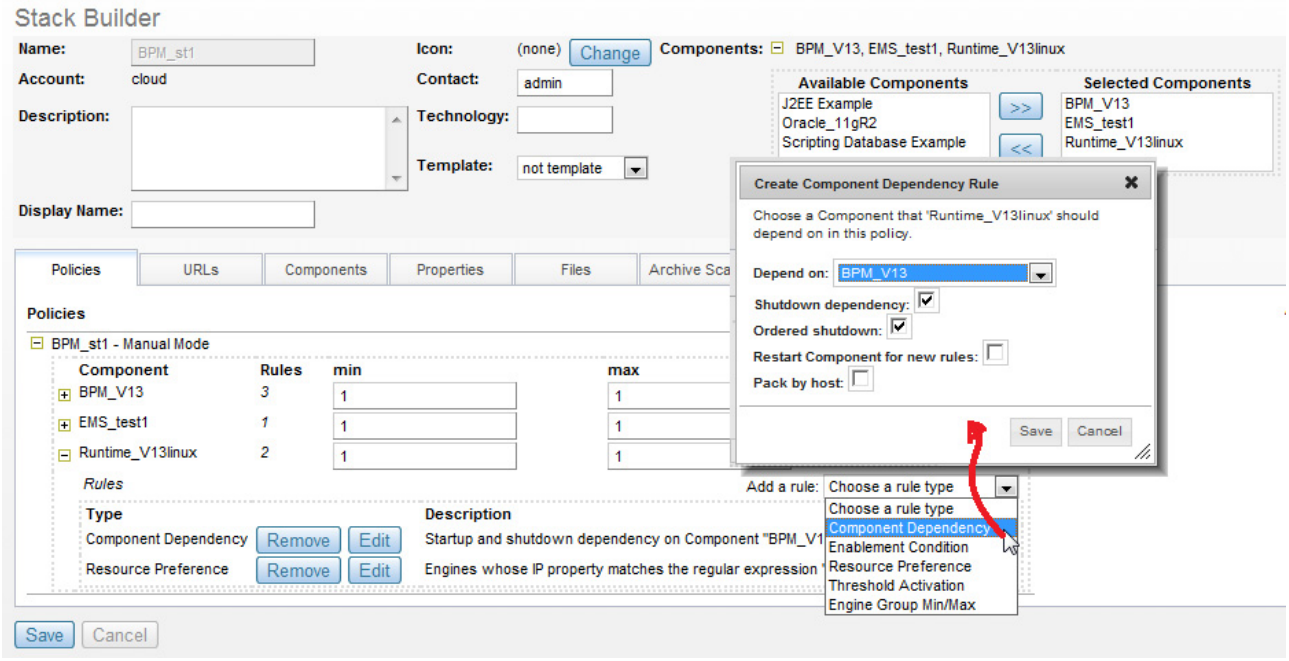
9. The *TIBCO Silver Fabric User's Guide* has more information on all of these settings.

10. **Save** the Component Dependency Rule and **Save** the Stack.

If you are using an EMS Enabler Component it should be added to the Stack and the ActiveMatrix BPM Server Component should get a component dependency rule that depends on the EMS Enabler Component so that EMS is up and running before the ActiveMatrix BPM Server Component is started.

A connection is supported between a component and one instance of the TIBCO ActiveMatrix BPM Administrator server. Connecting with more than one TIBCO ActiveMatrix BPM Administrator server is not supported.

Figure 22 Stack Builder page - Adding a Component Dependency



After initially defining a Stack, you can still update it by adding or removing ActiveMatrix BPM Components and configuring other properties of the Stack

Publish the Stack

Refer to the *TIBCO Silver Fabric User's Guide* for information on publishing and managing Stacks.

After your Stack is published with TIBCO ActiveMatrix BPM components refer to the *TIBCO ActiveMatrix BPM documentation* for information on management, operations, and maintenance of your ActiveMatrix BPM applications.

Configuring for Failover

TIBCO ActiveMatrix BPM Server running in BPM Server Only run mode and ActiveMatrix BPM Runtime Hosts can be configured to provide failover from one engine to another providing a secure transition from an unresponsive host machine.

Other ActiveMatrix BPM run modes running with ActiveMatrix Administrator are not supported for failover at this time.

Software pre-requisite: Failover configuration for TIBCO ActiveMatrix BPM instances requires the ability to create an TIBCO ActiveMatrix Administrator using TIBCO Silver® Fabric Enabler for ActiveMatrix® Service Grid and TIBCO ActiveMatrix® Service Grid Distribution for TIBCO Silver® Fabric version 3.3.0 Hotfix-003 or higher.

You must download and install this product according to the instructions in the *TIBCO Silver® Fabric Enabler for ActiveMatrix® Service Grid Installation Guide.*

To configure TIBCO ActiveMatrix BPM Hosts for failover:

Configure the components, stack and engines with the following settings to be prepared to move TIBCO ActiveMatrix BPM Hosts from one machine to another using the same TIBCO_HOME:

a. Create an ActiveMatrix Administrator Component with these settings:

— Use the TIBCO ActiveMatrix Service Grid Distribution for TIBCO Silver Fabric version 3.3.0 Hotfix-003 or higher for the ActiveMatrix Administrator Component.

— Enable Deployment Persistence.

b. Create an ActiveMatrix BPM Component using TIBCO ActiveMatrix BPM Distribution for TIBCO Silver Fabric.

— Set Deployment Directory to an accessible shared drive that let users from other machines to read, write, and execute.

c. Create TIBCO Silver Fabric Engines to make sure that the machines will use identical deployment directories.

— Mount the accessible shared drive defined in the ActiveMatrix BPM Component deployment directory with the same path on all Engines.

For example if the shared drive were `/mnt/shared` and the ActiveMatrix Administrator component was called `MyAMXComponent`, name the shared path: `/mnt/shared/MyAMXComponent`

`TIBCO_Home` would be in the directory:
`/mnt/shared/MyAMXComponent/<TIBCO_HOME>`

`ConfigHome` would be in the directory:
`/mnt/shared/MyAMXComponent/<ConfigHome>`

d. Create a Stack with the ActiveMatrix Administrator and ActiveMatrix BPM components according to the preconditions set above.

— Set Resource Preferences on the ActiveMatrix BPM Component so that it will run on a "Strongly Preferred" TIBCO Silver Fabric Engine that is prepared to act as TIBCO Host for ActiveMatrix BPM.

— Set the ActiveMatrix Administrator to run separately from ActiveMatrix BPM.

e. Save, publish and run the stack.

TIBCO ActiveMatrix BPM Hosts are monitored and polled by the enabler and any prolonged response failure first triggers attempts to reestablish a connection. If that fails, the component restarts, and if that fails, the TIBCO ActiveMatrix host starts on a different preferred host.

Configuring a Stack to Replicate an ActiveMatrix Administrator

Basic steps to create BPM Server component with ActiveMatrix Administrator Only mode and replication settings are as follows:

1. Create BPM Server component with **ActiveMatrix Administrator Only** mode. For more details, refer section Creating BPM Server Component with ActiveMatrix Administrator Only mode and Replication Settings.
2. Edit the engine daemon properties. For more details, refer section Editing the Engine Daemon Properties.
3. Create the stack for Administrator replication. For this set the value of **min** and **max** engine as `2` in the **Policies** tab. Add the resource reference rules that includes "Engines whose Group property `is` equal `to` `<group_name_set_in_step2>` are required", and " Engine whose instance property is equal to `<integer>` are Required. Refer Setting the Rules on page 70.

Creating BPM Server Component with ActiveMatrix Administrator Only mode and Replication Settings

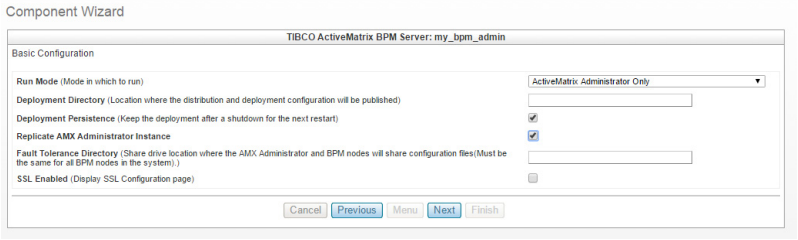
a. Configure a stack to run in Admin Replication mode by creating a TIBCO ActiveMatrix BPM Server component.

b. Select the Run Mode as **ActiveMatrix Administrator Only.**

Figure 23 ActiveMatrix Administrator Only mode

c. Select the **Replicate AMX Administrator Instance** check box.

Figure 24 Replicate AMX Administrator Instance check box selection



The **Fault Tolerance Directory** field gets enabled. The directory is the location of the share drive of the configuration files which are shared by the primary and replicated ActiveMatrix Administrator. This field is mandatory if you enable Replicate AMX Administrator instance. When replicating the ActiveMatrix Administrator, the Deployment Directory must be a local directory from the engine's host machine such as `/opt/<folder_name>/bpm`. The Fault Tolerance directory must be a shared drive folder accessible from the primary Administrator instance and its replica such as `/mnt/amx/share/`

You can select the **Use EMS Component Dependency** check box. It is not mandatory to use the EMS component, while it is suggested to use the EMS component.

Figure 25 Setting EMS Dependency

It is suggested to use external db and ldap servers when running AMX Administrator with replication mode. You must set the db and ldap information.

Admin only component, must set the value of "Maximum Engine Instances per Host (set to 0 for unlimited instances)" as "1" in the "Add/edit Application

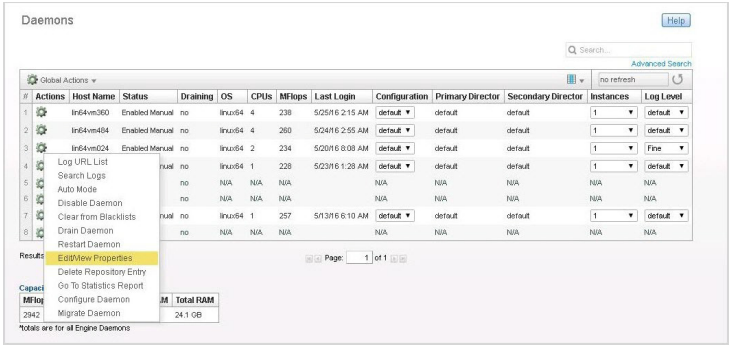
Component options" page. This ensures only one component instance can be activated on one daemon.

d. Click **Finish** to save your changes

Editing the Engine Daemon Properties

You can edit the Property of the Engine Daemons.

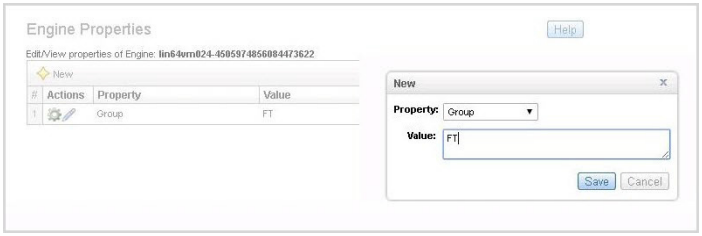
Figure 26 Editing or Viewing Properties



a. Pick up two engine daemons which will host the primary and replicated ActiveMatrix Administrator servers.

b. Edit the daemons by adding a Group property with same value say for example, **AMX-Admin-FT.** This creates a engine daemon group to run the ActiveMatrix Administrator with replication mode.

Figure 27 Setting Engine Properties



Setting the Rules

Perform the following steps to set the the policy rules in the stack:

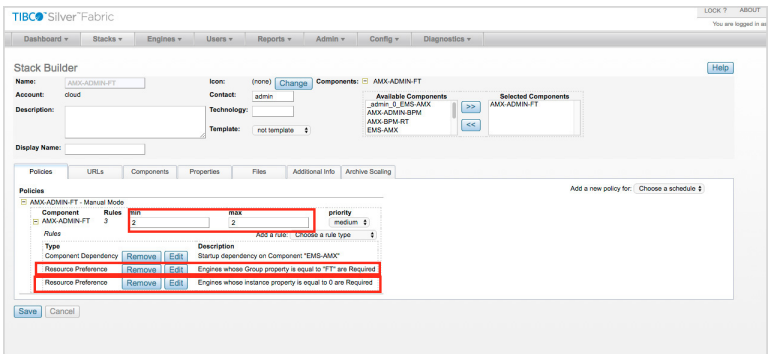
a. Set the min and max engine number to `2`.

b. Add a Resource Preference rule as "Engines whose Group property matches the regular expression `<Group_Name>` are Required"

c. Add a Resource Preference rule as "Engines whose instance property matches the regular expression `<Engine_Instance>` are Required".

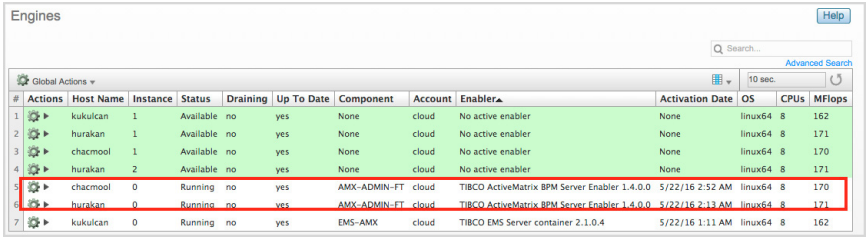
`<Group_Name>` is the group name set in step 2, `<Engine_Instance>` is the number of the engine instances. This information is required when there is more than one engine on each daemon. After the first allocation takes place for the ActiveMatrix Administrator Replica, you cannot move them to another engine instances. These rules make sure that the primary and replica ActiveMatrix Administrator instances are always running on the specific engine and daemon.

Figure 28 Setting the Rules



d. Publish and run the stack. The engines are deployed as highlighted in the figure below.

Figure 29 Deploying Engines



Configuring a Stack to Setup BPM Fault Tolerance

Setting of the complete BPM system with Fault Tolerance, perform the following steps:

a. Set **ActiveMatrix Administrator** with Fault Tolerance**.**

b. Create the **BPM Server Only** component with Fault Tolerance enabled. Set this shared drive with a different location than that of the **ActiveMatrix Administrator** replication mode.

c. Create BPM Runtime component with Fault Tolerance enabled, and set the same shared location as that of the **BPM Server Only** component.

d. Create a stack and set the component dependency as the **BPM Server Only** component depends on the ActiveMatrix Administrator Fault Tolerance component, and also the set BPM Runtime component depends on the **BPM Server Only** component.

Multitenancy Support

TIBCO Silver Fabric Enabler for ActiveMatrix BPM supports logical and physical separation of ActiveMatrix Administrator and ActiveMatrix BPM Server. Users can now set the component run mode to publish ActiveMatrix Administrator and ActiveMatrix BPM Server separately or together, to allow for more flexibility in deployment topologies which support multi-tenancy architectures.

Configuring Multi-Tenant Mode Using Component Dependencies

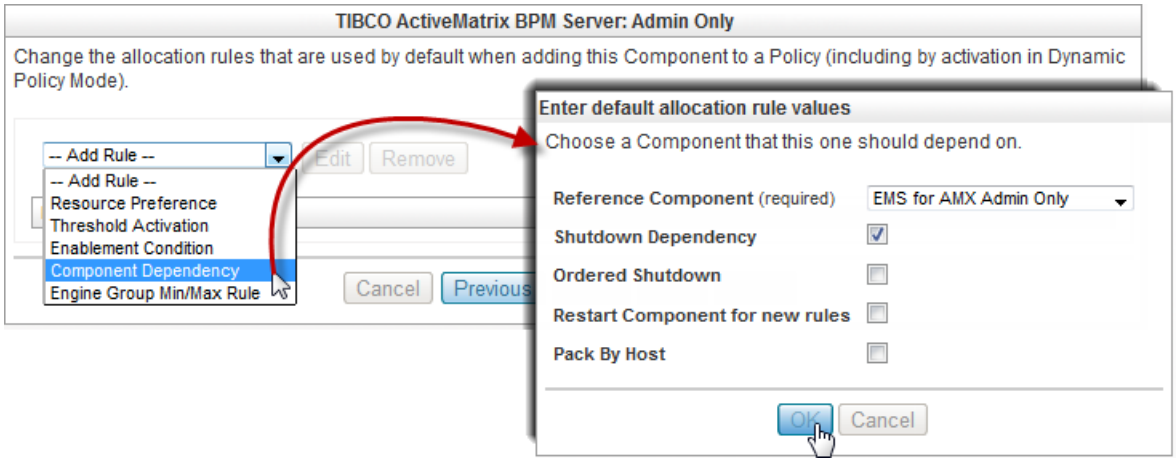
Task A Create a TIBCO ActiveMatrix Administrator Environment

1. Create a TIBCO EMS component for use by the ActiveMatrix Administrator .
2. Create a new TIBCO ActiveMatrix BPM Server Component and set the **Run Mode** as "ActiveMatrix Administrator Only".

Figure 30 Setting Run Mode as ActiveMatrix Administrator Only

3. Create a component dependency on the EMS server created in Step 1.

Figure 31 Setting a dependency on the EMS component



Upload the JDBC jars needed by the ActiveMatrix BPM nodes here. Also upload JDBC jars used for ActiveMatrix Administrator database connections.

ActiveMatrix BPM Servers require that the ActiveMatrix Administrator have a JDBC driver even if the Administrator is configured to use the in-process HSQL database. Failure to provide a supported JDBC database driver to the Administrator component will result in ActiveMatrix BPM Server failure to start.

4. Publish and deploy the EMS and Administrator components and the ActiveMatrix Administrator Environment is done.

Task B Creating Tenant Environments

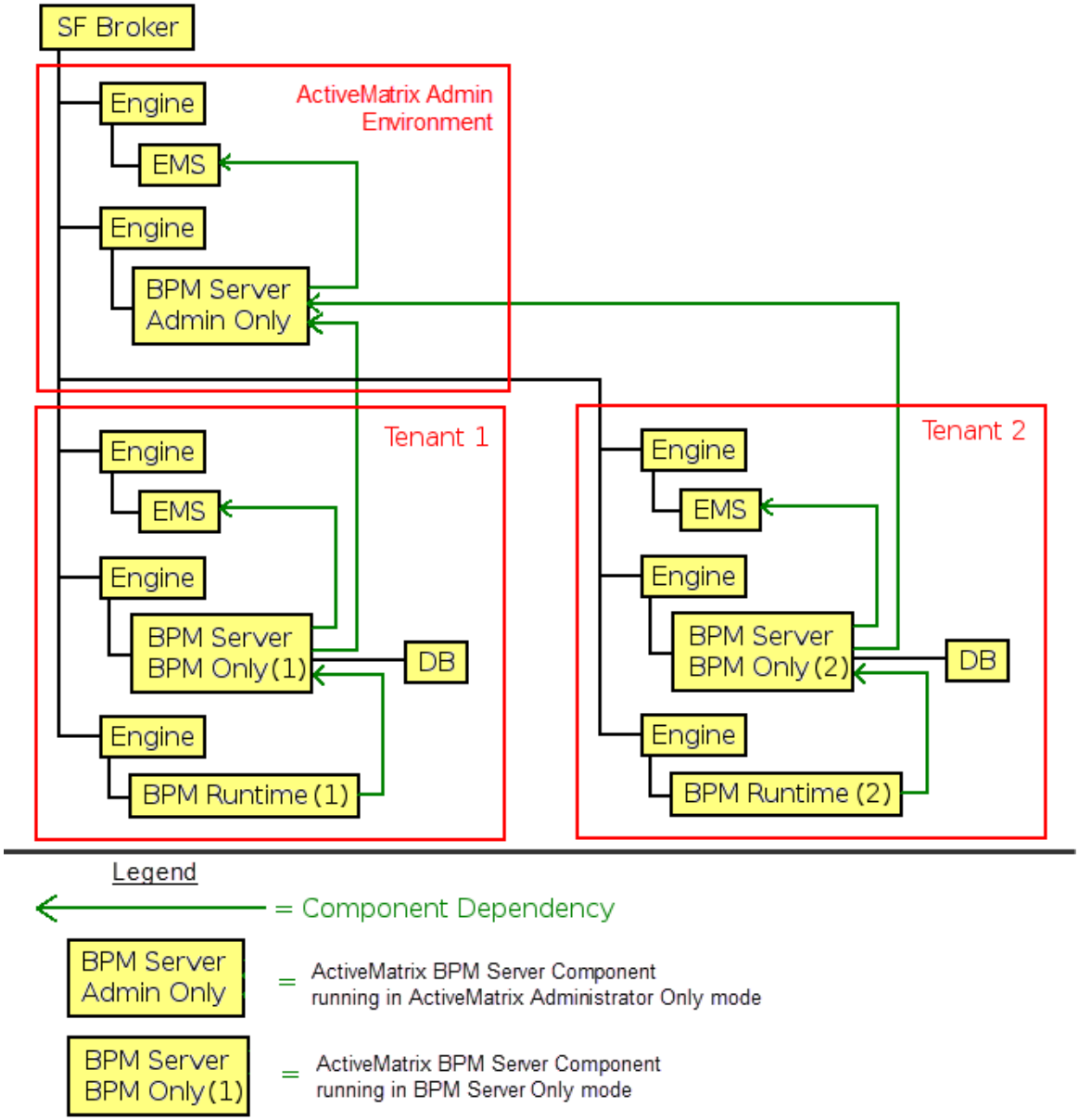
For each ActiveMatrix BPM environment you want to run follow these steps:

1. Create a TIBCO EMS component for use by the ActiveMatrix BPM Server.
2. Create an ActiveMatrix BPM Server component and set the **Run Mode** as "BPM Server Only".
3. Specify a unique environment name for each tenant environment.
4. Create a component dependency on the EMS server created in Step 1 of the Administrator environment (Task A).
5. Create a component dependency on the EMS server created in Step 1 of this task (Task B).

6. Make sure only one instance (min 1/max 1) of this component is configured to run in the stack.
7. Create one or more ActiveMatrix BPM Runtime components (of potentially different types - BPMNode, WorkManager, ProcessManager, WebComponents)

   The min/max number of instances of the Runtime node components can be arbitrary (can be greater than one).
8. Add a component dependency on the ActiveMatrix BPM Server added in Task B, Step 2.
9. Publish and deploy the components created in Task B, TIBCO EMS, ActiveMatrix BPM Server running in BPM Server Only mode, and any number of ActiveMatrix BPM Runtime nodes, as parts of a tenant environment stack and the tenant environment is done.

Diagram: Example of a Multi-Tenancy Topology



Each "tenant" represents an "Application" and "Environment" as displayed in the ActiveMatrix Administrator Enterprise.

Applying a Service Pack or Hotfix to an existing BPM Server

You may want to upgrade all of your existing TIBCO Silver Fabric instances of ActiveMatrix BPM Server with a Service Pack or Hotfix. This is not difficult, but the ActiveMatrix BPM Server must be running in persistent mode prior to upgrade so that neither the component nor Stack are restarted.

If instead you want to upgrade only selected components with a Service Pack, then you will want to refer Patching at the Component Level, page 45

Contact your TIBCO Support representative to get the appropriate gridlib archive files or download the appropriate Service Pack from the TIBCO eDelivery product download site before using this upgrade application procedure.

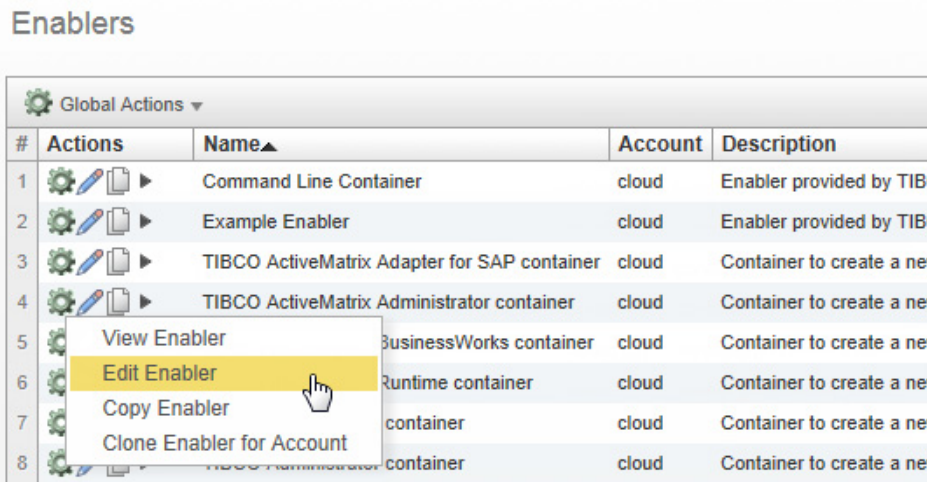
Prior to Enabler upgrade it is a good practice to back-up both the file system and the database as as described here in the *TIBCO ActiveMatrix BPM documentation*. Further, for best results, ensure that the TIBCO Host and ActiveMatrix BPM Component restart should be performed when the server is in a quiescent period to minimize any potential for in-flight messaging loss.

Applying a Service Pack or Hotfix to an Enabler for ActiveMatrix BPM Server (Runtime)

After you have the appropriate gridlib archives, put copies of those files onto the TIBCO Silver Fabric Broker.

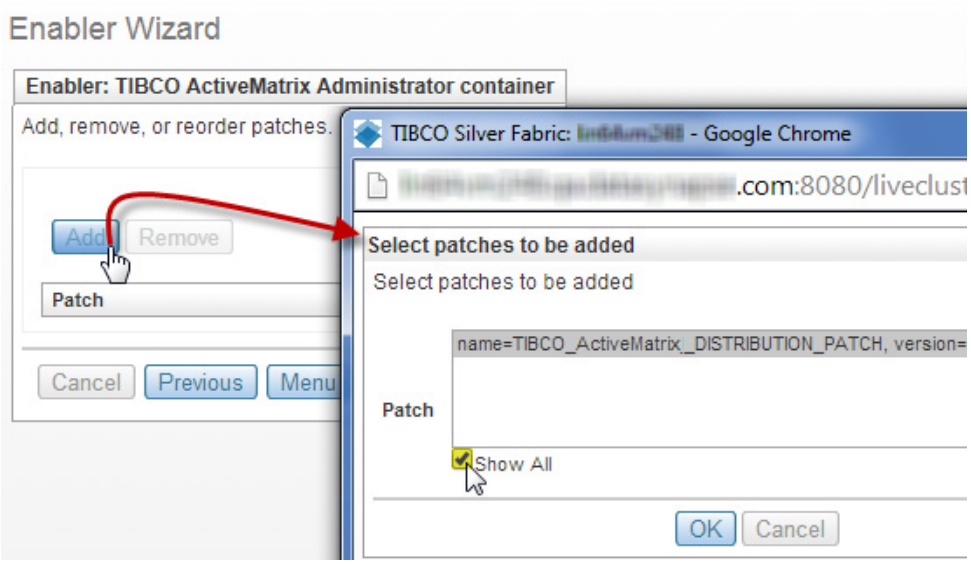
1. Paste Service Pack (or Hotfix) gridlib.zip archive into the broker directory:
   `SILVERFABRIC_HOME/webapps/livecluster/deploy/resources/gridlib/`
2. From the TIBCO Silver Fabric Administrator - Enablers page edit the TIBCO ActiveMatrix BPM Server Enabler.

Figure 32 Edit an Enabler from TIBCO Silver Fabric Administrator - Enablers page



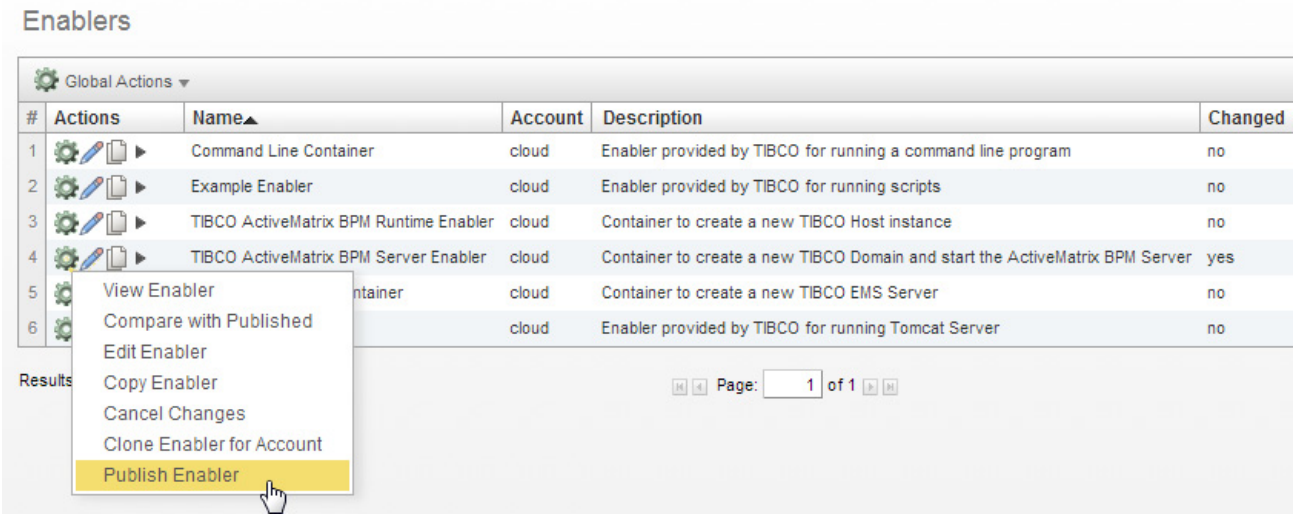
3. From the Enabler Wizard Menu select the option to "Add, remove, or reorder patches".
4. On the Add, remove, or reorder patches page click **Add** and a window opens to allow loaded patches to be selected.
5. On the "Select Patches to be added" window select the **Show All** check box to display the service pack or hotfix that was added to the broker in Step 1. Select the patch to be applied and click **OK**.

Figure 33 Adding a patch to the Enabler



6. Click **Finish** to save your changes to the enabler. The Enablers page is shown.
7. From the Enablers page **Publish** Enabler changes from the Actions menu.

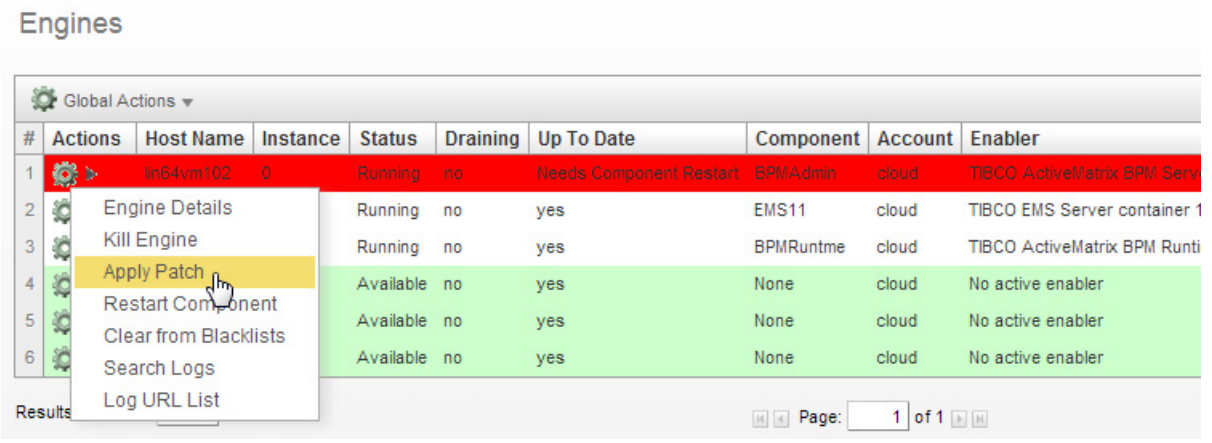
Figure 34 Publish the changed Enabler



8. Go to the **Engines** page. When the engine that is using an enabler without the upgrade is shown in red (amx.bpm.app in administrator should be in running state) then you are ready to apply the patch.

Use the Actions menu for the engine or engines you want to update and **Apply Patch** as is shown below.

Figure 35 Apply Patch to upgrade Engines running TIBCO ActiveMatrix BPM Server



After the patch has been applied (this can take significant amount of time due to upload delivery speeds into expanded archive directories) you can verify the upgrade was successful by checking the TIBCO Silver Fabric Engine log. The Engine log will show that the gridlib package is delivered and unzipped in the directory:

```
/[Engine_Home]/work/[Engine_instanceName]/fabric/patch-[index]
```

To ensure that component configuration or enabler upgrade changes are properly applied to existing components published and running on TIBCO Silver Fabric, the TIBCO Host and components must be restarted.

Prior to Enabler upgrade or component restart, make sure to back-up both the file system and the database as is described in the *TIBCO ActiveMatrix BPM documentation*. Further for best results , ensure that the TIBCO Host and ActiveMatrix BPM component restart should be performed when the server is in a quiescent period to minimize any potential for in-flight messaging loss.

Changing the Component Enabler

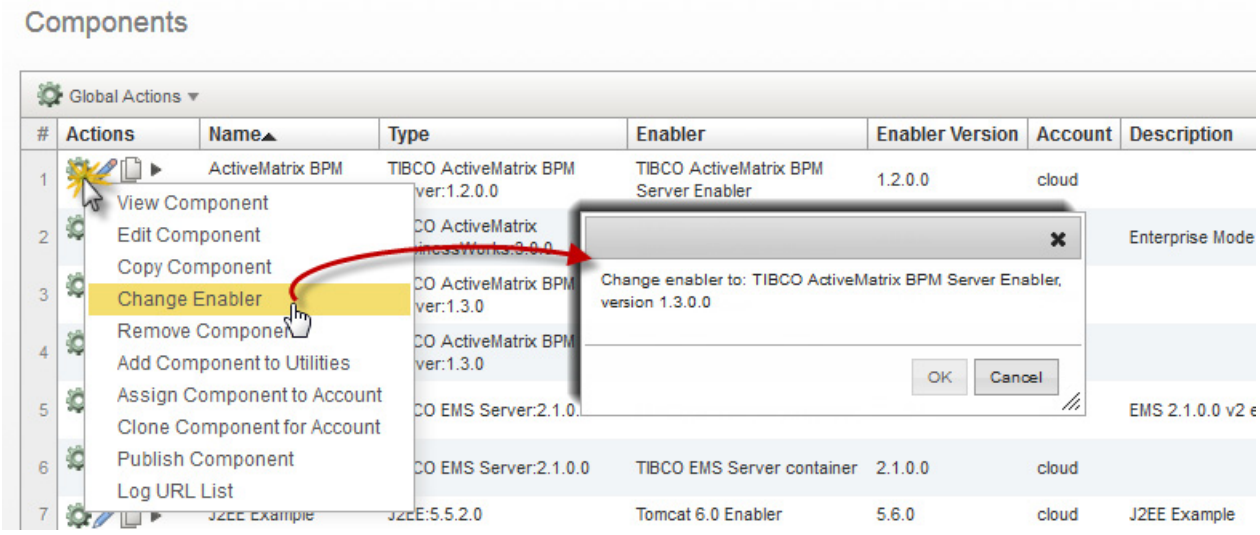
Upgrading a Component created with TIBCO Silver Fabric Enabler for ActiveMatrix BPM release version 1.2 to 1.3 is relatively easy.

Back up your engines and consider timing for when you wish to perform a component restart such that, brief operational downtime has minimal impact.

1. On the **TIBCO Silver Fabric Administrator > Components** page identify the TIBCO Silver Fabric Enabler for ActiveMatrix BPM component you wish to upgrade from release version 1.2 of the enabler. The **Enabler Version** column will make it easy to identify out of date Enablers.

   Click the Component Actions menu icon for the component you will update and choose **Change Enabler**. Select the enabler version upgrade target and click **OK**.

Figure 36 Upgrading a Component - Change Enabler



2. Click the Component Actions menu icon again, click **Publish Changes**, and then click **OK** to publish the selected component.
3. Switch to the Engines page and those stacks that used the upgraded component will appear in red if they require a component restart.

   Click the Engine Actions menu icon on the row that needs a component restart. Click **Restart Component**.

Chapter 3 VirtualRouter with TIBCO ActiveMatrix BPM Administrator

TIBCO Silver Fabric VirtualRouter acts as a service lookup to properly route initial requests from Web clients for virtualized HTTP enabled components running Web applications and Web services on the Silver Fabric Cloud.

Overview

In a private cloud, when you run an ActiveMatrix BPM Server and TIBCO ActiveMatrix BPM Administrator (among other HTTP-enabled tools), you do not necessarily know in advance the address of the machine where it will run unless you set resource preferences in the rules.

The VirtualRouter resolves that connection difficulty by routing web service requests to the correct TIBCO ActiveMatrix BPM machine or IP address when you attempt connection through the VirtualRouter.

By default, every Silver Fabric Broker includes an installation of the VirtualRouter running on the same application server.

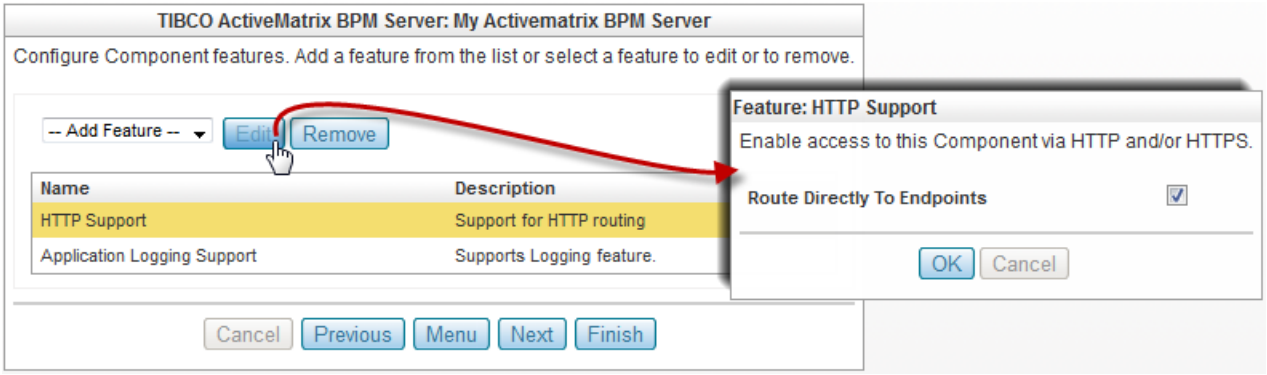
According to the Silver Fabric documentation:

> You can run additional instances of VirtualRouter externally, on other application servers. This provides greater overall performance, as VirtualRouter can handle greater loads without affecting general Broker performance. It also provides greater system stability in the event of a Broker failure, as VirtualRouter continues to function with its last known state. Multiple VirtualRouters may be run for redundancy and load sharing. They can also be partitioned such that some VirtualRouters only forward to a certain set of components and other VirtualRouters to other components.

Refer to the *TIBCO Silver Fabric documentation* for more details on these more complex uses of VirtualRouter.

VirtualRouter and TIBCO ActiveMatrix BPM HTTP Enabled Tools

Virtual Router forwards or redirects URL requests sent to the Silver Fabric Broker IP address to the appropriate TIBCO Silver Fabric host machine. By default, Virtual Router will route requests directly to endpoints, but if your implementation requires forwarding to endpoints you can change the HTTP Support setting for the component on the Add/Edit Application Component Features page as is shown here:



The URL for the HTTP web client request depends on these parameters:

- ***BrokerMachineName*** **-** The machine name or IP address where you installed the TIBCO Silver® Fabric Broker and TIBCO VirtualRouter.
- ***BrokerPort*** **-** The port of the Silver® Fabric Administrator GUI. The default value is `8080`.
- ***Admin_Enterprise_Name*** -The value of the Enterprise Name you entered when you configured TIBCO ActiveMatrix BPM Server component. Spaces that are present in the enterprise name are converted to underscores.
- ***BPM_Application_Name*** -The value of the BPM Application Name you entered when you configured TIBCO ActiveMatrix BPM Server component. Spaces in the ActiveMatrix BPM Application name are converted to underscores for VirtualRouter usage.

VirtualRouter and TIBCO ActiveMatrix BPM Administrator

The following URL forwards or redirects to TIBCO ActiveMatrix BPM Administrator:

`http://`*`BrokerMachineName`*`:`*`BrokerPort`*`/`*`Admin_Enterprise_Name`*`/`**`amxadministrator/`**

For example: `http://10.107.172.95:8080/N1/amxadministrator/`

The following addresses are also externally valid in a default configuration:

*http://BrokerMachineName:BrokerPort/Admin_Enterprise_Name/***<bpm_app_name>/openspace/openspace.html**

*http://BrokerMachineName:BrokerPort/Admin_Enterprise_Name/***<bpm_app_name>/workspace/workspace.html**

By default when **Route Directly to Endpoints** is true, on initial request these HTTP request URLs will be redirected with an HTTP 302 redirect response to the client. This will point to the actual endpoint, the correct machine name, and the port number where TIBCO ActiveMatrix Administrator or the requested service runs.

After the connection with the actual machine where the HTTP enabled component is running the browser URL will show the machine name or IP address of the host machine and not that of the VirtualRouter.

VirtualRouter and TIBCO® Openspace

Openspace is a web application that enables hosting of gadgets and widgets, self-contained blocks of dynamic web content. This application is hosted on BPM Server nodes as well as BPM Runtime instances configured with the "BPM" or "WebComponents" node types.

The following URL forwards or redirects to TIBCO Openspace.

```
http://BrokerMachineName:BrokerPort/Admin_Enterprise_Name/BPM_Application_Name/openspace/openspace.html
```

VirtualRouter and TIBCO® Workspace

Workspace manages BPM business processes, services, work items, and more. This application is hosted on BPM Server nodes as well as BPM Runtime instances configured with the "BPM" or "WebComponents" node types.

The following URL redirects to TIBCO Workspace.

```
http://BrokerMachineName:BrokerPort/Admin_Enterprise_Name/BPM_Application_Name/workspace/workspace.html
```

Chapter 4 Upgrade Scenarios

The following sections describe the most common TIBCO Silver Fabric Enabler for ActiveMatrix BPM upgrade scenarios.

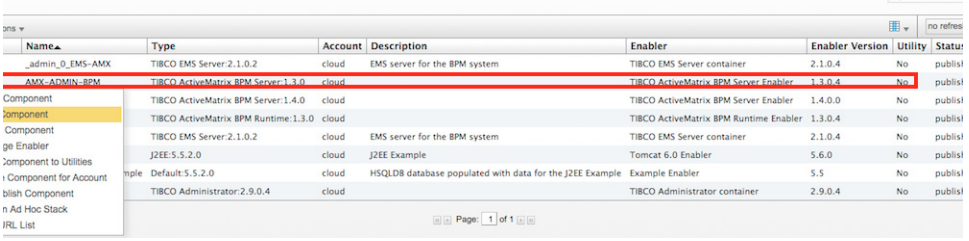
The following are the basic configuration before upgrade:

1. Enable Admin API role in the broker and give component publish permission.
2. TIBCO ActiveMatrix® Service Grid Distribution for TIBCO Silver® Fabric 3.3.0HF12 or above is required while upgrading to TIBCO ActiveMatrix® BPM Distribution for TIBCO Silver® Fabric 4.x.
3. Add a Shared drive folder. For more details, refer to the section Adding a Shared Drive Folder in TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.3.0 for Distributing Applications to Remote BPM Nodes

Steps for upgrading an existing TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.3.0 running TIBCO ActiveMatrix® BPM Distribution for TIBCO Silver® Fabric 2.2.0 to TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.4.0 and TIBCO ActiveMatrix® BPM Distribution for TIBCO Silver® Fabric 4.0 :

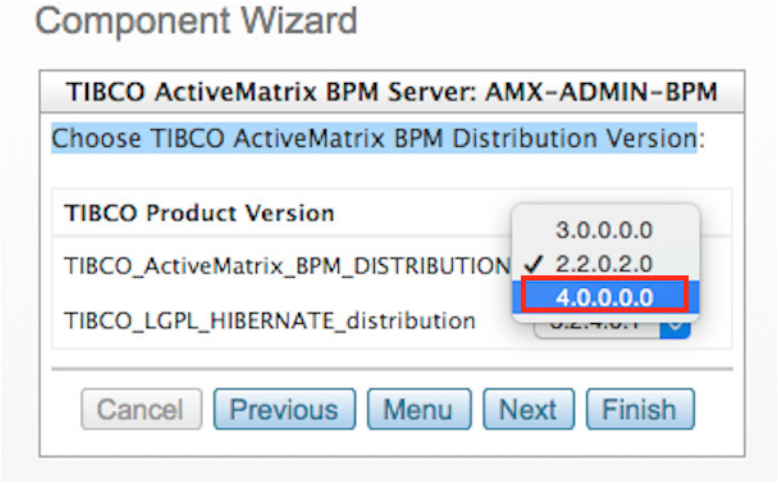
1. Using the **Change Enabler** option, the existing TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.3.0 components must be upgraded to enabler version 1.4.0
2. Click **Edit Component** to edit the existing TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.4.0 components.

Figure 37 Edit Component for Upgrading



3. In the Choose TIBCO ActiveMatrix BPM Distribution Version page, select the BPM distribution version that requires to upgrade.

Figure 38 Select the Distribution Version for the Upgrade



4. Click **Finish** and publish the changes.

To upgrade an existing BPM System, it is necessary to restart all of the BPM Nodes and ActiveMatrix Administrator in an ordered manner. To upgrade the complete BPM system and to prevent any downtime, perform the steps listed in the following example.

Suppose you want to upgrade two components to latest TIBCO ActiveMatrix® BPM Distribution for TIBCO Silver® Fabric 4.0.0 product:

1. Make sure that all components are upgraded to the latest TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.4.0 version.
2. Edit the component distribution and select the 4.0.0.0 version. If you want to transform the BPM nodes to run in a distributed mode, add the following runtime variables before upgrading.

Table 2 Runtime variables

Variable	Value	Description
`FT_DIRECTORY`	`${FT_DIRECTORY}`	Indicates the share folder where the configuration files are stored. eg: /mnt/amx
`FT_ENABLED`	`True`	Indicates whether the BPM node should run in the distributed mode and use the share drive folder from the `${FT_DIRECTORY}`

3. **Save** the changes and **Publish** them.

4. After the changes are published, restart the components from the engine list in the following order:

Table 3

Order	Component	Running Mode	Description	Expected Output after restart
1	BPM Server	Running in **ActiveMatrix Administrator and BPM Server mode**	According to BPM, the ActiveMatrix administrator is the first component that must be upgraded.	TIBCO HOST and ActiveMatrix Administrator are upgraded to BPM 4.0.0 The component turns red again after the first restart, which means a restart is required again.
2	BPM Runtime	Running in **BPM Node Type**	After the BPM Server Node host is upgraded, all of the remaining BPM Node Types (BPM Runtime enabler) must be upgraded next.	TIBCO HOST is upgraded.
3	BPM-Runtime	If there is more BPM Runtime nodes	Restart all of the remaining BPM Node Types (order between BPM Nodes types does not matter).	TIBCO HOST is upgraded.
4	BPM Server	Running in **ActiveMatrix Administrator and BPM Server** mode	After the first restart, the full BPM system must still proceed to upgrade BPM nodes and BPM applications. This must be done from the machine hosting the ActiveMatrix Administrator, after all of the remote BPM nodes are upgraded and so a second restart is required. This is the most critical step as applications are stopped in nodes and upgraded.	BPM system and BPM nodes should be upgraded to the latest version.

For more information on installation and upgrade scenarios and upgrading from an earlier version of ActiveMatrix BPM, refer https://docs.tibco.com/pub/amx-bpm/4.0.0/doc/pdf/TIB_amx-bpm_4.0_installation.pdf

Adding a Shared Drive Folder in TIBCO Silver® Fabric Enabler for ActiveMatrix® BPM 1.3.0 for Distributing Applications to Remote BPM Nodes

To distribute applications to remote BPM nodes from ActiveMatrix Administrator GUI, you must create a shared drive folder. This is done to upgrade BPM.

You must execute this steps if you have existing SFBP 1.3.0 components running.

This is done, because SFBP 1.3.0 does not have a field for the share drive folder in the UI. This folder is required for the BPM nodes to distribute the applications.

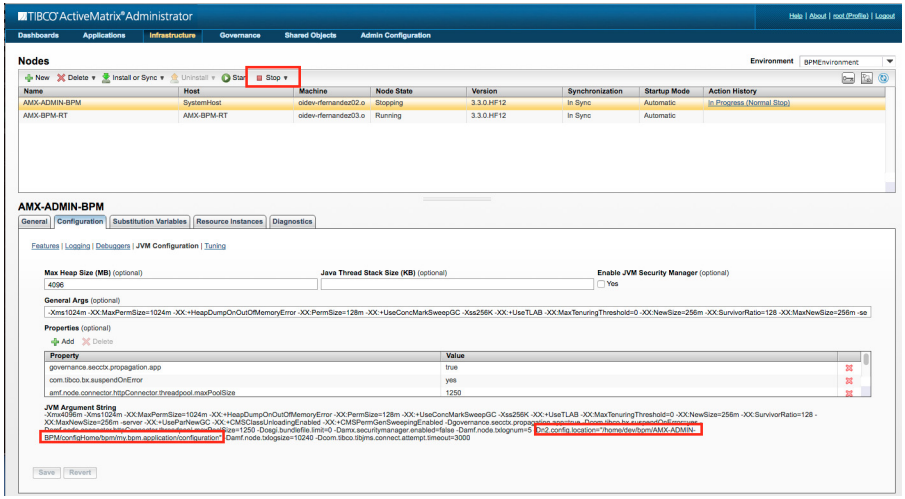
To enable the BPM nodes to run in distributed mode do the following:

1. Log in to your ActiveMatrix Administrator using the user name and password.

Figure 39 Log in to TIBCO ActiveMatrix Administrator

2. Click **Infrastructure** to view the **Nodes** pane.

Figure 40 TIBCO ActiveMatrix Administrator



3. Click **Stop** in the **Nodes** pane, to stop the node.
4. Copy the files from `${DEPLOYMENT_DIR}/configHome/bpm/${bpm_app_name}/configuration` from the main BPM Server node to the new share drive `${FT_DIRECTORY}/bpm/{bpm_app_name}/configuration`.

   Example:

   Move `/opt/AMX-BPM/configHome/bpm/my.bpm.app/configuration` to `/mnt/amx/bpm/my.bpm.app/configuration`

   Here, `${bpm_app_name}` is the value of a runtime variable. By default it is the component name say for example `my.bpm.application`
5. Click a node and then in the **Configuration** tab, click **JVM Configuration** and locate the General Args text box, as highlighted in the previous image.
6. Change the `Dn2.config.location` property value to `${FT_DIRECTORY}/bpm/{bpm_app_name}/configuration`.

   For example: `/mnt/amx/bpm/my.bpm.app/configuration`
7. **Save** the changes.
8. Click **Install or Sync** tab in the **Nodes** pane.
9. Click **Start** tab in **Nodes** pane, to start the node. Now the BPM node is in distributed mode.

Repeat the above steps for every node in the BPM environment, except copying the files from the main node. The copying of files must be done only once.

Index

M

N

O

P

R

S

T

U

V

W

X