TIBCO ActiveMatrix® Adapter for JD Edwards EnterpriseOne

Concepts

Software Release 6.0
August 2013

Two-Second Advantage®

Important Information

Contents

Figures

Preface

TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne allows one-way (publish or subscribe) or two-way (request-response) message exchange between TIBCO applications and the JD Edwards EnterpriseOne enterprise resource planning (ERP) system.

This document explains the basic concepts and supported features of TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne.

In this document, TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne is referenced as TIBCO ActiveMatrix Adapter for JDE, and JD Edwards EnterpriseOne is referenced as JDE.

Topics

Related Documentation

This section lists documentation resources you may find useful.

TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Documentation

The following documents form the TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne documentation set:

- *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Concepts* Read this manual to gain an understanding of the product that you can apply to the various tasks you may undertake.
- *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Installation* Read this manual for instructions on site preparation and installation.
- *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Configuration and Deployment* Read this manual for instructions on how to create, configure, and deploy adapter projects.
- *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Examples* Read this manual to work through the examples provided with the adapter.
- *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne 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

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO ActiveMatrix BusinessWorks™
- TIBCO Adapter® SDK
- TIBCO Administrator™
- TIBCO Designer™
- TIBCO Enterprise Message Service™
- TIBCO Hawk®
- TIBCO Rendezvous®
- TIBCO Runtime Agent™

Third-Party Documentation

You may also find it useful to read the JDE documentation on the following web site:

- http://www.oracle.com/technetwork/documentation/jdedent-098169.html

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
ENV_NAME *TIBCO_HOME*	TIBCO products are installed into an installation environment. A product installed into an installation environment does not access components in other installation environments. Incompatible products and multiple instances of the same product must be installed into different installation environments. An installation environment consists of the following properties: • **Name** Identifies the installation environment. This name is referenced in documentation as *ENV_NAME*. On Microsoft Windows, the name is appended to the name of Windows services created by the installer and is a component of the path to the product shortcut in the Windows Start > All Programs menu. • **Path** The folder into which the product is installed. This folder is referenced in documentation as *TIBCO_HOME*.
ADJDEXE_HOME	*TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne* installs into a directory within a *TIBCO_HOME*. This directory is referenced in documentation as *ADJDEXE_HOME*. The default value of *ADJDEXE_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco\adapter\adjdexe\6.0.`
`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 BusinessWorks 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 name 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.

Table 2 Syntax Typographical Conventions

Convention	Use
`[ ]`	An optional item in a command or code syntax. For example: `MyCommand [optional_parameter] required_parameter`
`|`	A logical `OR` that separates multiple items of which only one may be chosen. For example, you can select only one of the following parameters: `MyCommand param1 | param2 | param3`

Table 2 Syntax Typographical Conventions (Cont'd)

Convention	Use

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

Connecting with TIBCO Resources

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts. It is 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 TIBCO Documentation

You can access TIBCO documentation here:

http://docs.tibco.com

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, 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 Adapter Introduction

This chapter introduces basic concepts of adapters.

Topics

Adapter Overview

Many businesses use different applications from different vendors to implement their business processes and to manage information. Applications from different vendors need to be integrated with each other to exchange information. However, vendors typically have their own ways to format and expose data. Therefore, integrating the various applications across an enterprise poses significant challenges.

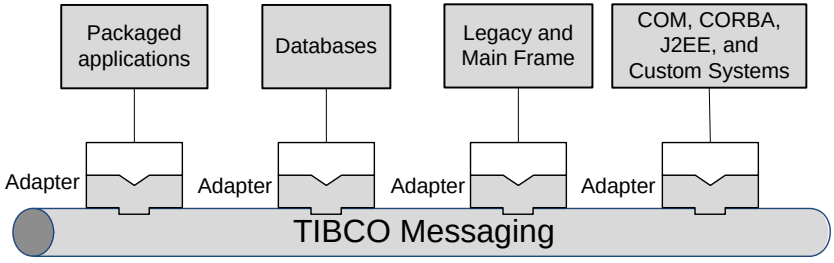
An adapter provides a bridge between an application and the TIBCO integration platforms. TIBCO adapters enable packaged applications, databases, and other technologies to participate actively in the enterprise information flow, regardless of their data formats or communication protocols. Integration of new applications does not require programming and does not interfere with existing infrastructure.

Adapters encapsulate the complex interaction patterns into a set of standard interaction patterns which are common to all adapters. This makes adapters easy to be understood and administered.

Adapters isolate applications from complex interaction and enable various applications to be part of TIBCO infrastructure without requiring any changes to the applications. Once data is published on TIBCO infrastructure, messages can be transformed and business process automation can be implemented across all applications.

As shown in Figure 1, adapters allow for the exchange of data among different technologies.

Figure 1 Adapters Provide a Bridge for Data



Adapter Services

Adapters are responsible for making information from different applications available to other applications across an enterprise. To do so, an adapter is configured to provide one or more services.

Table 3 summarizes the services introduced in this section.

Table 3 Adapter Services Summary

Service	Initiator	Target	Event Mode
Publication Service (sends to target)	Vendor application	TIBCO infrastructure	Asynchronous
Subscription Service (gets from initiator)	TIBCO infrastructure	Vendor application	Asynchronous
Request-Response Service (gets from initiator, waits for response then sends response to target)	TIBCO infrastructure	Vendor application	Synchronous
Request-Response Invocation Service (sends to target, waits for response, then sends response to initiator)	Vendor application	TIBCO infrastructure	Synchronous

Publication Service

An adapter *publication service* recognizes when business events happen in a vendor application, and asynchronously sends out the event data in real-time to interested systems in the TIBCO environment.

For example, an adapter can publish an event each time a new customer account is added to an application. Other applications that receive the event can then update their records just as the original application did.

Subscription Service

An adapter *subscription service* asynchronously performs an action, such as updating business objects or invoking native APIs, on a vendor application. The adapter service listens to external business events, which trigger the appropriate action.

Referring to the previous example, an adapter subscription service can listen for customer record creation events (happening in an application and published to the TIBCO infrastructure) and update another application.

Request-Response Service

An adapter *request-response service* is used to extend the integration process to the vendor application. In addition to asynchronously publishing and subscribing to events, an adapter can be used for synchronously retrieving data from or executing transactions within a vendor application. After the action is performed in the vendor application, the adapter service sends a response back to the requester with either the results of the action or a confirmation that the action occurred. This entire process is called *request-response*, and it is useful for actions such as adding or deleting business objects.

For example, an adapter receives a request message from the TIBCO infrastructure and sends it to an application. The adapter gets a response from the application and returns it to TIBCO infrastructure.

Request-Response Invocation Service

An adapter *request-response invocation* service is similar to the request-response service, except that the roles are reversed. The vendor application is now the requester or initiator of the service, instead of the provider of the service. The adapter service acts as a proxy, giving the vendor application the ability to invoke services on TIBCO infrastructure. TIBCO infrastructure may perform a series of steps to complete the requested service, including invoking services on other applications through TIBCO infrastructure and other adapters.

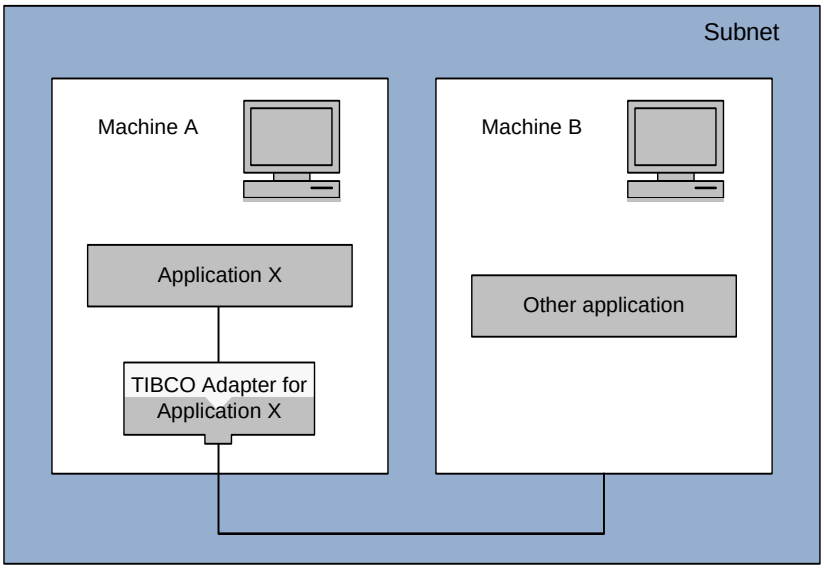
For example, the adapter sending a request message from application Y to application X. After it processes the message, it is returned to the adapter, which sends the response back to application.

Choosing an Adapter Service

A business integration scenario drives the choice of one adapter service or another. This section provides a simple flow chart that helps you to choose the service to use.

Consider the following environment that involves application X, an adapter, and another application:

Figure 2 A Business Integration Scenario



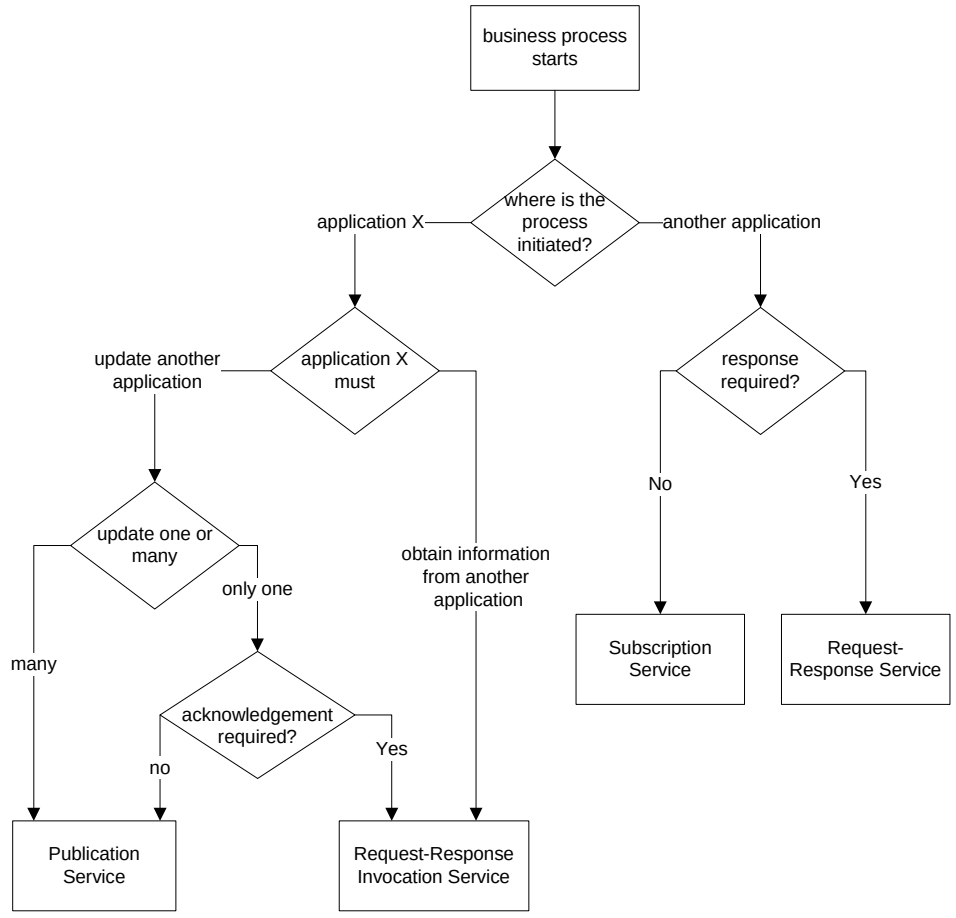
In this scenario, data is exchanged between the application X and another application. The other application could be a customer management system, such as PeopleSoft, or another TIBCO application, such as TIBCO ActiveMatrix BusinessWorks.

To choose the adapter service to use, start by finding out where the scenario begins or what triggers it.

For example, when a new customer account is created in application X, must the account information be propagated through the adapter to the other application? Or does a batch business process in TIBCO ActiveMatrix BusinessWorks need information from application X to generate a report?

This question is the starting point of the decision chart provided in Figure 3.

Figure 3 Choosing an Adapter Service



Working through the decision chart, if the business process is the creation of a customer record in application X and if many other applications need to be updated when the event occurs, but no acknowledgements are required, the publication service should be used.

Adapter Life Cycle

In general, the life cycle of an adapter includes these stages: installation, migration, configuration, deployment, starting, monitoring, undeployment, stopping, and uninstallation.

Installation

The installation stage includes installing the adapter and other software from TIBCO on which the adapter depends.

For many adapters, you do not have to install the adapter and vendor application on the same machine, while you must install TIBCO Runtime Agent on each computer that runs the adapter.

Migration

If you have previous releases installed, you may need to upgrade the adapter to the latest version. See the Migration and Compatibility section in the adapter's release notes for more information.

Configuration

In the configuration stage, an adapter instance can be created and configured with TIBCO Designer. The configuration information is required for a runtime adapter to interact with the vendor application and other applications.

Deployment, Starting, Undeployment, and Stopping

An adapter instance created by TIBCO Designer can be deployed, started, undepolyed, and stopped using TIBCO Administrator or TIBCO ActiveMatrix Administrator.

- TIBCO Administrator

  TIBCO Administrator is used to manage adapter services.

- TIBCO ActiveMatrix Administrator

  In this case, you must first import the Designer project into TIBCO Business Studio using TIBCO ActiveMatrix Binding Type for Adapters or convert the EAR file to a service assembly using TIBCO ActiveMatrix Proxy Implementation Type.

Monitoring

In this stage, use one of the following tools to manage and monitor the adapter:

- the built-in monitoring tools provided by TIBCO Administrator or TIBCO ActiveMatrix Administrator
- the TIBCO Hawk microagents

Uninstallation

The uninstallation stage includes uninstalling the adapter and other software from TIBCO on which the adapter depends.

Chapter 2 Adapter Infrastructure Tools

This chapter introduces the TIBCO infrastructure tools that work with an adapter.

Topics

TIBCO Runtime Agent

The TIBCO Runtime Agent provides basic connectivity between the adapter and other TIBCO infrastructure tools. TIBCO Runtime Agent is required on any machine on which an adapter is installed. TIBCO Runtime Agent runs on each machine on which an adapter runs and executes scripts, sends alerts, and performs recovery as specified.

TIBCO Runtime Agent has two main functions:

- Supplies an agent that runs in the background on each machine.
  - The agent is responsible for starting and stopping processes that run on a machine according to the deployment information.
  - The agent monitors the machine. That information is then visible through the TIBCO Administrator GUI.
- Supplies the runtime environment, that is, all shared libraries including third-party libraries required by the adapter.

TIBCO Domain Utility

The TRA contains the TIBCO Domain Utility, which is used to manage the components available on a TIBCO administration domain. The utility allows you to:

- Add or remove a machine to a TIBCO administration domain.
- Add or remove the TIBCO Enterprise Message Service server plug-in to a TIBCO administration domain.
- Change TIBCO Rendezvous parameters. This is an advanced option performed only by users familiar with TIBCO Rendezvous. If you want to perform this task, you must perform it on each machine in the TIBCO administration domain, then restart the TIBCO Administration Server.
- Change TIBCO administration domain credentials. This is an advanced option. You must perform it on the machine that hosts the TIBCO Administration Server.
- Remove a secondary TIBCO Administration Server.
- Enable TIBCO administration domain and security management on a machine that hosts TIBCO Administrator.
- Migrate previous TIBCO Administrator installations.

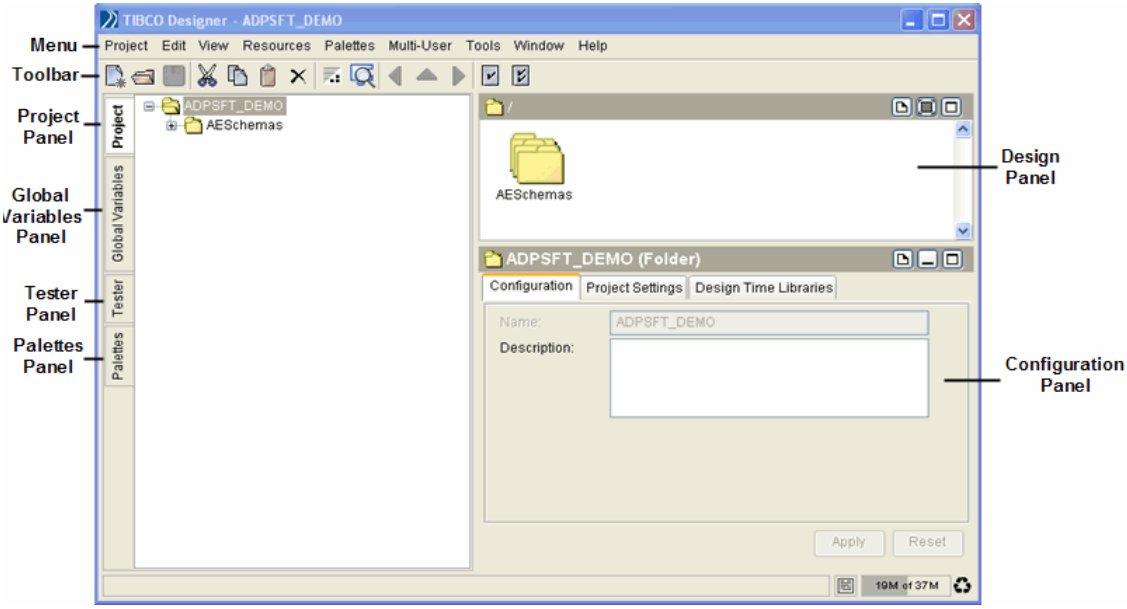
TIBCO Designer

TIBCO Designer provides the design-time environment for configuring a standalone adapter project. Using Designer, you create a project, add adapter services to it with a simple drag-and-drop interface, and specify the configuration information for each adapter service.

Before using TIBCO Designer, ensure that you have read the *TIBCO Designer documentation*. The documentation can be accessed via the TIBCO Designer **Help > Designer Help** from the menu bar. Figure 4 shows the TIBCO Designer interface.

The standalone adapter adds a palette to the TIBCO Designer environment which provides the adapter specific resources. After you install an adapter, a TIBCO Designer palette for the adapter is placed into the appropriate folder. This palette becomes available the next time you start TIBCO Designer.

Figure 4 TIBCO Designer Main Window



TIBCO Administrator

TIBCO Administrator provides user, resource, and application management modules for adapters.

User Management This module allows you to set permissions for adapter users. You can define authentication, users and groups, and assign access control lists to users. This includes security for server-based projects at design-time and for deployed applications at runtime.

Resource Management This module allows you to monitor machines and running applications in a TIBCO administration domain. Alerts can be created, for example, to notify an administrator if the number of processes or disk usage exceed a certain level.

Application Management This module allows you to upload EAR files, and create, configure, and deploy adapters. This module is also used to start and stop adapters.

Load Balancing An adapter can be served by a primary and secondary TIBCO Administration Server. The primary server allows read and write operations, while the secondary server supports read operations. Load balancing is implemented through the use of the TIBCO Rendezvous distributed queue protocol (RVDQ) and therefore is not available for HTTP.

To get the load balancing benefit with HTTP, you must either use an IP redirector or explicitly point to a backup server. Refer to the IP Redirector or HTTP Server documentation for instructions on how to do this.

Failure Recovery You can use a load-balanced TIBCO Administration Server for failure recovery. In a completely trusted environment, you can also use a database back-end for your server and use checkpoints in the database for failure recovery.

TIBCO Administration Domain

A TIBCO administration domain is installed only if you have installed the User Management module.

A *TIBCO administration domain* is a collection of users, machines, and components that an administration server manages. There is only one Administration Server for each administration domain. Components within an administration domain can communicate with systems outside of the domain, but the domain is the administrative boundary of your enterprise integration project.

Each TIBCO administration domain contains one or more machines. Each machine can belong to only one TIBCO administration domain.

By default, all machines within an administration domain are expected to be in the same subnet. You can set up your system to use TIBCO Rendezvous `rvrd` and then use the components across subnets. See the *TIBCO Administrator Server Configuration Guide* for details.

TIBCO Administration Server

The TIBCO Administration Server provides a central storage and distribution point for configuration data and schema data needed by an adapter. The server is included in both Administrator editions.

Each administration domain has one and only one TIBCO Administration Server. The *TIBCO Administration Server* is the machine process that handles the stored projects and requests to manage the TIBCO administration domain.

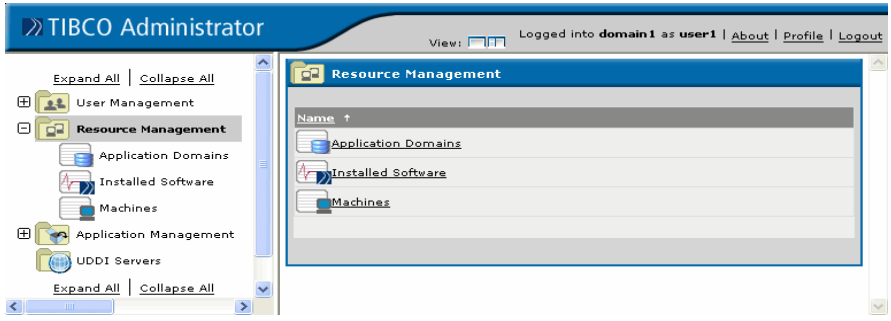
The TIBCO Administrator Server contains its own web server (Apache Tomcat) that can be accessed through the TIBCO Administrator GUI for configuration and monitoring information.

The TIBCO Administration Server supports centralized authentication and authorization. Using the TIBCO Administrator GUI, users with full administrative privileges can define who has access to projects that are managed by the repository server.

TIBCO Administrator GUI

You can access the TIBCO Administration Server using the web-based TIBCO Administrator GUI. The GUI allows you to create user profiles and assign access to projects managed by the Administration Server. You can invoke the GUI from any machine in a TIBCO administration domain.

Figure 5 TIBCO Administrator GUI



TIBCO ActiveMatrix BusinessWorks

TIBCO ActiveMatrix BusinessWorks is a scalable, extensible, and easy to use integration platform that allows you to develop integration projects. TIBCO ActiveMatrix BusinessWorks includes a graphical user interface for defining business processes and an engine that executes the process.

In TIBCO ActiveMatrix BusinessWorks, adapter services publish or subscribe to business data in a decoupled yet reliable manner. The business process receives data from, and routes data to, an adapter service.

TIBCO ActiveMatrix BusinessWorks provides the following activities for use with adapters:

- Publish to Adapter—Publishes data from the process to an adapter, which subscribes to data coming from the process and passes the data to the target application.
- Adapter Subscriber—Subscribes to incoming data published by the adapter.
- Invoke an Adapter Request-Response Service—Communicates (as a client) with an adapter Request-Response Service.
- Adapter Request-Response Server—Starts a process based on the receipt of a request from an adapter.
- Respond to Adapter Request—Sends a response to an adapter for a previously received request.
- Wait for Adapter Message—Waits for the receipt of a message from the Publication Service of the specified adapter.
- Wait for Adapter Request—Waits for the receipt of a request from a Request-Response Invocation Service.

See the TIBCO ActiveMatrix BusinessWorks documentation for more information.

TIBCO Hawk

TIBCO Hawk monitors and manages distributed applications and systems throughout the enterprise. System administrators can monitor application parameters, behavior, and loading activities for all nodes in a local or wide-area network and take action when pre-defined conditions occur. In many cases, runtime failures or slowdowns can be repaired automatically within seconds of their discovery, reducing unscheduled outages, and slowdowns of critical business systems.

Features

TIBCO Hawk features include:

- Extensive monitoring capabilities at the operating system and application levels including process data, disk, and CPU utilization, network statistics, log, and system files.
- Built-in routines within other TIBCO ActiveEnterprise components allow for proactive management. Problems can be found and fixed before failure occurs.
- Hawk Application Management Interface (AMI) routines can be embedded within custom adapters, allowing active management of those adapters by the Hawk Microagents.
- Distributed Microagents support autonomous network behavior so local management and problem resolution can continue during an outage.
- Fault tolerance is achieved through the independent operation of Hawk agents, which continue to perform local tasks even in the event of network failure.

Components

TIBCO Hawk consists of several components: a console display, a central repository for storage of configuration objects, agents, and Microagents whose monitoring duties are defined by the rule bases.

- Agents monitor local conditions and take action or publish alert information that appears in the TIBCO Hawk display.
- Microagents act as an interface to the managed objects and are invoked through their supported methods.

Microagents

TIBCO Adapter SDK provides two Microagents for all adapters. The associated sets of methods can then be used to monitor adapters.

- `COM.TIBCO.ADAPTER`. This Microagent allows you to perform queries on all running adapters, regardless of their classes or applications.

  If you choose this item on the TIBCO Hawk display and invoke one of the associated AMI methods, the corresponding method is invoked in all running adapters.

  Generally, `COM.TIBCO.ADAPTER:n` and `COM.TIBCO.ADAPTER.xyz:n` behave the same except if you invoke a method on `COM.TIBCO.ADAPTER:n` through network query in TIBCO Hawk display, `COM.TIBCO.ADAPTER:n` causes it to perform queries on all running adapters.

- `COM.TIBCO.ADAPTER.`*xyz* (where *xyz* is the name of a class of adapters and could stands for the source or target software package with which the adapter is interfacing). This Microagent allows performing queries on one class of adapter.

  If you choose this item on the TIBCO Hawk display and invoke one of the associated AMI methods, the corresponding method is invoked only in instances of the *xyz* adapter.

Each adapter has its class Microagent to retrieve and set its specific runtime parameters. See *TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne Configuration and Deployment* for more information.

You also can develop your own custom Microagent using the TIBCO Hawk Integration Classes provided by TIBCO Adapter SDK. See *TIBCO Adapter SDK Programmer's Guide* for more information.

Chapter 3 TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne

This chapter explains the features and services of TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne.

Topics

Overview of TIBCO ActiveMatrix Adapter for JDE

TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne allows one-way (publish or subscribe) or two-way (request-response) message exchange between TIBCO ActiveEnterprise applications and the JD Edwards EnterpriseOne enterprise resource planning (ERP) system.

The adapter interoperates with the JD Edwards EnterpriseOne system by making business function calls, extracting data using query lists, and publishing real-time events.

The adapter is compliant with the TIBCO ActiveEnterprise environment, and can be used in conjunction with adapters and products that are compliant with this environment, such as TIBCO BusinessWorks.

The adapter has two main components: design time and run time.

- The design-time component consists of the adapter palette. The adapter palette is an adapter-specific GUI that seamlessly integrates with TIBCO Designer. It allows you to configure the adapter, and will store the configuration in persistent storage.
- The runtime component uses this configuration to publish real-time events using the Publication service, post data to the TIBCO ActiveEnterprise applications using the Subscription service, or allow exchange of data between the TIBCO ActiveEnterprise applications and the JD Edwards EnterpriseOne system using the Request-Response Server service.

Adapter Features

This section summarizes the features of the adapter. These features make use of the interoperability that characterizes the JD Edwards EnterpriseOne system:

An Easy-to-Use GUI

The adapter provides its own design-time component, namely the adapter palette, which seamlessly integrates with TIBCO Designer. This easy-to-use interface allows you to quickly configure adapter-specific features. You can use it to enter, delete, and modify configuration information.

Flexible Adapter Services

A single adapter instance can contain any number of publication, subscription and request-response server services.

Support for Querying Master Data

The JD Edwards EnterpriseOne system stores all enterprise data within the JD Edwards EnterpriseOne System in database tables. The Request-Response Service of the adapter queries these application tables using the XML Query List function. If the query returns a very large data set, the adapter can also distribute the data across multiple invocations. Streaming the data across multiple invocations prevents the client programs from being overwhelmed by huge amounts of data in a single invocation.

Support for Mapping and Translation

The adapter performs mapping of functions and translation of data formats between external applications and the corresponding JD Edwards EnterpriseOne Business Functions. You can use TIBCO ActiveMatrix BusinessWorks with the adapter when data mapping is required.

Robust and Reliable Data Transmission

The following features of the adapter make it a robust integration solution and provide reliable transmission of data through the network:

- **Connection Pool** The adapter has an extensive connection pooling mechanism, which provides an efficient way to manage connections and share them across different service requests. With connection pooling support, the adapter can allocate, re-cycle, reuse and release connections to JD Edwards EnterpriseOne.

Connection pooling provides superior performance and reduces the number of idle connections.

If connections get invalidated or lost due to network outages or non-availability of the JD Edwards EnterpriseOne system, the connection pooling mechanism automatically recreates new connections when the outage is over or when the system is available again. If a connection has been idle for a long time, the connection pooling mechanism closes it. The next time that access to that connection is required, the connection is automatically re-created.

- **Multi-threading** The adapter contains a simple and efficient multi-threading model which allows it to receive and post events and execute Business Functions concurrently. This multi-threading ability increases the performance of the adapter.

  You can configure the adapter to run in single threaded or multi-threaded mode. The number of threads can be increased or decreased for concurrent and simultaneous processing of both outbound and inbound messages. Inbound and Outbound dispatchers determine the number of threads the adapter will have at runtime. The number of Inbound dispatchers equals the total number of outbound or publication services configured in the adapter instance. Each outbound message or real-time event is processed in parallel in a separate thread.

- **Authentication** The adapter provides a simple authentication system of username and password verification for logging into the JD Edwards EnterpriseOne system. Authentication is configured at design-time when setting connection parameters.

Support for Internationalization

The adapter provides internationalization support based on the Unicode support provided by the TIBCO Adapter SDK. Support is provided for string data only. Schema, metadata, error messages, and all other non-string information do not support extended character sets.

Support for Dual Message Transports

The adapter supports the following message transports:

- `TIBCO Rendezvous Transport`— This transport uses subject-based addressing to provide support for both multicast or broadcast and point-to-point communications. You can configure the delivery modes of the messages and the wire format used when you configure the adapter service.
- `JMS Transport`—TIBCO Enterprise for JMS must be installed to use the JMS transport. The JMS administration interfaces allow you to create and manage

administered objects such as Connection Factories, Topics, and Queues. JMS clients can retrieve references to these objects by using Java Naming and Directory Interface (JNDI). Creating static administered objects allows clients to use these objects without having to implement the object within the client. When a JMS client starts, it performs a JNDI lookup for the connection factories that it needs. For details on JNDI, see the *TIBCO Enterprise for JMS User's Guide*. You can configure the connection factory type and the delivery mode using TIBCO Designer, when you configure the adapter service.

Quality of Service Support

The adapter supports the following levels of service for TIBCO Rendezvous wire formats:

- Reliable message delivery, which ensures that each multicast or broadcast message is received as long as the physical network and packet recipients are working, and that the loss of a message is detected.
- Certified message delivery, which guarantees that every certified message reaches its intended recipient and is received in the order sent.
- Distributed Queue, explained in Distributed Queue Support on page 21.

The adapter supports the following levels of service for the JMS wire formats:

- Durable
- Non-durable

Distributed Queue Support

A distributed queue is a group of cooperating transport objects, each in a separate process. Each transport object is called a member. To balance the transmission load among servers, the adapter can use distributed queues for *one-of-n* delivery of messages to a group of servers. Each member of a distributed queue must listen for the same subject using the TIBCO Rendezvous Distributed Queue listener objects. Even though many members listen for each inbound message (or task), only one member processes the message. For details on distributed queues, see *TIBCO Rendezvous Concepts*.

In the queue mode within TIBCO Enterprise for JMS, each listener is a single receiver of a point-to-point message. However, the listeners can be configured as a set of receivers, each of which receives a fraction of the messages. For details on TIBCO Enterprise for JMS distributed queues, see the *TIBCO Enterprise for JMS User's Guide*.

Load balancing for the processing of TIBCO Rendezvous or JMS certified messages is supported using distributed queuing. The messages from TIBCO Rendezvous or TIBCO Enterprise for JMS are distributed equally among all instances that belong to the same group. This distributes the data load over several adapter instances. However, the order in which the data is sent to the application server is not guaranteed.

TIBCO Wire Format Support

Two wire formats are supported by the adapter:

- ActiveEnterprise Message wire format is the TIBCO ActiveEnterprise standard wire format. It can include metadata information.
- JMS Message format is a standard message which may have XML content. It follows Sun's Java Messaging Service wire format.

Tracing and Exception Handling Support

The adapter allows external applications to take advantage of the tracing and exception handling provided through ActiveEnterprise.

Adapter Services

The JDE adapter provides the following services:

- Publication Service
- Subscription Service
- Request-Response Service

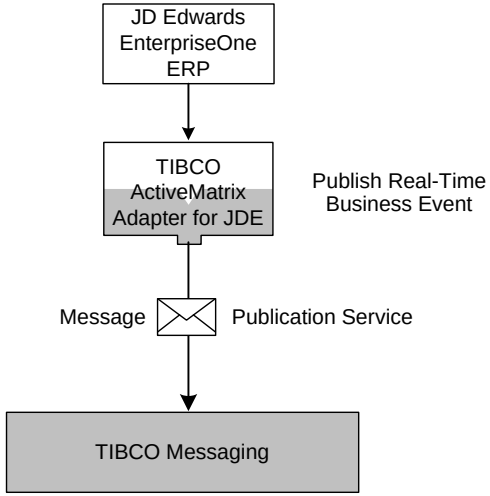
Publication Service

The adapter uses JD Edwards EnterpriseOne real-time event notification to publish data generated by JD Edwards EnterpriseOne applications to TIBCO ActiveEnterprise applications.

If the JD Edwards EnterpriseOne system has been configured so that an event is generated whenever data changes inside the system, the adapter can publish this event to TIBCO ActiveMatrix applications. A typical example is the publication of a sales order event whenever a new sales order is created.

In this publication service, the real-time event (an invoice prepared by the point-of-sale clerk, for example) originates in the JD Edwards EnterpriseOne system and is published by the adapter to other applications, such as "the BackOffice" Accounting System.

Figure 6 Typical Publication Service Flow



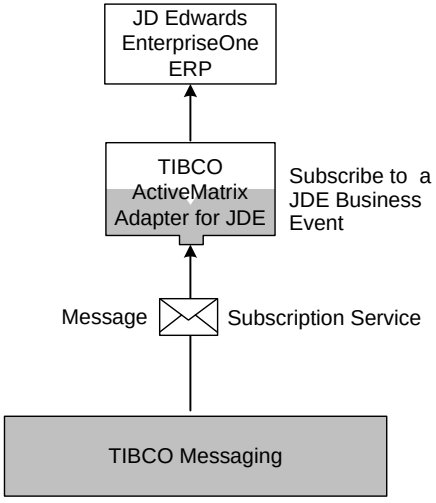
Subscription Service

In the subscription service scenario, the subscription service of the adapter gets the message from the TIBCO environment and sends it to the JD Edwards EnterpriseOne application. A subscription service does not return a response.

The adapter unpacks events from the message and performs a Business Function invocation on the JD Edwards EnterpriseOne enterprise system. The adapter can invoke one Business Function, or it can invoke multiple Business Functions that are chained together in a group.

For example, `CalculateQtyOnHand` is a single Business Function. However, a sales order can contain multiple Business Functions. For example, the task `CreateSalesOrder` requires multiple Business Functions `AddHeader`, `AddLine` (multiple times), and `EndDoc` to be executed in order. `CreateSalesOrder` is an example of a Business Function Group.

Figure 7 Typical Subscription Service Flow



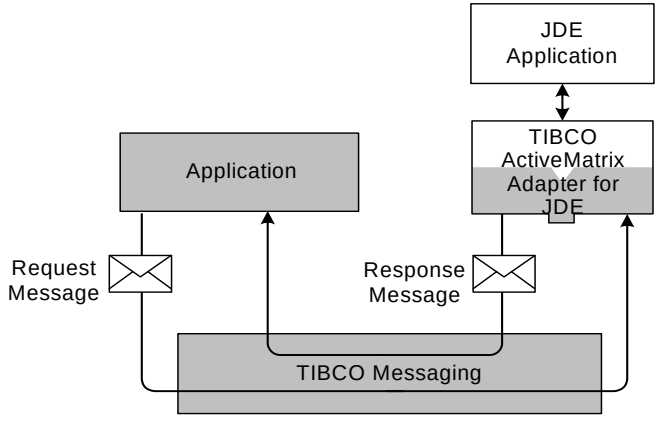
Request-Response Service

The adapter, acting as a Request-Response server, enables and allows the TIBCO ActiveEnterprise clients (for example, TIBCO ActiveMatrix BusinessWorks) to invoke Business Functions or Business Function Groups on the adapter. It also allows the users to perform efficient table queries with the XML QueryList operation. The adapter executes these operations in the JD Edwards EnterpriseOne system and returns the result of the invocations.

For extracting data from JD Edwards EnterpriseOne database tables, views, or table conversions, the adapter has a Request-Response Query service. The database table or view information is required when a one time data load is required in an external application. For example, if an application is implemented in the enterprise after the JD Edwards EnterpriseOne ERP system was implemented, using request response, the customer master information can be extracted from JD Edwards EnterpriseOne and imported into the new system. Similarly, item catalog information may be loaded from the JD Edwards EnterpriseOne system to the CRM system.

Using the Request-Response Service, the adapter gets a request from the TIBCO environment. It unpacks the request and converts it into a format understandable by the JD Edwards EnterpriseOne system. The adapter then uses this data to make a Business Function or XML query list invocation in the JD Edwards EnterpriseOne system. The data returned by the resulting invocation is packed into a reply and sent back to the calling application. If any errors are encountered during invocation they are sent back in the error field of the reply.

Figure 8 Typical Request Response Service Flow



Chapter 4 TIBCO ActiveMatrix Adapter and JD Edwards EnterpriseOne Integration

This chapter introduces the interaction between the adapter and JDE and the business scenario for use of TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne.

Topics

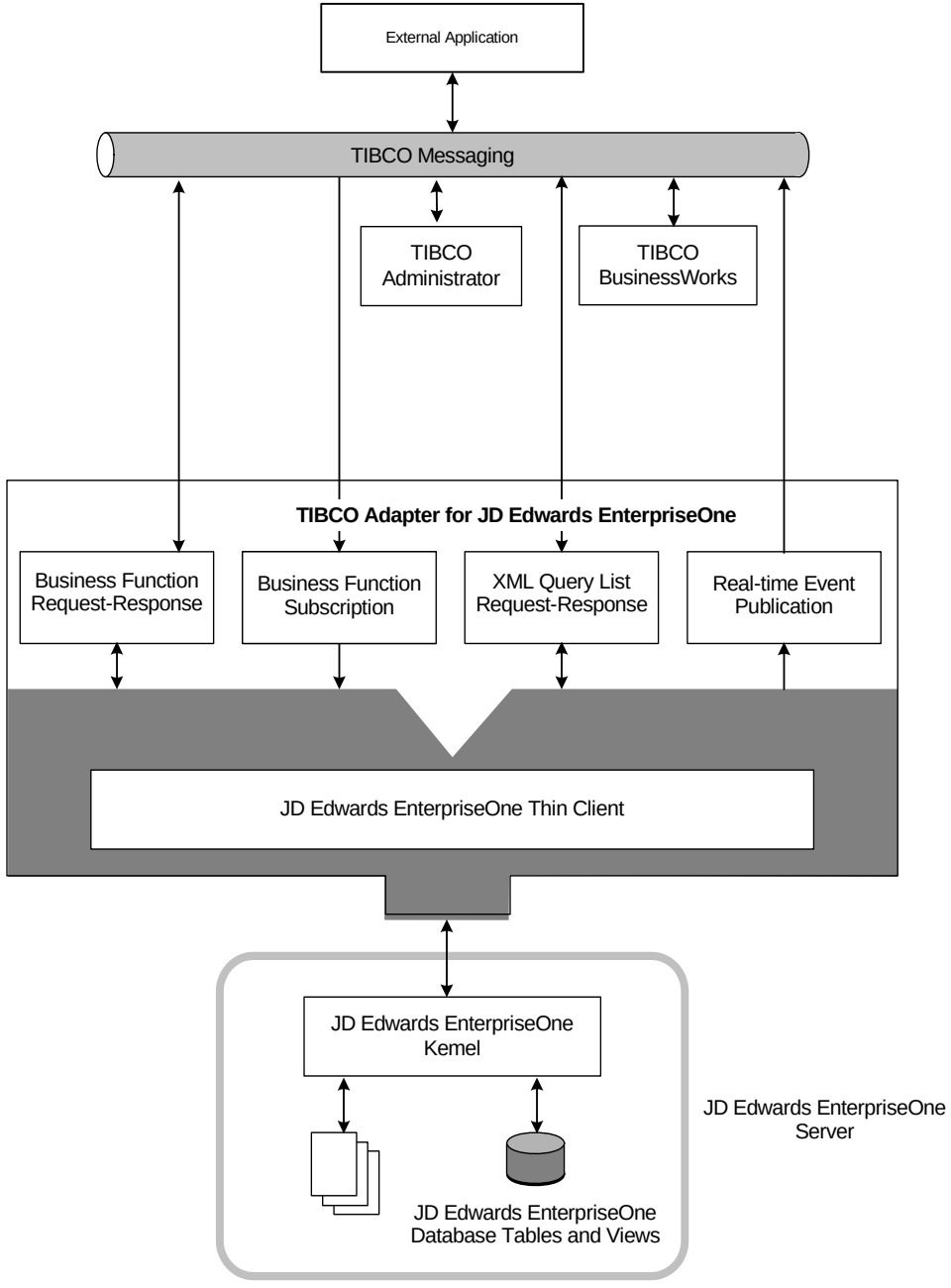
Adapter and JD Edwards EnterpriseOne Interaction

This section provides a brief overview of the way that the TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne interacts with the JD Edwards EnterpriseOne system and with TIBCO ActiveEnterprise applications such as TIBCO ActiveMatrix BusinessWorks.

Logical Architecture

A typical logical architecture that integrates the adapter is shown in Figure 9. The adapter can run on the machine that hosts the JD Edwards EnterpriseOne server, or on a different machine. The adapter requires the JD Edwards EnterpriseOne thin client libraries on the machine on which the adapter is running.

Figure 9 Logical Architecture



Typical Workflow

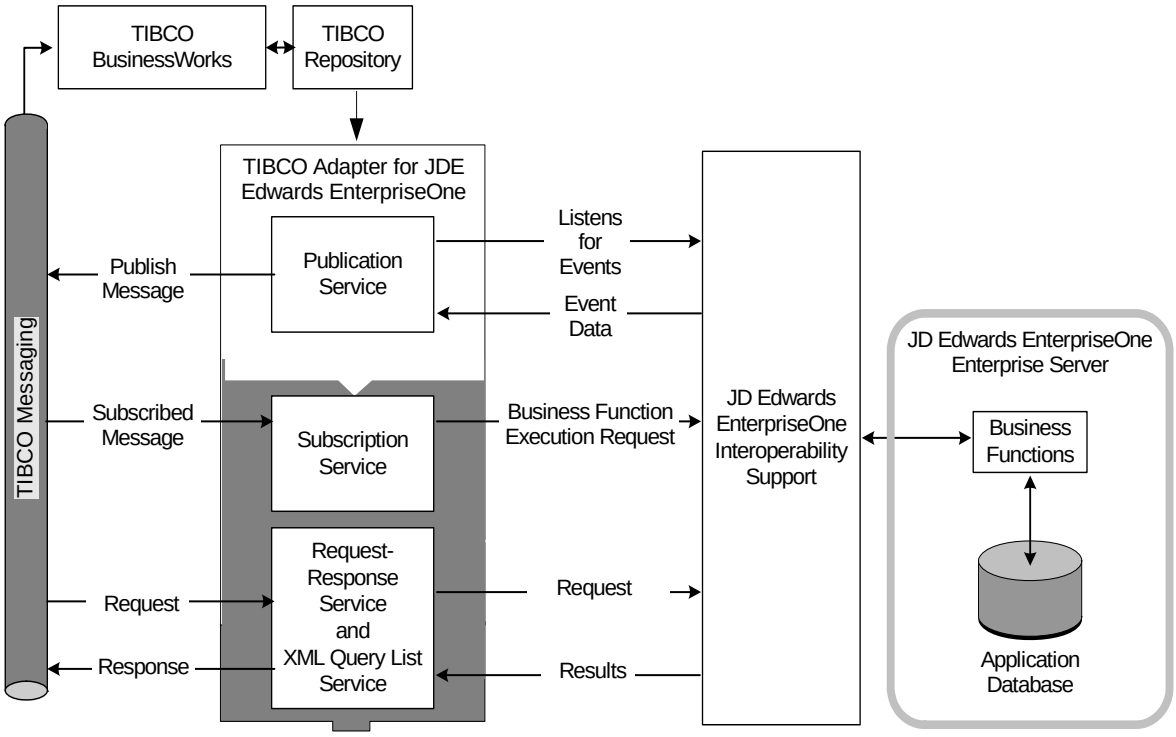
Figure 10 shows the adapter integrated with JD Edwards EnterpriseOne and illustrates the typical flow of a business function execution, and the adapter services related to the business function execution.

TIBCO Administrator manages the repositories that contain configuration data and schema used by the adapter services. TIBCO ActiveMatrix BusinessWorks is used to transform data.

As shown in Figure 10, four adapter services are available: Publication Service, Subscription Service, and two types of Request-Response Service.

- Using the Publication Service, the adapter listens for real-time events on JD Edwards EnterpriseOne server and publishes them using a TIBCO messaging transport.
- Using the Subscription Service, the adapter listens for messages sent from the ActiveEnterprise environment and posts them to the JD Edwards EnterpriseOne server.
- The adapter uses the Request-Response Service to pass requests from an application in the ActiveEnterprise environment to the JD Edwards EnterpriseOne server, then returns the responses to the application.
- The XML Query List Service is also a Request-Response Service that returns Application Table data from the JD Edwards EnterpriseOne server. This service can use JD Edwards EnterpriseOne Tables, Views or Table Conversions to return the data.

Figure 10 Typical Workflow



Use of the TIBCO Adapter for JD Edwards EnterpriseOne

Consider a scenario where a customer buys a few items from a Store. The Store manages all its enterprise functions using the JD Edwards EnterpriseOne system. The Point of Sale clerk creates an invoice; the customer pays by a store credit card. The delivery of items will be made to the customer in few weeks. The customer must also receive a detailed confirmation of his order and a statement of balance or amount he owes to the store.

The scenario above can be accomplished with the TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne and an integration platform such as TIBCO ActiveMatrix BusinessWorks.

The system integrator carries out the following tasks:

- Creates a real-time outbound event for the invoice document in the JD Edwards EnterpriseOne system.
- Creates a new adapter configuration using the TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne palette in the TIBCO Designer, and configures a new publication service and chooses the real-time event he just created as the publication data event.

Whenever a new invoice document is created or an existing one modified, the JD Edwards EnterpriseOne system automatically sends the modified or the new document to the adapter in the form of a real-time event. The adapter will publish this document for consumption by other TIBCO applications.

The system integrator uses TIBCO ActiveMatrix BusinessWorks to map the invoices (published as real-time events) to an accounting system. In the accounting system, information from the invoice is used to update the customer's account and eventually prepare a monthly statement.

The information about the line items or items the customer ordered and information pertaining to the ship to address of the customer is published to the shipping department, which is responsible for packaging and delivering the order to the customer. All the above transactions happen in real-time, immediately after the Customer places an order at the Point of Sale counter.

Glossary

A

ActiveEnterprise

One of the three product groupings within TIBCO Software Inc, focused on enterprise application integration. The other two are ActivePortal™ and ActiveExchange™ (see www.tibco.com for details).

adapter tester

The adapter tester is a tool that can be used for testing adapters.

AESchemas folder

When you configure a standard adapter, TIBCO Designer creates schema resources and places them in the appropriate location in the `AESchemas` folder. A number of schema resources are included in the `AESchemas` folder by default. These resources represent schema that are recognized by the TIBCO Adapter SDK. Each schema file contains a collection of classes, scalars, associations, unions, and sequences.

B

Business Functions

Business Functions are executable functions that can be used in any application of the JD Edwards EnterpriseOne system. Business Functions allow you to reuse the same code in several applications. A typical Business Function might be a line item or an address in a purchase order.

Business Function Groups

A Business Function group encapsulates the logic for the creation of one Business entity in the JD Edwards EnterpriseOne system. The Business Functions contained in the Business Function Group or their invocation order could be different in each deployment. Business Function Groups are groups of Business Functions that are executed together. The group can be set up within the TIBCO ActiveMatrix Adapter for JD Edwards EnterpriseOne. A typical Business Function group might be a complete purchase order, including header information, customer information and line items.

E

Enterprise Archive Resource (EAR) file

An EAR file contains information about the adapter instances and TIBCO ActiveMatrix BusinessWorks p processes you want to deploy. The format is used by TIBCO Administrator. The EAR file is imported into TIBCO Administrator where you can deploy, start, and manage the adapter instance on the machines of your choice.

G

global variable

Global variables provide an easy way to set defaults for use throughout your project. You can define a global variable either at design time in TIBCO Designer or in the adapter properties file or at runtime in TIBCO Administrator.

L

local repository

A project exported to a local repository is saved in `.dat` format. Projects saved in `.dat` format should only be used for development and testing. The format can be used where data is not shared by more than one adapter. It is possible to have a few local adapters accessing a local project in read-only mode. It is, however, not possible to have more than one local adapter accessing a local project in read and write mode.

Data are loaded at startup for local projects, so a local project has higher memory requirements.

P

palette

Palettes organize resources and allow you to add them into your project. Palettes are available from the palette panel in TIBCO Designer. Resources are visible components in a palette. You select resources in the palette panel and drag-and-drop them into the design panel to add them to your project.
Each adapter you install adds one or more palettes during installation. Which palette is displayed depends on the resource selected in the project tree and on your preferences. In the default view, the current selection in the project tree determines which palettes are displayed in the palette panel.

project

A collection of configured adapter resources and it contains configuration information for one or more adapter instances.

A local project is typically used at designtime for testing. For production, a project is typically managed by an administration server provided by the TIBCO Administrator for the standalone adapter and TIBCO ActiveMatrix Administrator for the adapter service engine.

publication service

An adapter publication service recognizes when business events happen in a vendor application, and asynchronously sends out the event data in real-time to interested systems in the TIBCO environment

R

request-response service

In addition to asynchronously publishing and subscribing to events, an adapter can be used for synchronously retrieving data from or executing transactions within a vendor application. After the action is performed in the vendor application, the adapter service sends a response back to the requester with either the results of the action or a confirmation that the action occurred. This entire process is called request-response, and it is useful for actions such as adding or deleting business objects.

request-response invocation service

An adapter request-response invocation service is similar to the request-response service, except that the roles are reversed. The vendor application is now the requester or initiator of the service, instead of the provider of the service. The adapter service acts as a proxy, giving the vendor application the ability to invoke synchronously functionality on an external system.

Real-time Events

These are events that happen in the JD Edwards EnterpriseOne system or another application, and are communicated between systems by the adapter, which listens for and publishes events. A real-time event might be the issuing of a certain purchase order.

S

schema

Schema defines the date used by an adapter. Adapters use schema to describe data received from or sent to the TIBCO environment. For example, when you define a service such as a publication service, you need to define the schema that describes the data that will be published. When you save the project, both the adapter configuration and the corresponding schema are saved.

Server Repository

A project exported to a server repository is managed by a TIBCO Administration Server running in a separate process, typically elsewhere on the network. One or more adapters can communicate with a project managed by an Administration Server. Each can support multiple projects.

standalone adapter

Standalone adapter projects are created and configured by using TIBCO Designer and deployed by using TIBCO Administrator.

You can create adapter projects which run as a standalone process.

Subscription Service

An adapter subscription service asynchronously performs an action, such as updating business objects or invoking native APIs, on a vendor application. The adapter service listens to external business events, which trigger the appropriate action.

T

Table Conversion

Table conversion is a special form of Universal Batch Engine (UBE) that enables you to do a high-speed manipulation of data in tables. You can use table conversions to gather, format, import, and export data. The adapter can run and query the table conversion output using XML Query List.

Z

ZIP Archive

A project exported to a ZIP archive is written to the location you specify as a read-only ZIP file. A project exported as a ZIP archive can be imported into TIBCO Designer.

Index

U