



TIBCO ActiveMatrix BusinessWorks™

Concepts

Version 5.16.1 | February 2025

Contents

Contents	2
Business Integration	6
The Challenge of Application Integration	6
Integration Benefits	6
Integration Platform Requirements	7
Business Integration Scenario	8
Problem Definition	8
Addressing the Challenge	9
Example Scenario Run-Time Implementation	10
The ActiveMatrix BusinessWorks Integration Platform	12
ActiveMatrix BusinessWorks Prerequisites	12
TIBCO Administration Domain	13
Architecture	15
Fundamentals	15
Design-Time Architecture	17
Run-Time Architecture	20
ActiveMatrix BusinessWorks Features	21
Messaging	22
Adapters	22
Business Process Modelling	24
Schemas and Data Mapping	25
Manual Activities	27
ActiveMatrix BusinessWorks Methodology	28
Introduction	28
Phase 1: Analysis	29
Phase 2: Domain Setup and Installation	29

Planning the Domain	30
Installing Components	31
Phase 3: Services Configuration	32
Adapter Configuration Overview	32
Web Services Overview	33
Services Used by the Example	33
Phase 4: Process Design	33
Overview	34
Activities Used by the Example	34
Phase 5: Deployment Configuration and Deployment	35
Phase 6: Production	35
Phase 1: Analysis	37
Step 1: Define and Delimit the Problem	37
Step 2: Identify Processes	38
Step 3: Identify Components	38
Shared Resources	39
Services and Corresponding Activities	39
ManualWork Activities	40
Transitions and Conditions	40
Mapping	40
Exceptions	41
Step 4: Describe Business Events and Objects	41
Step 5: Design Business Processes	41
Step 6: Consider Domain Setup	42
Phase 2: Domain Setup	43
TIBCO Administration Domain	43
TIBCO Administration Server	44
TRA	44
Phase 3: Services Configuration	45

Introduction: ActiveMatrix BusinessWorks Services	45
Service Characteristics	46
Invocation Modes	46
Service Types	46
Adapter Service Configuration Steps	47
Step 1: Installing the Adapter	47
Step 2: Setting up the Design-Time Adapter	48
Step 3: Configuring the Run-Time Adapter	50
Step 4: Accessing the Adapter Service From the Process	50
Phase 4: Business Process Design	53
Introduction	53
Business Processes	53
Process Design Steps	54
Step 1: Define Shared Resources	54
Step 2: Create Process Definitions	55
Step 3: Add a Process Starter	57
Step 4: Add Activities	58
Step 5: Optionally, Add Manual Work Activities	59
Step 6: Create Transitions Between Activities	60
Step 7: Perform Mapping and Transformation for Each Activity	61
Step 8: Optionally, Group Activities As Needed	61
Step 9: Test the Process	62
Phase 5: Deployment	64
Introduction	64
ActiveMatrix BusinessWorks Project Phases	65
OpenTelemetry	66
OpenTelemetry Tracing	66
OpenTelemetry Metrics	68
Phase 6: Production	72

Introduction	72
Architecture	72
Monitoring and Management Options	74
Glossary	75
TIBCO Product Documentation and Support Services	81
Legal and Third-Party Notices	84

Business Integration

TIBCO ActiveMatrix BusinessWorks™ is a scalable, extensible, and easy to use integration platform that allows you to develop integration projects. ActiveMatrix BusinessWorks™ uses the TIBCO Designer™ graphical user interface (GUI) for defining business processes and the ActiveMatrix BusinessWorks process engine executes the business process.

ActiveMatrix BusinessWorks also works with TIBCO Administrator™, a web-based GUI for monitoring and managing run-time components.

This section discusses what is required of an integration platform, and how ActiveMatrix BusinessWorks meets these requirements.

The Challenge of Application Integration

Application integration is one of the most pressing challenges of today's enterprise. An enterprise may need to integrate back-office systems with the Internet, a purchase order management system with the customer service management system, or any of the above with legacy or ERP (Enterprise Resource Planning) systems.

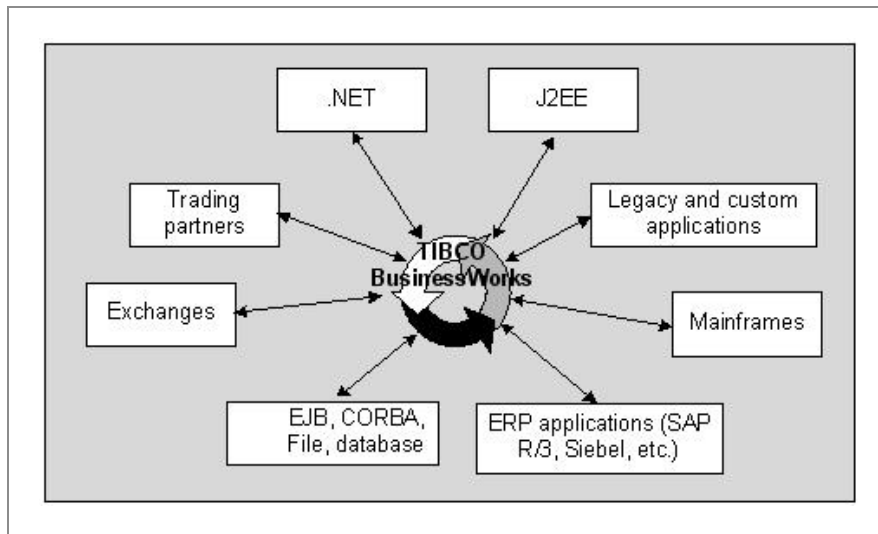
An integration platform should allow you to design the business process, that is, the flow of data. The business process should transparently receive and send data throughout the enterprise and beyond.

Integration Benefits

The integrated enterprise works seamlessly. Different departments and groups share communications and can together respond quickly to customer needs. With less time spent on administrative and manual tasks, employees become more productive and the integrated system yields a significant return on investment. This return increases as the company grows.

The illustration below shows how ActiveMatrix BusinessWorks connects to applications of different types, databases, trading partners and exchanges, and so on. This capability allows you to use ActiveMatrix BusinessWorks to integrate all aspects of your enterprise.

Figure 1: ActiveMatrix BusinessWorks communication throughout the enterprise



Integration Platform Requirements

To be successful, your integration platform must meet the following requirements.

- **Short deployment cycle**—The integration project must be ready to go to production within a realistic timeframe and deploying from development to a running project must go smoothly.
- **Scalability and extensibility**—The project must be scalable (respond to increasing demand) and extensible (allow integration of new applications or addition of new business processes). Extensibility also means that the project must be flexible and adaptable so you can potentially adapt it to multiple departments in the same company.
- **Ease of use**—Integration projects are often developed by outside companies or consultants. When the project is complete, the company itself becomes responsible for maintenance and updates, and employees usually face a steep learning curve. If the integration platform is easy to use, the project can be developed in house. Cost of ownership is greatly reduced because the expertise is already there.

Business Integration Scenario

To illustrate some of the functionality available as part of ActiveMatrix BusinessWorks, the rest of this manual uses a simple example scenario presented in this section. The section starts with a problem definition, then discusses the run-time implementation. Individual aspects of the integration project are discussed in later sections.

Problem Definition

EasyWare Incorporated is a manufacturer of computer hardware. So far, the department responsible for purchase order management has received orders by telephone and has manually entered them into a PeopleSoft Order Management system. The customer service department uses the information in the PeopleSoft system as well, but finds that not all information they need is available there. An additional concern is that shipping information is not included in the Order Management system. As a result, customers do not receive notification when items are shipped and customer service representatives must access the shipping information in a two-step process: first extracting the Order ID from the PeopleSoft system, then extracting the shipping information from the shipping log using the Order ID.

Because a high volume of sales cannot be handled efficiently with this setup, and because currently information available to the customer service department is incomplete, customer satisfaction becomes an issue. Management decides to make the following changes:

- Make order entry possible by way of a secure web server outside the firewall which communicates with an application server that supports JMS.
- Add a Siebel customer service system that receives information about each purchase order.
- Before order fulfillment, require approval by a credit check specialist for orders over \$10 000.
- Before the order is entered into the Siebel system, add information about shipping date and time to each item. The information should be retrieved interactively from the shipping company's web site via the Internet.

Customer service representatives can then have easy access to all ordering information.

Addressing the Challenge

The challenge faced by EasyWare is to add the new capabilities to the business process while, at the same time, reducing complexity for the end user. After careful deliberation, the IT department decides to prepare a prototype using ActiveMatrix BusinessWorks.

Design

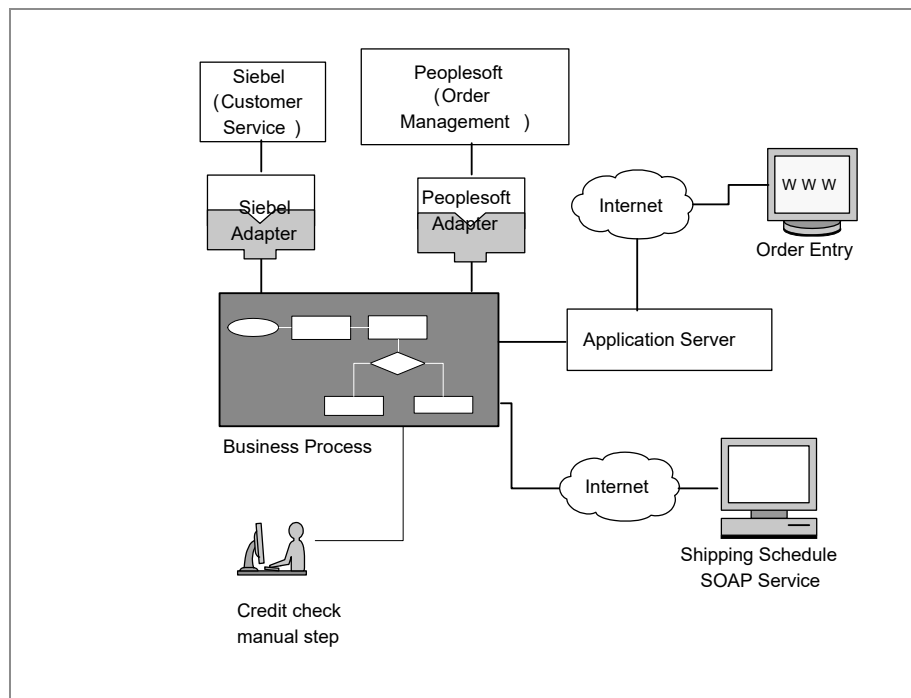
The goal of the integration project is to allow EasyWare to receive orders either by telephone—with direct access to PeopleSoft as before—or from an application server. The PeopleSoft Order Management system continues to be used, but data can be entered manually as before or can arrive from the Internet via the application server. In addition, the system must integrate with the Siebel customer service system.

This results in the following components of the integration project:

- At the center is the business process, which interacts with the different services using the appropriate messaging protocol.
- An order is entered by way of a web application and the order is sent to the PeopleSoft order-entry system through the PeopleSoft adapter. The PeopleSoft order-entry system responds with a new order that includes an order ID. This part of the process is a request-response service.
- A ManualWork activity sends the order to the credit check specialist. The process is on hold until approval has been made.
- The business process accesses the shipping schedule by connecting to the shipper's web site using SOAP.
- A Siebel adapter enters the complete order, including the PeopleSoft Order ID and the shipping information, into the Siebel system.

[Example scenario components](#) shows the components that are needed to execute the integration project at runtime.

Figure 2: Example scenario components



Example Scenario Run-Time Implementation

At runtime, the project would proceed as follows (numbered steps match the numbers in [Example scenario run-time implementation](#)):

Procedure

1. Orders arrive from the distributors on the Internet.
2. The orders are processed by a customized order capture system built on top of an application server.
3. A ActiveMatrix BusinessWorks JMS Queue Receiver activity receives the incoming order document.

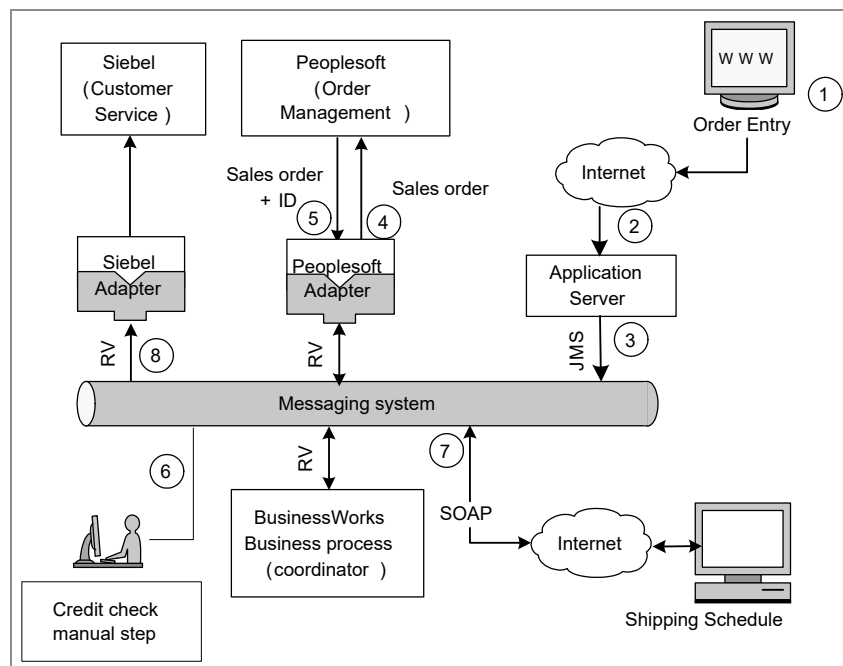
In [Example scenario run-time implementation](#), the box labeled Business process (coordinator) represents the ActiveMatrix BusinessWorks engine executing the business process.

4. ActiveMatrix BusinessWorks transforms the order, which was supplied in XML to a format PeopleSoft expects and the PeopleSoft adapter submits the order to the

PeopleSoft Order Management system.

5. The PeopleSoft system accepts or rejects the order. When it accepts the order, it includes an Order ID.
6. If the order amount is greater than \$10,000, the order is sent out for credit check. When the credit check is not successful, the customer is informed immediately and the order placed on indefinite hold. Otherwise, if approval arrives, the process continues.
7. ActiveMatrix BusinessWorks checks the shipping schedule using a web service activity and adds the ship date to the order.
8. ActiveMatrix BusinessWorks sends the information about the order, including the Order ID and the shipping information, to a Siebel adapter. The Siebel adapter adds a new customer service record to the Siebel system.

Figure 3: Example scenario run-time implementation



The ActiveMatrix BusinessWorks Integration Platform

This section introduces the ActiveMatrix BusinessWorks integration platform by discussing the following topics:

ActiveMatrix BusinessWorks Prerequisites

The following products are prerequisites for ActiveMatrix BusinessWorks and must be installed prior to installing ActiveMatrix BusinessWorks:

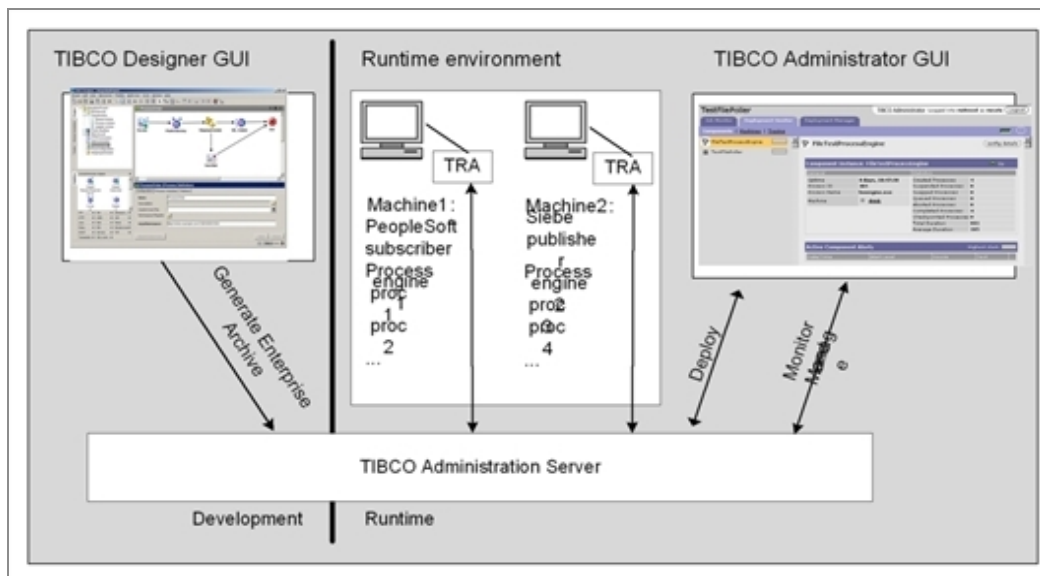
- **TIBCO Runtime Agent™** supplies a number of TIBCO and third-party libraries used by ActiveMatrix BusinessWorks and other TIBCO products both at design-time and runtime. This includes, for example, TIBCO Rendezvous software. You must install TRA™ on each machine that hosts a ActiveMatrix BusinessWorks process engine.
 - **TIBCO Designer** is a graphical user interface (GUI) that is installed as part of TRA. TIBCO Designer provides an easy-to-use design-time environment that supports adapter configuration, process design, and testing of the integration project.
- **TIBCO Administrator** supports deployment, security administration, and monitoring and management of processes and machines. TIBCO Administrator consists of the TIBCO Administration Server and the web browser based TIBCO Administrator GUI.

Once the prerequisite products are installed, you may install the following components of ActiveMatrix BusinessWorks:

- The ActiveMatrix BusinessWorks engine runs the business processes in test mode and at run-time.
- The ActiveMatrix BusinessWorks design-time plug-in provides the palettes and resources to TIBCO Designer for creating business processes. ActiveMatrix BusinessWorks was designed using a plug-in architecture. There are special-use plug-ins that you can procure separately, such as the ActiveMatrix BusinessWorks EJB Plug-in that allows you to work with Enterprise Java Beans (EJBs).
- The ActiveMatrix BusinessWorks engine runs the business processes in test mode and at run-time.

The illustration below shows how the products work together.

Figure 4: ActiveMatrix BusinessWorks development



TIBCO Administration Domain

The TIBCO administration domain combines the benefits of a distributed run-time environment with centralized design time deployment, and run-time monitoring and management. This section discusses the TIBCO administration domain, the associated security implementation, and how you monitor and manage the domain.

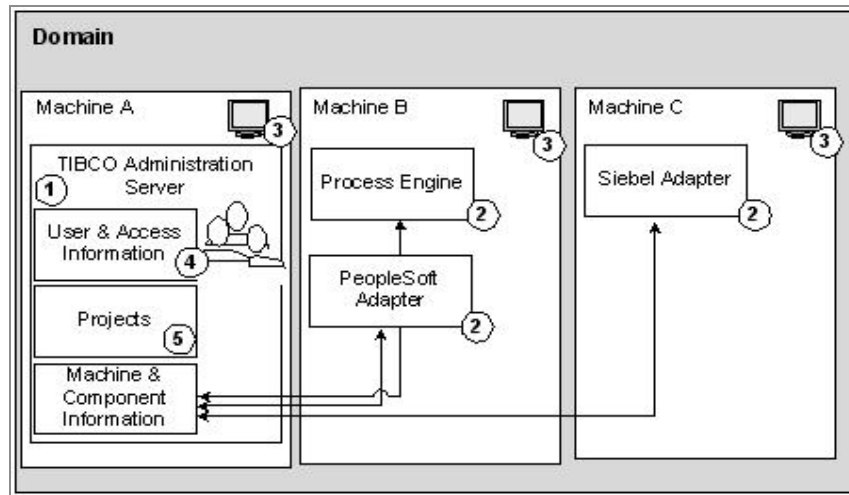
TIBCO Administration Domain Overview

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

Note that when the TIBCO Administration Server goes down, all process engines and adapters continue to run.

[Components of a TIBCO administration domain](#) illustrates the TIBCO administration domain and its contents.

Figure 5: Components of a TIBCO administration domain



A TIBCO administration domain consists of these elements, numbered correspondingly in [Components of a TIBCO administration domain](#):

Procedure

1. **TIBCO Administration Server**—Each TIBCO administration domain has one and only one TIBCO Administration Server.
2. **Components**—Component software includes the ActiveMatrix BusinessWorks engine and adapters.
3. **Machines**—Each TIBCO administration domain contains one or more machines. A machine can be added to an administration domain when a ActiveMatrix BusinessWorks component or adapter is installed. By default, all machines within TIBCO administration domain are expected to be in the same network subnet. You can, however, set up your system to use TIBCO Rendezvous rvr and can then use ActiveMatrix BusinessWorks across subnets.
4. **User and Access Information**—User and authorization information is specified with the TIBCO Administrator GUI and stored in the domain data store. For more information, see [Security](#).
5. **Projects**—A project is created with the TIBCO Designer GUI. You create an Enterprise Archive file (EAR file) for the project with TIBCO Designer, then send that file to the machine where TIBCO administration server resides. You can then deploy the project with TIBCO Administrator. Once deployed, the project becomes visible in the TIBCO Administrator GUI and its components can be started, stopped, and monitored from there. For more information, see [Projects](#).

Security

The TIBCO Administration Server supports centralized authentication and authorization. Using the TIBCO Administrator GUI, users with full administrative privileges can define which users should have access to which part of the system.

- **Authentication**—The verification of the identity of a person or process.
- **Authorization**—Permission to view or execute. TIBCO Administrator gives users access rights to the functionality of the product they need. For example, access rights are given to view or to write to projects at design time from TIBCO Designer or to view or to manage modules from the TIBCO Administrator console. The TIBCO Administration Server controls that access.

ActiveMatrix BusinessWorks supports authentication and authorization for both data stores and components (process engines or adapters) in the TIBCO administration domain. For example, only authorized users may start or stop process engines or adapters. This fine-grained user authorization scheme allows you to customize the system to your company's needs.

Monitoring and Managing the Domain

The TIBCO Administrator GUI allows you to monitor and manage the different elements of the TIBCO administration domain if you are authorized to do so.

View, add, and delete users and assign access privileges to each user. Monitor and manage the machines in the TIBCO administration domain. Monitor deployments. This includes viewing component status and throughput and looking at traces, which can also be exported to a file. Manage deployments. This includes stopping and starting process engines and adapters.

Architecture

This section explains the ActiveMatrix BusinessWorks architecture at design time and runtime. It discusses these topics:

Fundamentals

The ActiveMatrix BusinessWorks architecture is based on the following set of fundamentals:

Support for Standards

Your integration platform must support standards for several reasons. A standards-based integration platform supports you best as you add applications to your enterprise or need to communicate with new business partners. Standards are essential as you are planning for the future of the project because standards facilitate updates. Some of the applications you use may already be using standards, and integration development will be faster and easier. Support for standards also removes dependency on one company's services and makes applications from different companies interact more easily.

ActiveMatrix BusinessWorks supports the most widely used standards for the different aspects of an integration project:

- **J2EE Compliant**—JMS, EJB, JNDI
- **Protocols**—Web services (SOAP, WSDL), HTTP, HTTPS
- **Messaging**—JMS, TIBCO Rendezvous
- **Data Description**—Native support for DTD, XSD, and TIBCO AE Schema
- **Data Representation and Expressions**—Native support for XML, XPath
- **Data Transformation**—XSLT

ActiveMatrix BusinessWorks also supports a plug-in for B2B interactions.

Integrated Development Environment

Your integration project must be supported by an integrated development environment that spans all phases of the project. With ActiveMatrix BusinessWorks, the process design, deployment, and run-time environment are tightly integrated even though the run-time environment supports a distributed architecture.

Procedure

1. At design time, you use TIBCO Designer to configure services, such as adapters, and design your business processes.
2. You can use the TIBCO Designer in test mode to debug the process definitions in your integration project.
3. You can use TIBCO Administrator to deploy processes to process engines and to deploy adapter services to adapters on the individual machines.
4. Optionally, you can configure manual activities using TIBCO Designer. When you

deploy the project, that information is then used by InConcert. The users authorized to handle the manual activities can be specified in TIBCO Administrator, then exported to TIBCO InConcert.

5. You can start the processes using the TIBCO Administrator GUI. You can then monitor and manage all processes using TIBCO Administrator.

Extensibility and Scalability

As your enterprise grows, new applications are added and volume of data increases. Scalability to support higher volume and extensibility to support additional applications or a larger number of process engines or adapter instances become paramount.

ActiveMatrix BusinessWorks has been designed to be extensible and scalable. Working in a distributed fashion, you deploy the different adapter services and business processes on different machines in the TIBCO administration domain. When the business process applications' volume increases, you add machines to the domain. As you acquire new applications for your enterprise, you install the appropriate adapters into the domain. You configure the adapter, modify your process definition, then deploy the adapter service on the machine of your choice.

Design-Time Architecture

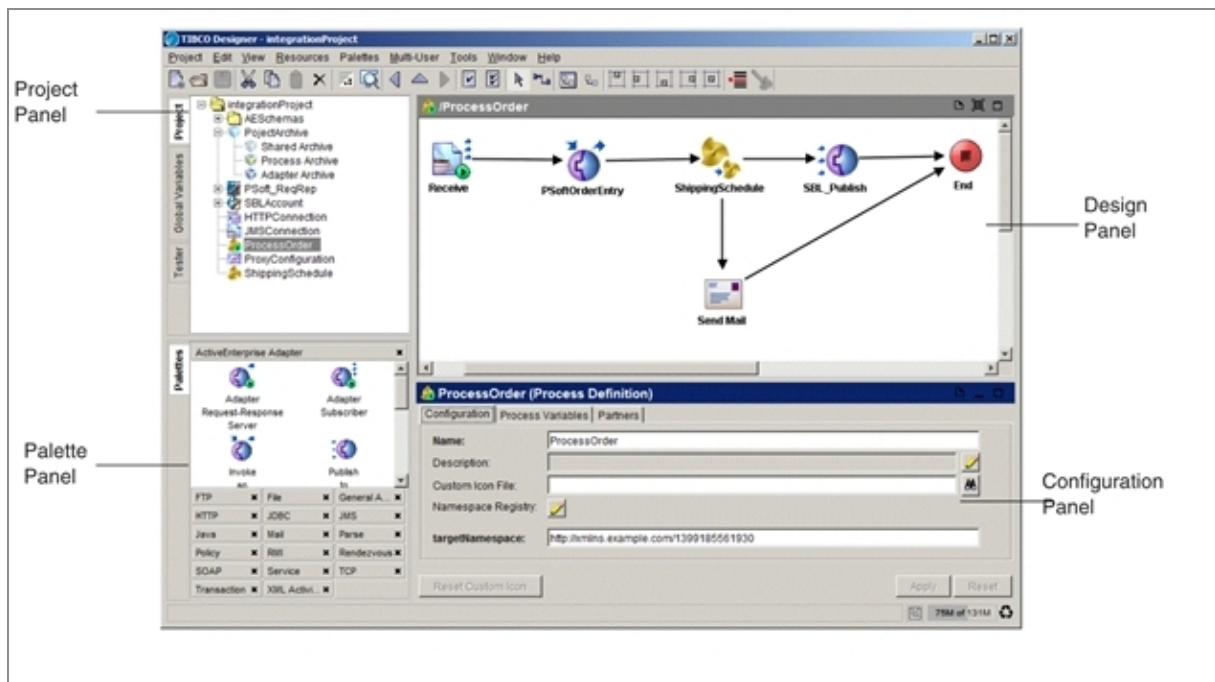
At design time, you work with the TIBCO Designer GUI to configure adapter services and design business processes. You design a business process by dragging activities (e.g. Read File or Send Mail) into the design window and joining the activities using transitions. The TIBCO Designer test mode allows you to debug the business process.

You can provide input, add breakpoints, supply values for variables, and so on. For more information, see *TIBCO Designer™ User's Guide*.

TIBCO Designer Layout

The TIBCO Designer main window has three or four panels that contain the design-time components of an integration project. You can configure the TIBCO Designer GUI to either display the project and palette panels separately or together. The TIBCO Designer GUI is discussed in more detail in the *TIBCO Designer™ User's Guide*.

Figure 6: TIBCO Designer main window



Projects

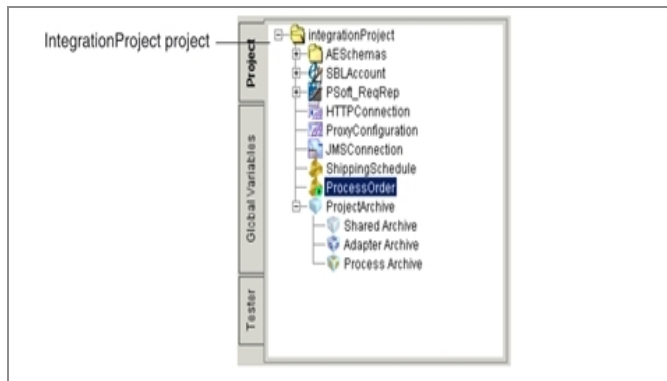
A project consists of resources that contain the functionality needed for your integration system. This includes services (producers and consumers of information) and any business logic that may be applied to that information.

In TIBCO Designer, you click the project folder to display the project's resources. The IntegrationProject project, shown in the project tree panel in [ActiveMatrix BusinessWorks project and resources](#), consists of several components:

- A JMS shared resource (JMSConnection)
- Two process definitions (ProcessOrder and ShippingSchedule)
- A shared resource used by the SOAP activity (HTTPConnection)
- A Siebel adapter and a PeopleSoft adapter (SBLAccount and PSoft_ReqRep)
- An Enterprise Archive for the project (ProjectArchive)

For a description of the example scenario that was used as the basis for this project. For more information, see [Business Integration Scenario](#).

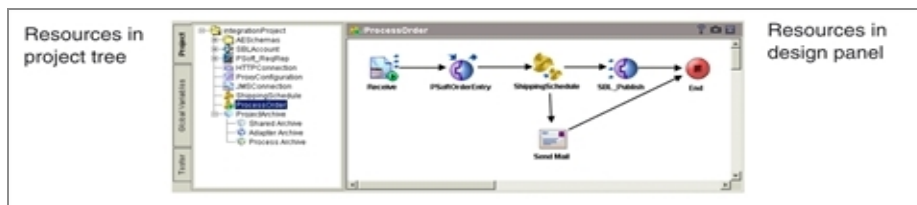
Figure 7: ActiveMatrix BusinessWorks project and resources



Resources

Resources are the components of a project. A TIBCO Designer resource corresponds to an object in a TIBCO application, such as an FTP activity, a process definition, or a specific adapter instance.

Figure 8: Resources in project tree and design pane



Palettes

Context-sensitive palettes organize resources into related groups. Which palette is displayed depends on the currently opened resource and on your preferences. You drag and drop resources from the palette into the design panel to add them to your project. The main window shown in [TIBCO Designer Layout](#) has several palettes in the palette panel.

Note: You can display the palette panel separately from the project panel or together with the project panel. When the palette panel and project panel are combined, a tab named Palettes appears in the project panel that allows you to access the palettes that are currently available. If you wish to display the panels separately, you can modify your viewing preferences on the Edit > Preferences menu.

Enterprise Archive

The Enterprise Archive resource allows you to create an Enterprise Archive file (EAR file) that you can use to deploy the project. The EAR file contains shared archives and process archives that you specify. These archives contain the adapter configurations and process definitions you wish to deploy. After saving the Enterprise Archive file, you can send it to the machine where the TIBCO administration server resides. TIBCO Administrator can use the EAR file to create a deployment configuration.

i Note: For more information about creating Enterprise Archive files and deploying projects, see *TIBCO ActiveMatrix BusinessWorks™ Administration*.

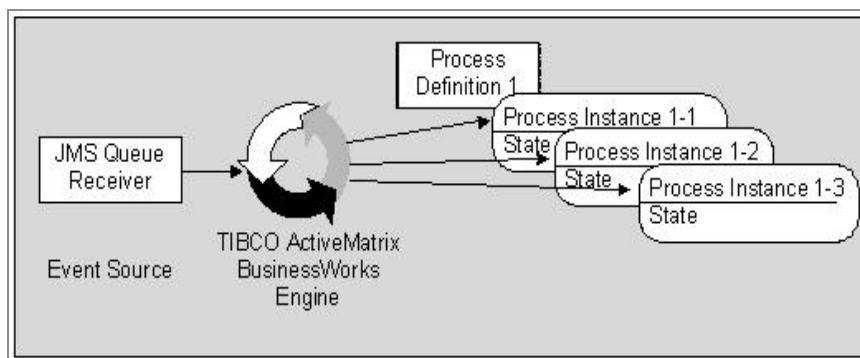
Run-Time Architecture

When the integration project is deployed, the different process engines and adapters are ready to run on the machines in the TIBCO administration domain.

You deploy your project and start each component individually from the TIBCO Administrator GUI. After all adapters and process engines have been started, process instances are created by process starters. A process starter could be, for example, a File Poller or an Adapter Subscriber activity waiting for incoming data. When data arrives, the process starter creates a process instance using the process definition to which it belongs, and the activities in the process are executed in sequence.

In [Process instances created from a process definition](#), a JMS Queue Receiver activity creates an instance of the process definition to which it belongs each time it receives input.

Figure 9: Process instances created from a process definition



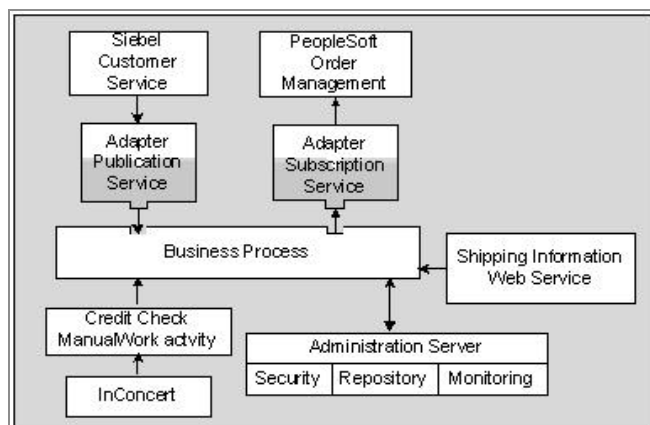
While different process instances are running, any alerts that were scheduled during deployment configuration are sent to the specified recipient by the TIBCO Administration server. In addition, the TIBCO Administrator GUI allows monitoring of the running project at different levels of detail, and can collect tracing information for later analysis.

For the example discussed in this manual, the process engine could perform these tasks:

- Receive data from an application server via JMS, data from a PeopleSoft Order Management System via the appropriate adapter, and data from a shipping service via SOAP.
- Enter data into a PeopleSoft Order Management system and data into a Siebel customer service system via the appropriate adapters.
- Send certain orders out for credit approval and receive approval or refusal.

All components are monitored and managed by way of TIBCO Administrator, which also provides security and repository management. Users can access TIBCO Administrator using the TIBCO Administrator GUI.

Figure 10: Example scenario data flow



ActiveMatrix BusinessWorks Features

This section discusses some ActiveMatrix BusinessWorks features.

Messaging

To support your integration project at run-time, you need a messaging system that can reliably handle the volume of messages that will be sent and received. The system should have these characteristics:

- **Guaranteed delivery and fault tolerance**—Message delivery must be guaranteed, and the system must be fault tolerant. If a message cannot be delivered because the recipient was unavailable, the messaging system must queue that message and continue to operate. The queued message must then be redelivered as appropriate.
- **Distributed architecture**—A distributed, loosely coupled system is much more likely to support the fault-tolerance you require than a monolithic system that depends on one centralized server.
- **High throughput**—High throughput without performance degradation is needed. Requirements vary throughout the day and throughout the business year, and you cannot afford performance degradation at the time when business increases.
- **Scalability**—As your business grows, you want to be able to update your business integration in a simple and cohesive way. Furthermore, you want to be able to connect your integration project with other departments using a similar system. The messaging system must support this scalability.

ActiveMatrix BusinessWorks is based on messaging standards with proven track records. Supported protocols include TIBCO Rendezvous, JMS, and HTTP.

Adapters

Business information is distributed among different business applications (such as SAP R/3 or PeopleSoft) or available from databases or files. Adapters help make this information available to the business process by "adapting" the applications to a common messaging system.

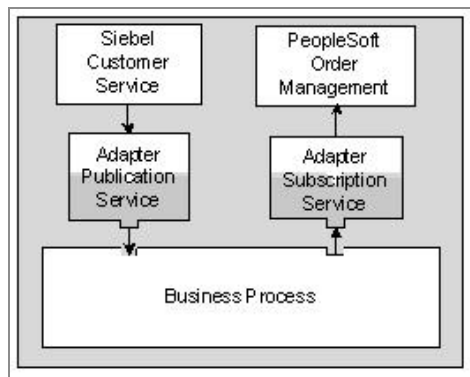
What are Adapters?

Adapters translate information into the appropriate format:

Adapters receive information from a source application and publish it to the business process in a shared format. Adapters subscribe to information from a business process and translate it to a format the target application can understand. Adapters can also be set up to work in a client/server mode (using remote operations.)

The illustration below shows how a Siebel customer service system communicates with the business process using an adapter publication service and the business process communicates with the PeopleSoft Order Management system using an adapter subscription service.

Figure 11: Adapter data flow



In ActiveMatrix BusinessWorks, adapters provide services to activities inside the business process.

Adapter Features

Companies in a wide range of industries have successfully used TIBCO adapters to integrate different packages and custom applications. ActiveMatrix BusinessWorks includes second-generation adapters that are based on the same technology but have the following new features:

- **Easy Configuration With Design-Time Adapter**—All adapters included with ActiveMatrix BusinessWorks use a unified GUI that simplifies adapter configuration. A Design-Time Adapter allows you to directly access the source application from the TIBCO Designer GUI and specify the data the business process needs.
- **Easy Inclusion in Business Processes**—The business process can communicate with adapters by using activities found in the adapter palette. These activities interact with each of the standard adapter services. For example, the Publish to Adapter activity sends a message to an adapter subscription service.
- **Easy Deployment and Monitoring**—When you are ready to deploy your project, the TIBCO Designer deployment palette allows you to assign each adapter to its own machine. At run-time, the TIBCO Administrator GUI monitors each adapter in its own panel. You can therefore easily see if one of the adapters is a bottleneck in the

business process flow.

ActiveMatrix BusinessWorks Adapters

A fully integrated ActiveMatrix BusinessWorks adapter can be installed into a TIBCO administration domain and later monitored and managed from the TIBCO Administrator GUI. ActiveMatrix BusinessWorks fully integrates with the following adapters:

- **Technology adapters**—Includes adapters that access files or databases.
- **Application adapters**—Includes adapter for PeopleSoft, SAP R/3, Siebel, and others.

Other TIBCO Adapters can be loaded into TIBCO Designer and configured using the Generic Adapter Configuration resources. These adapters can then be used in process design and run as part of the integration project. They cannot, however, be installed into the TIBCO administration domain or monitored and managed via TIBCO Administrator.

For more information, see [Phase 3: Services Configuration](#).

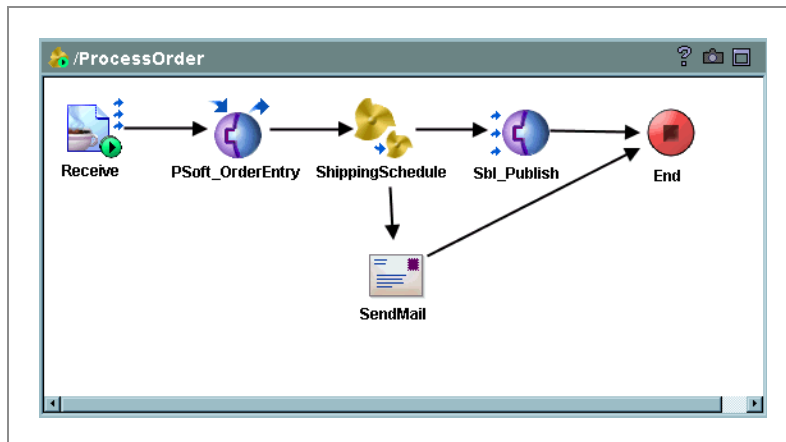
Business Process Modelling

The business processes describe the actual flow of data inside the enterprise. In ActiveMatrix BusinessWorks, you use the TIBCO Designer GUI to design and test your processes. Features include:

- Configuration of adapter services.
- A complete set of commonly used activities such as File Read, File Write, and File Create, a set of email activities, timers, FTP activities, and so on.
- A transformation tool that lets you map the output of one activity to the input of subsequent activities.
- Conditional transitions supporting XPath syntax.
- Grouping of activities.
- An easy-to-use design-time process debugger.

For more information, see [Phase 4: Business Process Design](#), and the *TIBCO ActiveMatrix BusinessWorks™ Process Design*.

The illustration below shows a simple process that is part of the example scenario in the design window.

Figure 12: Example process

Schemas and Data Mapping

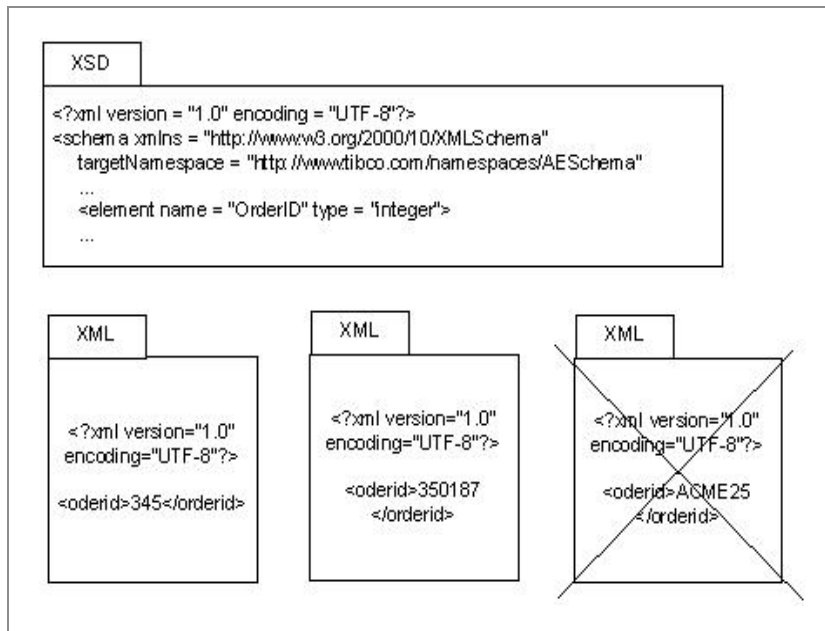
Different applications in your enterprise use different data representations. For example, a purchase order in a PeopleSoft system differs from a purchase order in a Siebel customer service system. ActiveMatrix BusinessWorks allows you to view and manipulate the data coming from and going into each service or activity using XML schemas.

This section first gives a brief introduction to schema ([Understanding Schemas](#)), then discusses "[Schemas in ActiveMatrix BusinessWorks](#)."

Understanding Schemas

The example below shows a simplified XSD (XML Schema Definition) that includes an Order ID element restricted to integer data. Incoming XML documents that use integers for the Order ID are allowed, while an alphanumeric Order ID is rejected.

Figure 13: XML files conforming or not conforming to XSD



Schemas are especially useful if you are deploying a complex system. Schemas are used by the running application but are not included in the code. The use of schemas makes it possible to enforce that outgoing and incoming data strictly comply with the prespecified data description.

Schemas in ActiveMatrix BusinessWorks

In the TIBCO Designer GUI, you can define the schema for adapters and view and manipulate the schema for each activity in the business process.

For business process activities, you can view the available process data and define the input schema for each activity. The process data is the list of available data for that activity. The input schema (required or optional) defines input values for an activity.

You can map the process data to the input data using a drag and drop interface. You can specify conditional mapping using XPath, and you do not need detailed knowledge of XPath for simple conditions. For more information, see [Step 7: Perform Mapping and Transformation for Each Activity](#).

Data mapping is discussed in detail in the *TIBCO ActiveMatrix BusinessWorks™ Process Design*.

Manual Activities

ActiveMatrix BusinessWorks includes a ManualWork palette with activities that you can add to your business processes when the process requires user interaction for completion. In our example, orders under \$10 000 were processed automatically. For orders over 10 000, an additional credit check is required.

In that case, the order is assigned to a pool of users for approval. One user accepts the request, and approves or rejects it. If no one accepts the request, the manual approval times out, and then the status of the request is checked. If no errors were returned, then the work is still in the users' queue, so the process waits for the completion of the manual work. If errors were reported in the manual work, the work is marked as not approved and the process completes.

ActiveMatrix BusinessWorks allows you to:

- assign a task to a pool of users,
- check the status of the task,
- change the status of the task,
- download documents associated with a task,
- or wait for the completion of a task.

The ManualWork palette works with TIBCO InConcert. Users and groups are defined either in TIBCO InConcert or TIBCO Administrator (and then later exported to TIBCO InConcert). An activity that assigns work creates a TIBCO InConcert job. The job can be viewed and modified using ActiveMatrix BusinessWorks web interface to manual tasks.

ActiveMatrix BusinessWorks Methodology

A ActiveMatrix BusinessWorks integration project is developed in phases. Having a well-defined methodology helps new users come up to speed quickly and allows different developers to work together more easily.

This section gives an overview of ActiveMatrix BusinessWorks methodology.

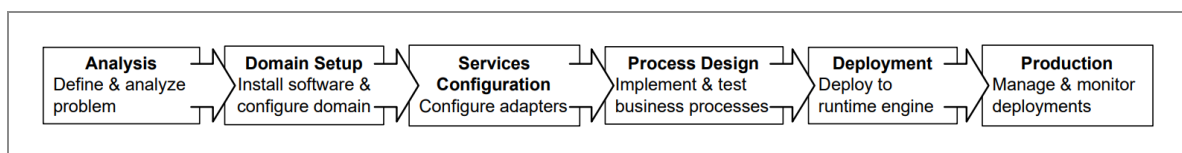
Introduction

ActiveMatrix BusinessWorks components are designed to support development in phases and to let you seamlessly move from one phase to another.

- Using TIBCO Designer, you configure services, for example, an adapter service.
- You can then access the adapter service from activities inside the business process.
- After you've configured adapter services and business processes, you can use TIBCO Designer to assign adapter services to adapters and processes to process engines. You assign each adapter and process engine to a machine in the TIBCO administration domain and deploy the project to the run-time environment.
- You can then start and stop the adapters and process engines using the TIBCO Administrator GUI and manage and monitor them from there.

Following the phases in sequence results in a fast deployment that closely meets the specifications. Note that as a rule, you perform analysis, installation, and services configuration only once, then iterate through the other phases until you have arrived at the optimal configuration.

This section gives an overview of each phase, using examples from the example scenario as appropriate (For more information, see [Business Integration Scenario](#)). A more detailed discussion for each phase is then given in a separate section.



Phase 1: Analysis

Problem definition and analysis is the critical first phase in an integration project. Because the ActiveMatrix BusinessWorks graphical user interface is so easy to use, it is tempting to start development right away. However, detailed problem analysis results in a faster overall development process. By clearly identifying and analyzing the problem, you avoid pursuing dead-end design paths and the steps to solve the problem become apparent.

As a rule, it makes sense to start with a business analysis that includes a problem definition, which states your project's goals clearly. An engineering analysis goes a step further and identifies the components of the integration project, the process flow, error handling, etc.

✓ **Tip:** The analysis should include consideration of expansion possibilities. In the example scenario, one could consider expansion to include direct communication with a business partner. Because of the ActiveMatrix BusinessWorks distributed architecture, most expansions are straightforward.

ActiveMatrix BusinessWorks implicitly supports analysis and design by offering a set of objects representing services and activities as the basis for the project flow. The design team can use these objects during project design. For more information, see [Phase 1: Analysis](#).

Here are some questions that are commonly asked during analysis:

- What are the services my business process will access? In the example, the process is accessing two adapter services (PeopleSoft and Siebel), the web service that supplies shipping information, and an application server.
- What are the transports being used? In the example, the adapter services are accessed using TIBCO Rendezvous. The web service is accessed via SOAP. The application service is accessed via JMS.

Phase 2: Domain Setup and Installation

A TIBCO administration domain is the set of software and hardware resources used by your integration project. For more information, see [TIBCO Administration Domain](#).

This section gives an overview of planning the TIBCO administration domain setup and installing the components.

- ✓ **Tip:** Domain setup is different during development and during deployment testing and production.
- ✓ **Tip:** During development, each developer may install an TIBCO Administration Server and set up TIBCO administration domain on their machine and develop and test the project there.
- ✓ **Tip:** During deployment testing and production, one TIBCO Administration Server manages the project and the ACL (Access Control List). Only authorized users can create and save projects or start and stop processes.

Planning the Domain

When you install a ActiveMatrix BusinessWorks¹ component, you must specify the TIBCO administration domain to which a machine belongs. Before installing the software, you should therefore determine what resources should belong to a TIBCO administration domain. Ask yourself these questions:

- For development environments, do I need to share work with other developers? If so, all machines that need to share work should be part of the same TIBCO administration domain. If not, this machine can have its own TIBCO administration domain.
- What machines do I need to run my project? By default, all machines within the TIBCO administration domain are expected to be in the same network subnet. You can, however, set up your system to use TIBCO Rendezvous rvr and can then use ActiveMatrix BusinessWorks across subnets.
- Which components (adapters or process engines) should run on which machine?
- Where should I run my TIBCO Administration Server?
- Who are the users that need to make changes to the project? For which component does each user need to make changes? (for example: start or stop an adapter)
- Who are the users that need to view information about the running project? Which

1

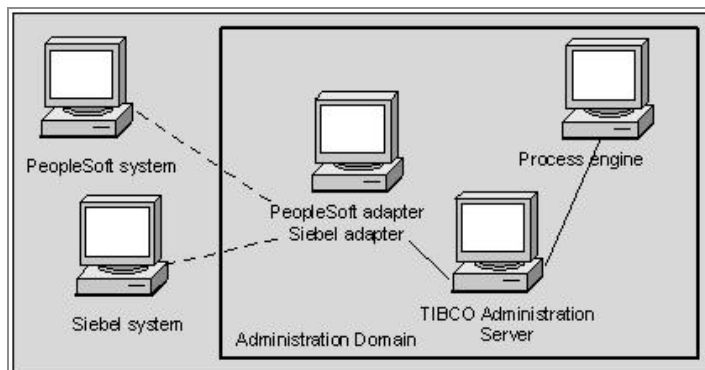
component(s) does each user need to view?

- How will the project handle load balancing and fail-over?

The illustration below shows a possible TIBCO administration domain setup for the example scenario:

- The TIBCO Administration Server runs on machine 1.
- The PeopleSoft and Siebel adapters run on machine 2.
- The process engine runs on machine 3.
- The PeopleSoft and Siebel systems run outside the TIBCO administration domain.

Figure 14: Domain setup for example program



Installing Components

A flexible installer allows you to install one or more components on each machine following these steps:

Procedure

1. Install TRA on each machine in your domain.
2. Install the TIBCO Administration Server and specify the TIBCO administration domain name, and the administration user and password.
3. Install other ActiveMatrix BusinessWorks process engine(s) into TIBCO administration domain.
4. Install adapters into TIBCO administration domain.

Phase 3: Services Configuration

ActiveMatrix BusinessWorks uses different types of services that can be accessed from within the process:

- Adapter services are configured using TIBCO Designer and the Design-Time Adapter (DTA). The services can then be accessed from the ActiveMatrix BusinessWorks process.
- Web services can be configured from ActiveMatrix BusinessWorks or externally, and accessed using the SOAP Request-Reply activity.

This section gives an overview of adapter configuration. For more information, see [Phase 3: Services Configuration](#). Then, it discusses the services used by the example.

Adapter Configuration Overview

Adapter services send data to and receive data from your business process. Data can come, for example, from an application such as PeopleSoft, SAP R/3, and so on, from a legacy source, from a database, or from the Internet. The following service types are supported:

- Adapter publisher service—Sends data from the source application to the business process.
- Adapter subscriber service—Receives data in the source application from the business process.
- Adapter client service—Acts as a client in a request-response interaction.
- Adapter server service—Acts as a server in a request-response interaction.

When designing and implementing the integration project, you must identify the adapter services precisely. This includes connection parameters such as the names of the host application, their location, and so on.

You then configure a service with the appropriate adapter and save it as part of your project. To configure an adapter first provide connection information. A Design-Time Adapter (DTA) allows you to access the metadata provided by the adapter at design time. Each type of adapter has its own DTA. Only one DTA needs to run in a network, even if several users access different adapter instances of that adapter type. With configuration complete, you save the adapter service configuration. The adapter service can then be used by the activities in your business process and later be invoked at runtime. For

example, a Publish to Adapter activity expects an adapter subscriber service that receives the data being published.

For more information, see [Phase 3: Services Configuration](#). The User's Guide for each adapter contains detailed set-up information for that adapter.

Web Services Overview

ActiveMatrix BusinessWorks can function both as a server and a client in a web services interaction.

For more information, see the SOAP palette activities and the WSDL file activity in the *TIBCO ActiveMatrix BusinessWorks™ Process Design*.

Services Used by the Example

The example needs to access four services from which it retrieves or to which it sends information:

- The application server is configured externally. It is accessed via a JMS Queue Receiver.
- To access the PeopleSoft Order Management system, you configure a PeopleSoft request-response adapter. The adapter is accessed via an Invoke an Adapter Request-Response Service activity.
- To access the Siebel customer service system, you configure a Siebel adapter subscriber service. A Publish to Adapter activity can then connect the process to the adapter subscriber service.
- The Shipping web service is configured externally. It is accessed via a SOAP Request Reply activity.

Phase 4: Process Design

The flow of business data in your enterprise can be captured by business processes. This section starts with an overview. It then briefly discusses the activities used in the example scenario. For more information, see [Phase 4: Business Process Design](#).

Overview

The TIBCO Designer GUI supports defining business processes with these major elements:

- Each process has a starting and ending point.
- Activities are added to the process. Activities can access data from an adapter service, manipulate the data, and send the data elsewhere. Examples include sending email, querying a database, or adding content to a file.
- The process can choose from different execution paths depending on certain criteria. For example, if the amount of a purchase order exceeds a certain number, you could add an additional credit check.
- Activities can be grouped. Grouping allows you to create loops. These loops can be used, for example, to have one error condition for the group, or to group activities as transactions that commit to a database only when all activities in the group are completed.
- Most processes have one main process, which starts with a process starter activity. Different activities, for example, a SOAP Event Source activity or a Receive Mail activity can function as process starters. A process can call different subprocesses as it executes.
- If the business process requires user interaction, for example, approval of certain orders, it is possible to use the activities in the Manual Work palette. Manual Activities interact with TIBCO InConcert and allow a pool of users to accept outstanding tasks. For more information, see *TIBCO ActiveMatrix BusinessWorks™ Palette Reference*.

An integral part of process design must be testing. TIBCO Designer includes a test mode that allows you to run any of the processes in your project at design time. You can set breakpoints and provide required input as needed. You can also see the values of variables as they are passed through the different activities in the process.

Activities Used by the Example

The example includes all the services (for more information, see [Services Used by the Example](#)). In addition

- A Send Mail activity sends an email to the customer if shipping the order is delayed.

- A Manual Work activity handles the credit check approval.

Phase 5: Deployment Configuration and Deployment

The TIBCO administration domain supports a simple installation and deployment paradigm:

During installation, you install all components into the same TIBCO administration domain. After you have installed the TIBCO Administration Server, any machine on which you install a ActiveMatrix BusinessWorks core component or an adapter can be added to the TIBCO administration domain. The component or adapter is then visible and accessible at design time by way of the TIBCO Designer GUI and at runtime by way of the TIBCO Administrator GUI.

- You create an Enterprise Archive file (EAR file) in TIBCO Designer that contains the adapter configurations and process definitions you wish to deploy.
- The EAR file is used by TIBCO Administrator within a deployment configuration. A deployment configuration allows you to assign processes and adapter services to different process engines and adapters installed on the machines in the TIBCO administration domain.
- When deployment configuration is complete, you deploy the project. As part of that process, startup scripts and other information about the different components are sent to the machines to which the components were assigned. The project data store (repository) and the TIBCO Administration Server are updated with the new deployed components.

TIBCO ActiveMatrix BusinessWorks™ Administration discusses deployment in more detail.

Phase 6: Production

In the production phase, your project's components are running on the different machines in the TIBCO administration domain. Recovery is performed automatically as previously specified as part of the deployment configuration.

Authorized users can monitor the TIBCO administration domain, all machines, and all processes, using the web browser based TIBCO Administrator GUI. TIBCO Administrator can be used for these tasks:

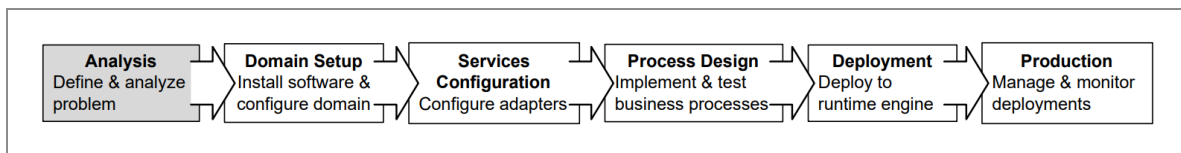
- **User Management**—Manage the ACL, for example, create users for the TIBCO administration domain and assign them permissions to perform certain activities. Change the ACL as needed.
- **Domain Monitoring**—View the machines in the TIBCO administration domain and their CPU and disk usage. View a domain inventory of all TIBCO products installed in the TIBCO administration domain.
- **Deployment Monitoring and Management**—View the status of components and generate tracing information. Start and stop process engines and adapters.

[Phase 6: Production](#), gives an overview of the most important components of the GUI. For more information, see *TIBCO ActiveMatrix BusinessWorks™ Administration*.

Phase 1: Analysis

During the analysis phase, the different departments participating in the integration project must come to a clear understanding of the requirements.

This section gives a brief introduction to elements of the analysis phase that typically require special attention.



Step 1: Define and Delimit the Problem

The ultimate success or failure of your business integration depends on how clear you are about the problem you are trying to solve. You can only succeed if you understand the scope of the problem, the components involved, and the deployment parameters.

Before you start, it is therefore critical that you have a definition of the problem that is as clear and precise as possible.

For example, a working definition of the example scenario could be the following:

Procedure

1. EasyWare receives orders for computer hardware. The goal is to allow customers to place orders through a web site, and to allow later expansion for other ways to place orders.
2. Each order is processed by a customized order capture system built on top of an application server.
3. The business process receives the incoming order document via JMS.
4. Each order is automatically entered into the PeopleSoft Order Management system. Conversion to PeopleSoft format is required.
5. When the PeopleSoft Order Management system acknowledges acceptance of the

order, it includes an ID for the order.

6. Next, the business process checks the shipping schedule, which is available from an external vendor's web site.
7. If shipping is delayed, an email is sent to the customer.
8. Otherwise, the order is sent and all order information, including the generated Order ID and the shipping information, is entered into a Siebel system. The Siebel system creates a new customer service record based on the information.

Step 2: Identify Processes

A clean design of a complex process consists of a main process that calls sub processes as needed. The problem defined above lends itself to the creation of a process for each task that is performed.

Procedure

1. The main process listens for incoming data from the application server. When a purchase order arrives, the main process calls the other processes in sequence.
2. The `AddtoOrderMgmt` process enters the order into the PeopleSoft Order Management system. It returns the purchase order and the Order ID to the main process.
3. The `CheckShippingSchedule` process accesses the web site of the shipping company and sends an email to the customer if shipping is delayed. This process could also update the status of the order in the Order Management system.
4. The `AddtoCustMgmt` process enters the order, including the Order ID generated by PeopleSoft and the shipping information, in the company's Siebel customer service system.

You can use TIBCO Designer to create the four processes and have the main process call the other processes in the appropriate sequence.

Step 3: Identify Components

After you have identified the processes, you must understand the components of the process. This section discusses some potential components.

- ✓ **Tip:** Many components are activity resources. A complete list of all activities is included in the *TIBCO ActiveMatrix BusinessWorks™ Process Design*. You can also access documentation for an activity from TIBCO Designer using the **What is This** option from the right-button menu of the corresponding resource's menu.

Shared Resources

Some activities use shared resources. For example, a `WSDL File` shared resource is used by SOAP activities and a `JDBC Connection` shared resource is used by JDBC activities.

Services and Corresponding Activities

The business process uses services to retrieve or send data. There are two kinds of services:

- **Adapter services**—The source and target applications your enterprise uses cannot directly talk to each other or to ActiveMatrix BusinessWorks. To enable this communication, you can use TIBCO adapters. Adapters can be configured using TIBCO Designer.
- **Web services**—Web services are external to ActiveMatrix BusinessWorks but are supported by several activities that are part of ActiveMatrix BusinessWorks.

The example discussed in this manual requires the following activities:

- An application server sends the incoming orders over JMS. The business process uses a `JMS Queue Receiver` activity that receives the order.
- A PeopleSoft adapter receives a request and sends data back to the process. For the problem at hand, a request-response adapter service is well suited. The business process uses an `Invoke an Adapter Request-Response Service` activity to interact with the adapter.
- A web service provided by the shipping company is invoked through a `SOAP Request Response` activity.
- A Siebel subscriber adapter service receives the order from a `Publish to Adapter` activity and enters the order, including the Order ID assigned by PeopleSoft, into the Siebel customer service system.

Other activities include a Send Mail activity that sends an email to the customer if shipping is delayed.

ManualWork Activities

The activities in the ManualWork palette are useful for automated business processes that have a few steps which require user interaction. In many cases, the Assign Work activity is appropriate for implementing the interaction. Other activities in the palette include, for example, a Modify Work activity that allows administrators to perform actions on a work item.

In order to use the activities in the Manual Work palette, TIBCO InConcert must have been installed and users must have been created with TIBCO Administrator and exported to InConcert.

For more information, see [Step 5: Optionally, Add Manual Work Activities](#) and *TIBCO ActiveMatrix BusinessWorks™ Palette Reference*.

Transitions and Conditions

Transitions go from each activity in the process to the next activity. Each activity must have at least one incoming and at least one outgoing transition. Each activity can potentially be called from several other activities or can conditionally call several activities.

For example, if a credit check were included in a business process, the result of the credit check outcome could determine the next activity. A negative credit check could result in an email to the customer. A positive credit check could result in a different email and placement of the order.

Mapping

For each activity in the process there is an appropriate input. For example, the Send Mail activity needs input that includes information about the customer and the Order ID. At times, the output of one activity directly maps into the input of the next activity in the process. Very often, however, an activity requires a subset, or a superset of the incoming process data, or data may need to be modified.

To give each activity the appropriate input, ActiveMatrix BusinessWorks lets you map the process data to the input of the activity. For example, the Send Mail activity uses the customer email address and the Order ID but ignores the shipping address which is also part of the order.

Exceptions

As you analyze your business problem, you should include as much information about exceptions as possible. For the employees of your company, graceful exception management will result in a noticeable increase in productivity. If exception handling is flawed, integration automation might actually result in decreased productivity because a lot of time is spent on dealing with exceptions.

If exceptions are included in your initial design, you can make them part of the business process, and ultimately deliver a more robust system faster.

Decide on exception handling standards across business phases to make it easier to identify exceptions and understand how they are related.

Step 4: Describe Business Events and Objects

Early in the analysis, you identified the business objects and events. At this stage of the analysis, you must describe the events and objects in more detail. This includes:

- Know the required data format at each place in the process (and how to get it).
- Know the required data content (and how to get information that is not directly available).
- Understand what appropriate actions in case of a business or system-level error could be.

Step 5: Design Business Processes

After you have identified and described the components of the process, you can prepare the actual design of the business process. This includes:

- **Activities involved.** For example, the purchase order is received from a `JMS Queue Receiver` activity and a `Publish to Adapter` activity sends the purchase order to a PeopleSoft adapter subscriber service. Data flow. Data flow includes both flow of data for the non-exception case and for exception cases.
- Data flow may involve mapping, that is, putting the value from one field into another. It could also involve transformation, that is, translating data from one format to another. For example, PeopleSoft may store phone numbers in a 10-digit format, while Siebel may use parentheses or dashes as part of the phone number.
- Part of massaging the data for each system requires transformation. You can set up transformation using the mapper included with TIBCO Designer. Custom java activities can be used to perform more complex transformation.
- **Exception flow.** After you have identified the possible exceptions, you determine what the system should do in case of an exception.

Step 6: Consider Domain Setup

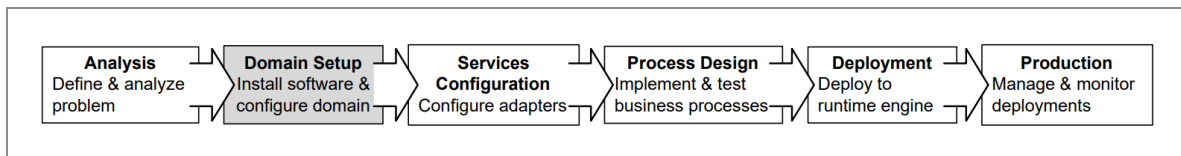
After you've completed the design of your business process, you should consider the domain setup required to support it. For example:

- What hardware is required to run the project? Is security an issue that might influence platform choice?
- What TIBCO software components do you expect to install?
 - How many versions of TIBCO Designer and the ActiveMatrix BusinessWorks engine during design time.
 - How many versions of the ActiveMatrix BusinessWorks engine at runtime? If you want to run in fault tolerant mode, you should plan on appropriate hardware resources to support it.

Phase 2: Domain Setup

ActiveMatrix BusinessWorks allows you to progress from project design to deployment and production with minimal configuration and setup. To guarantee that the transitions will be trouble-free, it is important that you understand setup of a TIBCO administration domain and installation of different TIBCO products into the TIBCO administration domain.

This section gives an overview of domain setup. For more information, see *TIBCO Administrator User's Guide*.



TIBCO Administration Domain

A TIBCO administration domain is a set of hardware and software resources. [TIBCO Administration Domain](#) gives an overview of a domain's elements.

Understanding the domain before installing the software is important because the domain is set up during installation. Typically, the set-up depends on the phase of the project:

- During early stages, each developer typically installs the complete ActiveMatrix BusinessWorks package, including the TIBCO Administration server, on one machine and develops and tests the project there.
- Later, for final testing and production, the development team uses one TIBCO Administration Server which manages different components (engines and adapters) running on different machines.

The advantage of TIBCO administration domain is that the components of your integration project automatically know about each other. They can be monitored as a group. Security, that is, access to both processes and data stores, is shared across the TIBCO administration domain.

The TIBCO administration domain is managed by a TIBCO Administration Server, which is assisted by a TRA running on each machine in the domain.

TIBCO Administration Server

The TIBCO Administration Server manages the TIBCO administration domain. The server is installed only on one machine in the environment and can be accessed with the browser-based TIBCO Administrator GUI.

The TIBCO Administration Server's main responsibilities are the following:

- Enforce security for the domain. ActiveMatrix BusinessWorks supports both authentication and authorization of users that want view access or full access to the run-time components.
- Manage registration, that is, add deployed projects and machines to a domain.
- Send appropriate information to each machine's TRA when a project is deployed, and pick up alerts sent by those Runtime Agents.
- Start and stop process engines and adapters.
- Manage engines or adapters running in fault-tolerant mode if fault-tolerant setup has been performed.

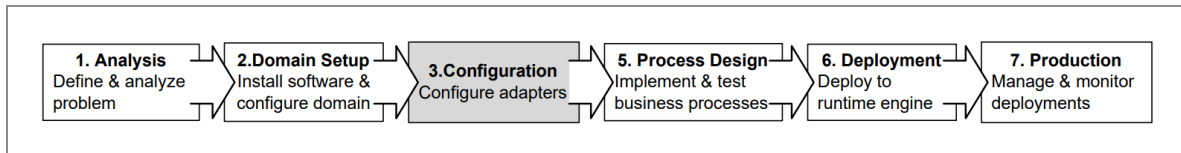
TRA

When you install a ActiveMatrix BusinessWorks component or a fully integrated TIBCO ActiveEnterprise adapter on a machine, a TRA is automatically installed. The TRA has two main functions:

- Supplies an agent that is running 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 via TIBCO Administrator.
- Supplies the run-time environment, that is, all shared libraries including third-party libraries.

Phase 3: Services Configuration

The first phase in the actual business integration development is configuration of adapter services. Adapters ensure that the different applications in your enterprise can communicate in real time.



Introduction: ActiveMatrix BusinessWorks Services

In ActiveMatrix BusinessWorks, services are responsible for publishing or subscribing to business data in a decoupled yet reliable manner. The business process receives data from a service and routes data to a service. Examples of services are could be a `SendPurchaseOrder` activity or a `RequestShippingInformation` activity. ActiveMatrix BusinessWorks supports web services for interaction with the Internet and adapter services for interaction with files, databases, or different ERP (Enterprise Resource Planning) applications.

Enterprise integration platforms typically use a tightly coupled architecture, where all components of a business process are running on one server. If any component in the system changes, the system no longer works. If the server goes down, all components of the business process stop.

In contrast, a service-centric architecture supports self-contained services. Each service is configured separately and can be deployed on a different machine. If one machine goes down, all other parts of the process can still run. Service descriptions are stored with the TIBCO Administration Server. The business process coordinates the services in the appropriate way. For example, a process could contain a `Publish to Adapter` activity that accesses an adapter subscriber service.

This loosely-coupled architecture makes it easy to change individual components as needed.

Service Characteristics

The following characteristics are associated with a service:

- Service interface, for example WSDL/SOAP or AE Services
- Transport, for example, HTTP, JMS, TIBCO Rendezvous
- Data syntax, for example, XML or ActiveEnterprise message format
- Data schema, for example DTD or XSD schema

Invocation Modes

Services can be invoked in several ways.

- A one-way operation is executed once and does not wait for a response.
- A request-response operation is executed once and waits for one response. In a request-response service, communication flows in both directions. The complete interaction consists of two point-to-point messages—a request and a response. The interaction is only considered complete after the response has arrived.
- Publication (notification) means an operation sends information on an as-needed basis, potentially multiple times.
- Subscription means incoming information is processed on an as-needed basis, potentially multiple times.

Publication and subscription are driven by events, usually the arrival or creation of data. Communication is in one direction (publisher to subscribers).

Service Types

ActiveMatrix BusinessWorks includes both web services and adapter services. Web services are accessed by SOAP activities. Adapter services are accessed by activities available in the TIBCO Designer ActiveEnterprise Adapter palette that you can add to your business process.

Web Services

ActiveMatrix BusinessWorks supports Web Services Description Language, or WSDL. WSDL is an XML-formatted language used to describe a Web service's capabilities as collections of communication endpoints capable of exchanging messages.

You can use ActiveMatrix BusinessWorks both to set up a web services server or to set up a web services client. The activities you need are included in the SOAP palette.

Adapter Services

Adapter services allow your business process to publish data or subscribe to data used by the enterprise. A publication service sends data to the business process, a subscription service receives data from the business process.

Adapter services allow you to communicate with enterprise applications or interact with other incoming and outgoing data. They include:

- **Technology adapters**— Allow publication to and subscription from files and databases.
- **Enterprise application adapters**—Allow you to interact with enterprise applications such as Siebel, SAP R/3, and PeopleSoft.

Adapter Service Configuration Steps

Configuring an adapter service consists of these steps, discussed in this section:

Step 1: Installing the Adapter

Different business processes have to connect to different enterprise applications. ActiveMatrix BusinessWorks therefore does not include adapters in its base package. Instead, you install the adapter you need separately and add it to the TIBCO administration domain during installation. Adapter installation therefore consists of these steps:

Procedure

1. Install ActiveMatrix BusinessWorks. As part of the installation, specify the name of the

TIBCO administration domain and specify the administrative user and password for the TIBCO administration domain.

Note: You must establish the TIBCO administration domain and it must be running and accessible before you install the adapter.

A TIBCO administration domain is a set of hardware and software resources. TIBCO administration domain is set up to include one or more machines. For more information, see [TIBCO Administration Domain](#).

2. Install the adapter. During installation, specify the (already existing) TIBCO administration domain and the administrative user and password.

After installation, a palette for that adapter becomes available from TIBCO Designer the next time you start it.

Step 2: Setting up the Design-Time Adapter

The TIBCO Designer GUI allows you to connect with the source or target application for the adapter at design time using a design-time adapter. After you have established design-time connection parameters, you can use the TIBCO Designer GUI to specify schema information from the adapter.

To set up the design-time adapter, follow these steps:

Procedure

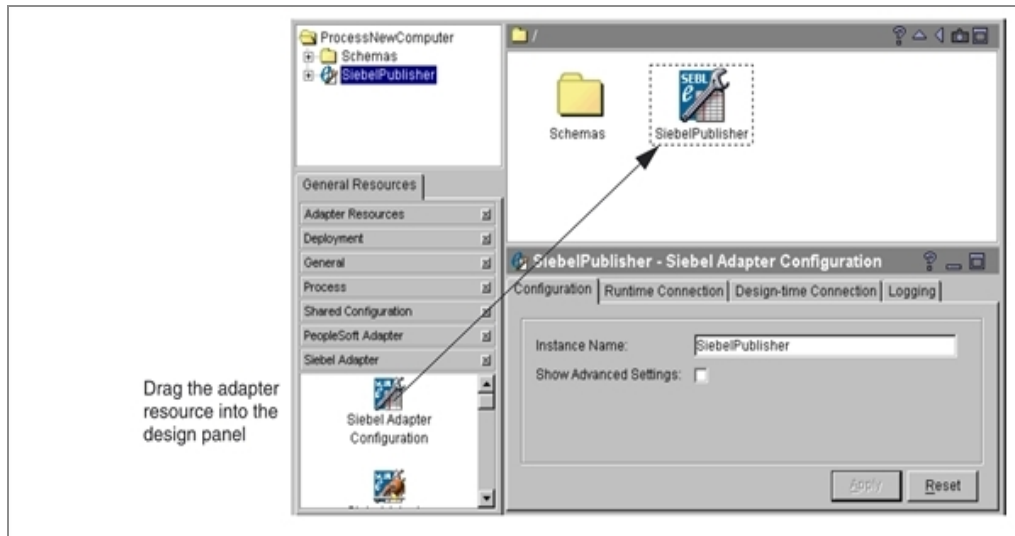
1. Launch the TIBCO Designer GUI.

The appropriate adapter palettes should now be included.

2. Select the adapter palette and drag an adapter resource from the palette panel into the design panel and name the adapter service.

[Figure15: Adding an adapter to the project](#) shows how you would drag a Siebel Adapter Configuration into the design panel. The adapter service has been named SiebelPublisher.

Figure15: Adding an adapter to the project



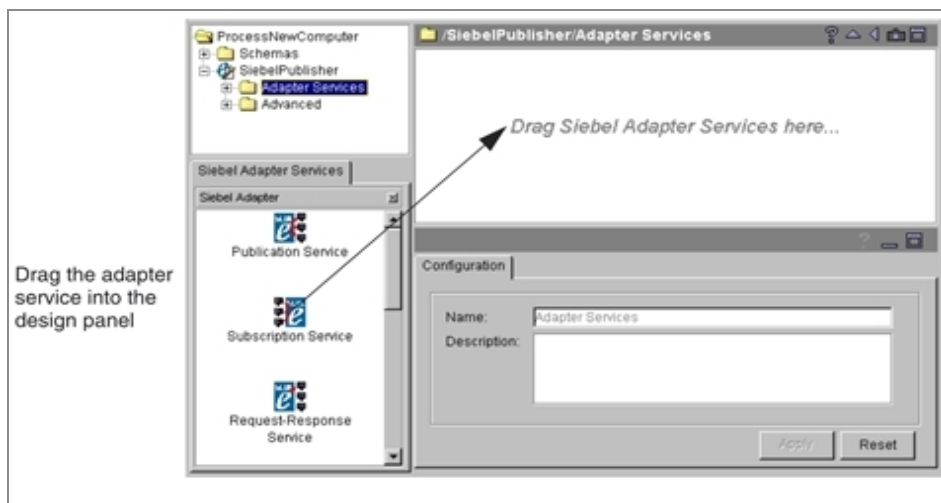
3. Specify connection information for the application you want to access so your design-time adapter can connect.
4. In the project tree panel, double-click the adapter, then select its Adapter Services folder to open it.

In the palette panel, the Services palette is displayed.

5. Drag an adapter service, for example, a publication service, from the palette panel into the design panel.

The configuration panel is updated to allow you to configure the service.

Figure16: Adding a service to an adapter



6. From a command prompt, start the design-time adapter.

7. Specify the data the adapter should publish or subscribe to interactively.
8. Save the adapter.

Step 3: Configuring the Run-Time Adapter

With the design-time adapter running, you can configure the run-time adapter. You configure each adapter service separately using the tabs in the configuration panel as follows:

- Specify run-time connection information using the `Runtime Connection` tab. You can either specify the same information as that used by the design-time adapter, or different information. For example, you can specify a different host machine or user name or password.
- Define adapter services and choose schema from the pop-up list provided via the design-time adapter.

The exact process for defining services may vary slightly depending on the adapter you are using.

- Specify tracing information if desired. TIBCO Designer allows you to specify simple tracing to a file or standard out using the configuration panel directly. You can also specify advanced tracing, such as tracing to a network sink.

If your adapter uses advanced features, such as Advisory activities, you can configure them using the Advanced folder of the adapter.

For more information about adapter configuration, see the documentation for that adapter, available via **Help > Help For** from TIBCO Designer.

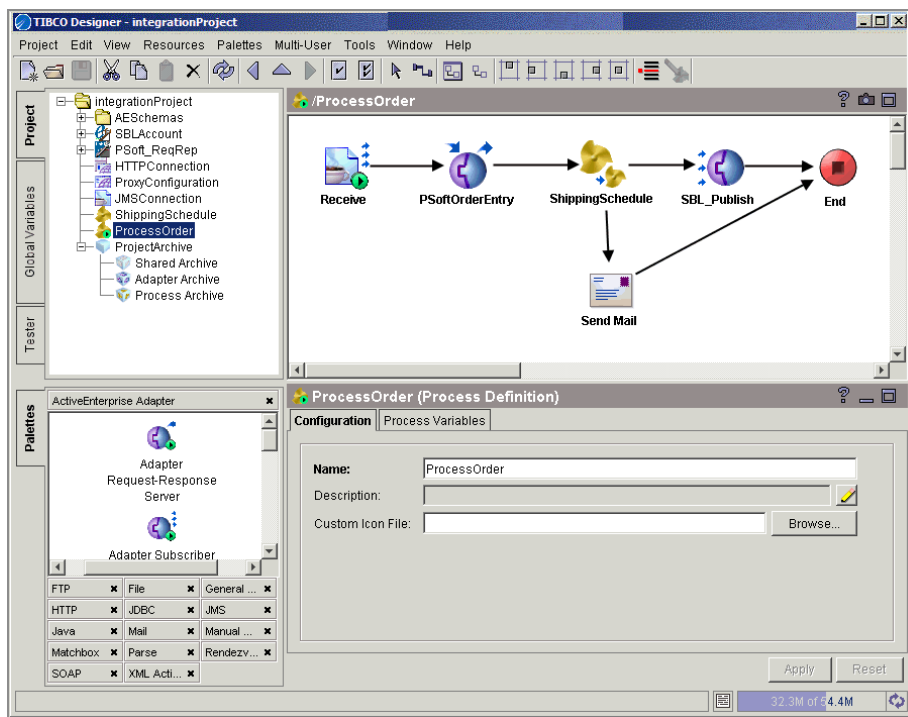
Step 4: Accessing the Adapter Service From the Process

After you have configured the adapter service, you can access it from the process definition. Creating a process is explained in [Phase 4: Business Process Design](#). Once a process is part of your project, you can access adapters as follows:

Procedure

1. Select the process definition, then open the ActiveEnterprise Adapter palette.

Figure 15: Adding activities that access adapter services



2. Drag the activity that accesses the service you need into the design panel. You have the following choices:
 - a. 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. In the example above, a Publish to Adapter activity interacts with a Siebel subscriber.
 - b. Adapter Subscriber—Subscribes to incoming data published by the adapter.
 - c. Invoke an Adapter Request-Response Service—Communicates (as a client) with an adapter request-response service.
 - d. Adapter Request-Response Server—Starts a process based on the receipt of a request from an adapter.
 - e. Respond to Adapter Request—Sends a response to an adapter for a previously received request.
 - f. Wait for Adapter Message—Waits for the receipt of a message from the

publication service of the specified adapter.

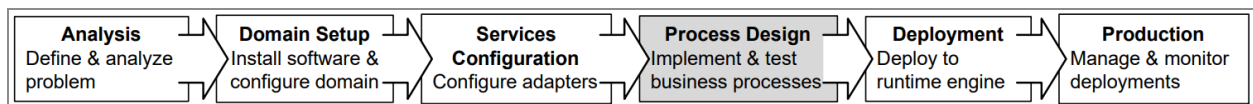
- g. Wait for Adapter Request—Waits for the receipt of a request from a request-response invocation service.
3. Specify the adapter information in the configuration panel.
 4. Save your project.

Phase 4: Business Process Design

This section discusses business process design. Using the TIBCO Designer GUI, you create your business process using predefined activities and add conditions and mapping as appropriate.

Business process design is discussed in more detail in the *TIBCO ActiveMatrix BusinessWorks™ Process Design*.

You can also display information about each resource by choosing its **What is This** right-button menu command in TIBCO Designer.



Introduction

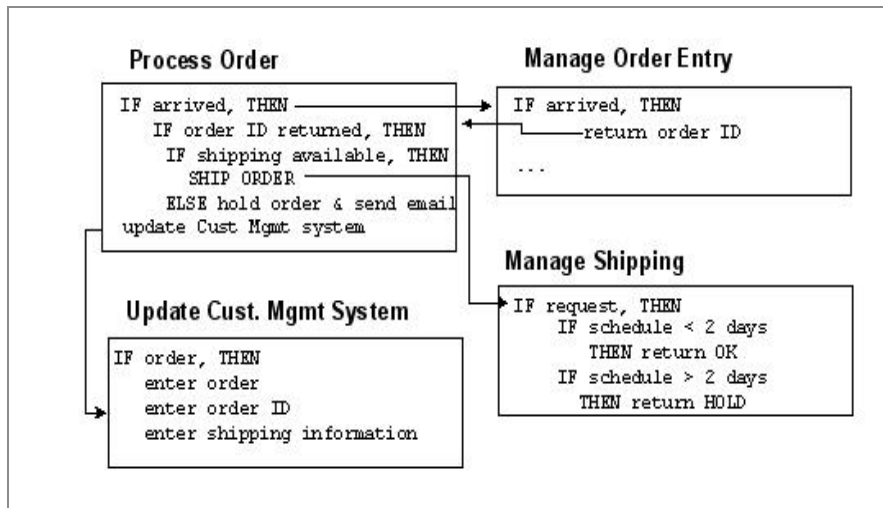
In an integrated enterprise, business processes manage the flow of information between different data sources and destinations. The business processes receive information, process it if needed, and hand it off as appropriate.

Business Processes

In many companies, the business rules that tie enterprise applications together are handled by custom-written code or even by manual processes. ActiveMatrix BusinessWorks allows you to establish this data flow using activities available in the TIBCO Designer process design palettes. You design the process using predefined activities and can execute your process without writing a lot of custom code.

The following diagram illustrates a business process flow that describes the business rules between the various systems in an enterprise.

Figure 16: Business process flow



Process Design Steps

Process design consists of a number of steps discussed in this section.

Note: The steps discussed in this section do not necessarily have to be performed in this order. For example, you could define shared resources as needed or add transitions each time you add an activity

Step 1: Define Shared Resources

Shared resources allow activities to share information. For example, you can define a JDBC Connection resource, then use it in any of the JDBC activities in your business process.

Tip: You may also choose to define the process and create shared resources as needed.

Shared resources are available in some palettes and include Rendezvous Transport, JDBC Connection, JMS Connection, and so on.

To define a shared resource, follow these steps:

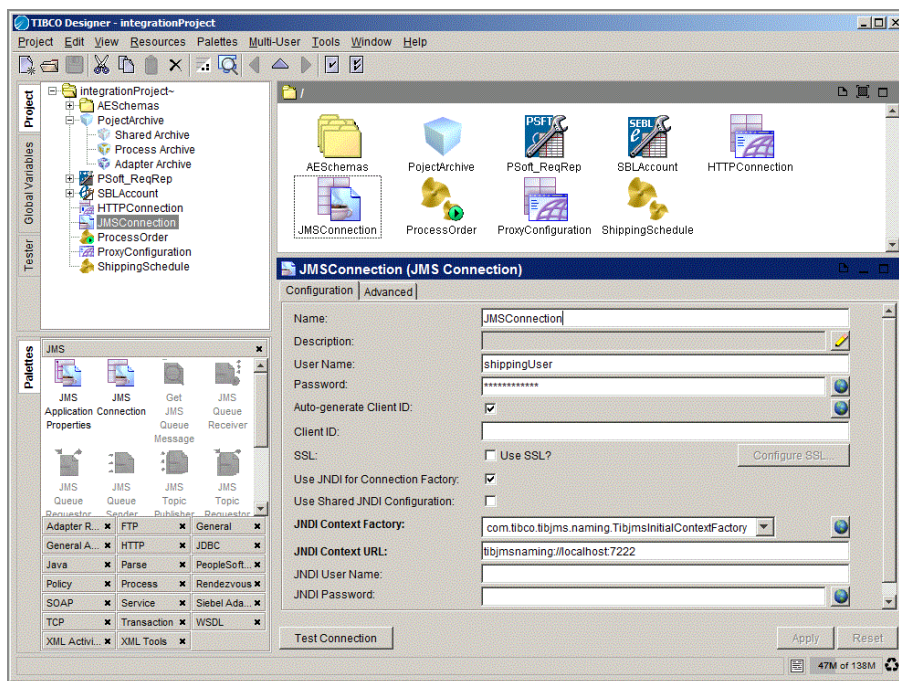
Procedure

1. In the palette panel, select the palette that contains the shared configuration resource. For example, the JMS palette contains two shared configuration resources, JMS Connection and JMS Application Properties.
2. Drag and drop the icon for the resource you need from the palette panel to the design panel.
3. Name the resource and specify its configuration information in the configuration panel, then click **Apply**.

You can now use the shared resource in any activity that requires it.

For example, the JMS Queue Receiver activity the example uses to connect to the application server requires a JMS Connection shared resource.

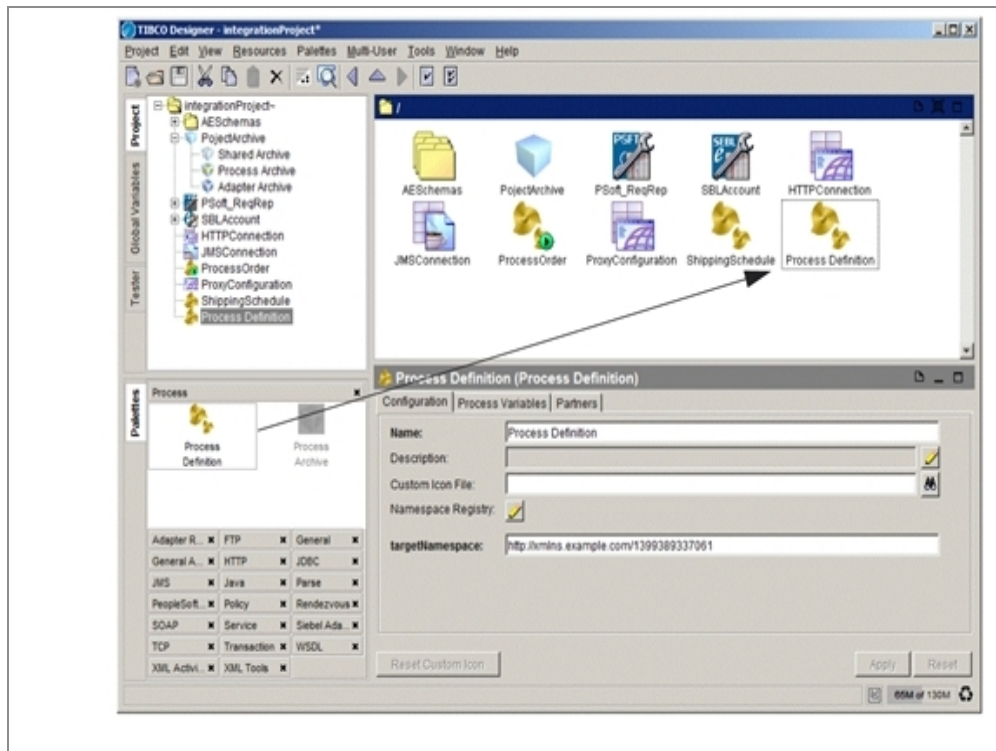
Figure 17: Shared resources in your project



Step 2: Create Process Definitions

In TIBCO Designer, you create process definitions by dragging Process Definition resources from the palette panel to the design panel. You name each process definition and give the process a description, then add activities to the process.

Figure 18: Adding process definitions to your project



For simple business processes, activities are added in sequence, then transitions are added as appropriate.

For complex business processes, it makes sense to design a main process and several subprocesses. Using subprocesses makes your process easier to understand and debug. Subprocesses also potentially allow reuse of business process components.

Our example program includes several different processes:

- In the main process, which is called ProcessOrder, a JMS Queue Receiver waits for input. When it arrives, the main process first interacts with the Order Management system, then calls the ShippingSchedule process:
- The ShippingSchedule process retrieves information about the shipping schedule from the Shipping web site and adds shipping information to the order. It sends an email to the customer if shipping is delayed, otherwise enters the order into the Siebel system right away.

Step 3: Add a Process Starter

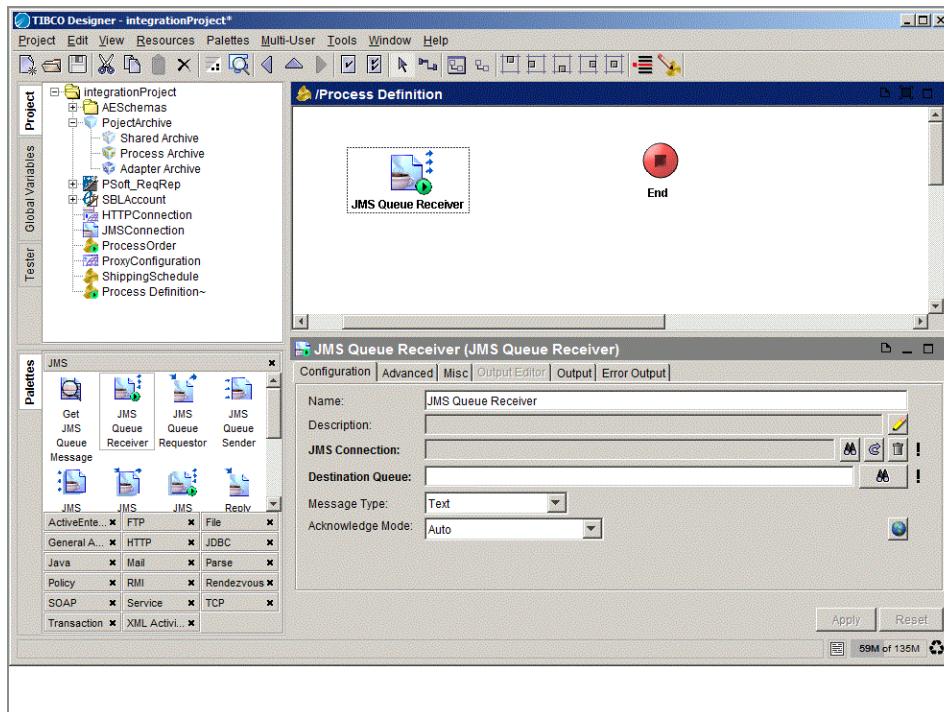
A process starter waits for input from an external process and creates a process instance each time the input arrives. For example, the process could be waiting for a document that arrives from an application server using JMS. A process starter could also be polling a directory and start whenever a file is added.

To add a process starter, follow these steps:

Procedure

1. Choose the process to which you want to add the process starter
2. Select the palette for the process starter in the palette panel.
In our example, the JMS Queue Receiver is the process starter.
3. Drag the process starter into the design panel.
The process starter activity replaces the default Start activity.
4. Specify configuration information, then click **Apply**.

Figure 19: Adding a process starter



Note that the default Start activity that is included with each process you instantiate is not a process starter. A Start activity must be called explicitly from another process.

Step 4: Add Activities

Activities are the individual units of work within a process definition. Activities are generally operations that interface to external systems, but activities can also perform internal processing.

When you select a process definition, activities become available on the various TIBCO Designer palettes. Each palette has a set of activities that can be performed for that palette. For example, the ActiveEnterprise Adapter palette has activities that can publish messages to a specified adapter or invoke an operation by way of an adapter. The JMS palette includes activities such as JMS Queue Sender and JMS Queue Receiver.

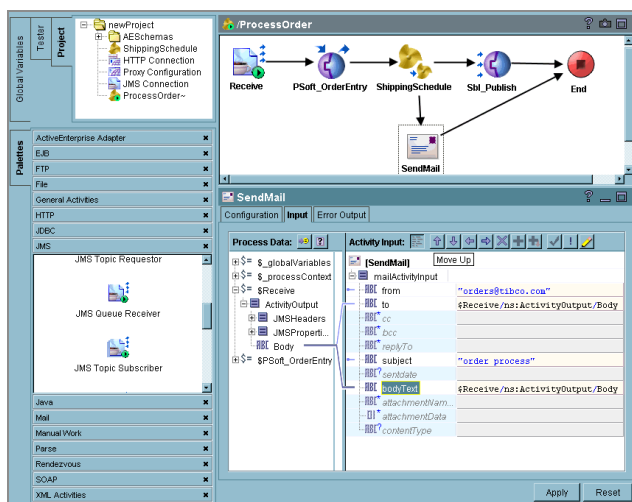
A general-purpose Java Code activity allows you to write and execute standard Java code to perform custom processing within your process definition.

To add an activity to the process definition in TIBCO Designer, follow these steps:

Procedure

1. Select the appropriate palette. If the palette is not visible, make sure you have selected the appropriate parent resource and make sure the palette has not been closed.
2. Drag the activity into the design panel.
3. Select the activity and specify configuration information about it, for example, the originator and address for a Send Mail activity.

Figure 20: Adding activities



Step 5: Optionally, Add Manual Work Activities

The activities in the Manual Work palette are useful for automated business processes that have a few steps that require user interaction. This could include handling of unexpected situations or other activities such as credit approval or handling of customers from abroad if the company usually deals with local customers. The functionality is implemented by two shared configuration resources and four activities:

Shared Configuration Resources

- **Workflow Schema**—The data associated with a ManualWork task. This is the data a user needs to complete the task. The user can change any of the data associated with a task.

A workflow schema supports string, int, date, and document data types.

- **Workflow Connection**— The connection to the TIBCO InConcert workflow server. This server is used to track and manage manual tasks.

Manual Work Activities

- **Assign Work**—Creates a new task (with associated data) and assigns it to the specified pool of users. A user then acquires the task, views the data supplied by the activity, and completes the work.

The process definition can either wait for the user to complete the work or it can continue processing and later query for the status of the task.

- **Download Document**—Manual work schemas can contain elements to hold documents. An example of a document is a loan application that must be attached to a credit request. The document may be any type, for example, Microsoft Word, PDF, and so on.

Documents can be uploaded or downloaded to a manual work task by way of the web interface ActiveMatrix BusinessWorks provides for managing manual work tasks. This activity allows you to download a document from an existing manual work task into a process variable.

- **Get Work Status**—Retrieves the current status of a task that was previously created with the Assign Work activity. Normally you use this activity to determine if the task has been completed or if there are any errors.
- **Modify Work**—This activity allows administrator users to change the status of an

existing task created with the Assign Work activity to one of the following:

- a. Update — modifies the data associated with an uncompleted task
 - b. Complete — completes the task
 - c. Reassign — reassigns the task to a pool of users
- Wait for Completion—Waits for the completion of the task for the specified period. This is useful if the timeout for the Assign Work activity has passed and you wish to wait for an additional amount of time.

Step 6: Create Transitions Between Activities

You use transitions to connect activities to form a process flow. A transition is represented by an arrow between two activities. Arrows are unidirectional, and you cannot draw a transition to a previously executed activity. Control flow in a process definition must proceed sequentially beginning with the starting activity and ending with the End activity.

Conditions

A transition can optionally specify a condition. The condition determines whether a transition is taken when an activity completes processing. After an activity completes, all transitions whose conditions are met are taken. You can have transitions from one activity to many other activities.

For example, if the shipping schedule indicates a delay in shipping the order, you want to notify the customer and enter the information into the customer service system. If it does not, you just enter the information into the customer service system.

Adding Transitions

To add transitions, click the **transition tool** on the toolbar to draw transitions between activities. You use XPath syntax to define conditions.

If your process definition includes cyclical subprocesses (loops), you can group them using the GUI.

Step 7: Perform Mapping and Transformation for Each Activity

As data flow through your business process, different activities require different components of the data. On the Input tab of each activity, TIBCO Designer displays the available process data and the activity's input represented as schema trees.

- The process data is the list of available data items within the process at the point where the activity is located (an activity has access to all output data from any activity that is executed before it in the process definition).
- The activity input is the list of input values that are required or optional for the activity.

For each activity, you map the process data to the activity input.

Each item in the activity input schema has an expression field for specifying the contents of the item. You can conditionally map the process data to the input using XPath expressions. You do not need detailed knowledge of XPath to create simple expressions. For the most part, you can drag and drop items from the process data schema to the activity input schema, and the correct XPath expression appears automatically.

When you specify the input schema for an activity, the specification is represented internally as Extensible Stylesheet Language Transformation (XSLT) code. Normally, you do not need to examine the XSLT code generated by the mappings. However, if you are familiar with XSLT and you wish to see the actual code, you can right-click on any node in the input schema and choose Copy from the popup menu. Then open a blank text document and choose Paste. The XSLT is displayed in your text document.

For example, when the ShippingSchedule process fails, an email is sent upon return to the main process. The email address information could be mapped from the original order to the mail activity's input fields.

Step 8: Optionally, Group Activities As Needed

Groups are used to specify related sets of activities. Grouping allows you to create loops. The main uses of groups are the following:

- Create a set of activities with only one condition for the group. This allows you to

catch any error that occurs inside of the group, instead of trying to individually catch errors on each activity. This type of group is similar to a `try...catch` block in Java.

- Create sets of activities that are to be repeated. You can repeat the activities once for each item in a list, until a condition is true, or if an error occurs.
- Create sets of activities that participate in a transaction. Activities within the group that can take part in a transaction are processed together or rolled back, depending upon whether the transaction commits or rolls back.

Step 9: Test the Process

Once the process definition is complete, you can perform preliminary testing from TIBCO Designer. In test mode, a ActiveMatrix BusinessWorks engine is started to perform the processing specified in the process definition.

The ActiveMatrix BusinessWorks testing environment allows you to step through your process models and find sources of errors. Entering the testing environment starts a ActiveMatrix BusinessWorks engine. The engine starts process instances based on the process definitions stored in your project. The testing environment displays the running process definitions and highlights the currently executing activity.

Testing a process definition typically involves these steps:

Procedure

1. Select the process definition you wish to test in the project panel.
2. Click the Start Test Mode icon on the toolbar. This starts a ActiveMatrix BusinessWorks engine that will execute the processes.

Once the engine is started, the test mode icon displays on the toolbar to let you know you are in test mode. Once you are in test mode, you cannot change your process definition. You must exit test mode to make changes.
3. Set breakpoints in the process definition at points where you wish to stop a running process and examine its state.
4. If necessary, supply input data to the process starter.
5. Click the Start/Resume Testing icon to create a process instance from the displayed process definition.
6. Start a process by creating an event that the process starter is expecting. For

example, if the process starter is listening for a TIBCO Rendezvous message, publish a message on the expected subject.

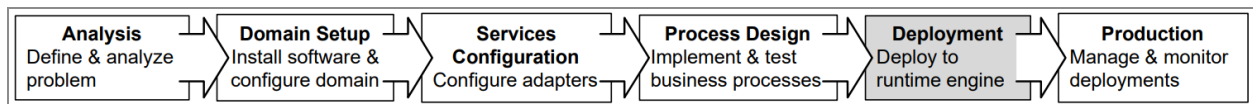
7. If you started multiple process instances, select the desired process instance from the list of processes in the toolbar. Examine the process data by selecting any of the activities in the process. The activity's current data is displayed on the Input and Output tabs.
8. Use the toolbar buttons (Pause Testing, Step to Next Activity, Step Into SubProcesses, Stop Testing, Start/Resume Testing) to either continue through the process instance or to stop the current process instance.

When your process definition operates as expected, you can go on to deployment, the next step in your integration project, which is discussed in [Phase 5: Deployment](#) and, in more detail, in the *TIBCO ActiveMatrix BusinessWorks™ Administration*.

Phase 5: Deployment

For the success of your integration project, ease of deployment is at least as important as ease of design. Because ActiveMatrix BusinessWorks uses the TIBCO administration domain and allows you to perform deployment configuration from the TIBCO Designer GUI, deployment is a relatively simple task.

This section gives an overview of the deployment of a ActiveMatrix BusinessWorks integration project. For more information, see *TIBCO ActiveMatrix BusinessWorks™ Administration*.



Introduction

When you have completed and tested the first prototype of your integration project in the development environment, you are ready to deploy it to a testing environment.

In a traditional business integration project, domain configuration is a labor-intensive process that is likely to require multiple iterations before all components are in place. The TIBCO Administrator must tweak the configuration files for different components on different machines. To optimize the configuration, TIBCO Administrator must manually keep a record of the different configurations that were tested.

ActiveMatrix BusinessWorks, in contrast, allows you to use the TIBCO Administrator GUI to create a deployment configuration and then deploy the project.

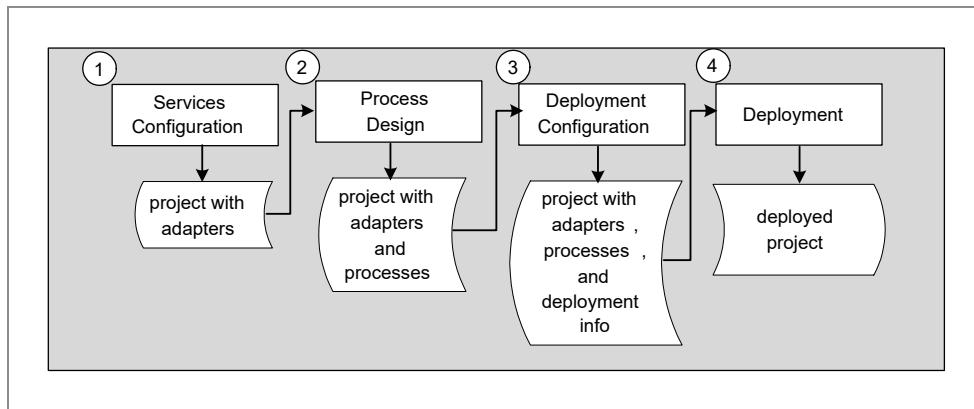
- You use the TIBCO Designer GUI to create an Enterprise Archive file (EAR file) containing the adapter configurations and process definitions you wish to deploy.
- TIBCO Administrator uses the EAR file when creating the deployment configuration.
- You can deploy the project in TIBCO Administrator and the necessary scripts and other information is sent to the machines within the deployment.

ActiveMatrix BusinessWorks Project Phases

When process design is complete and all testing from the TIBCO Designer GUI results in success, the project is ready for deployment.

The illustration below shows how a project moves through the development and deployment phases.

Figure 21: Project development phases



Procedure

1. Using TIBCO Designer, the developer configures adapter services and saves the project with configured adapters. Different adapters could potentially be configured by different developers and included in one project.
2. Using TIBCO Designer, the developer configures activities for the business process(es) and saves the project, which now includes process definition(s).
3. The built-in test mode is used for preliminary testing and debugging of the business process(es).
4. Using TIBCO Designer, the developer prepares the Enterprise Archive File (EAR file) by creating Enterprise Archives containing the desired adapter configurations and process definitions.

At this stage, the developer moves from the TIBCO Designer development GUI to the TIBCO Administrator run-time GUI and performs these tasks:

Procedure

1. From the TIBCO Administrator GUI, the deployment configuration is created and

deployed. The TIBCO Administration Server sends all necessary information to the individual machines. All components now become visible in the TIBCO Administrator GUI (but are not started).

2. From the TIBCO Administrator GUI, the developer starts each component (adapter and process engine). As a result, all process starters are waiting for the events that cause them to create process instances.
3. Each time an event arrives that triggers a process starter, the ActiveMatrix BusinessWorks engine creates a process instance, which uses the associated process definition to process the incoming data.
4. Using the TIBCO Administrator GUI, the developer can monitor the different processes running on the different machines.

If appropriate, the project can be modified and saved from TIBCO Designer. In that case, you must create a new EAR file and place the new file into the deployment configuration. You must then stop and restart the desired deployment for the changes to take effect.

For more information about creating EAR files and deployment configurations, see *TIBCO ActiveMatrix BusinessWorks™ Administration*.

OpenTelemetry

OpenTelemetry is an open source, vendor neutral standard for distributed systems that are used to track the current state of the job. OpenTelemetry is a set of APIs, SDKs, tooling, and integrations designed to create and manage telemetry data such as traces and metrics.

For more information, see [OpenTelemetry Documentation](#).

OpenTelemetry Tracing

The OpenTelemetry tracing related properties can be configured in any of the following files:

- `bwengine.tra`
- `designer.tra`
- `<application_name>.tra`

Configuration file option where changes can be applied

For the already deployed applications:

1. Add the necessary properties in the application TRA file located at <TIBCO_HOME>\tra\domain\<Domain_name>\application\<application_name>\<application_name>.tra.
 2. After making the changes, restart the application's service instances.
- No Redeployment is required.

To apply the configuration across all the applications deployed and created from a specific TIBCO_HOME, add the properties to bwengine.tra. This configuration is applied for newly created applications and its configuration file.

For the default auto-configuration mode

Following are the 3 properties needs to be added in the .tra file

1. To enable OpenTelemetry, set the following property:
`java.property.bw.engine.opentelemetry.enable=true`
2. There are two types of protocol supported for the OpenTelemetry tracing which are GRPC and HTTP.

Use the HTTP protocol to view ActiveMatrix BusinessWorks 5.x activities traces through the TIBCO Control Plane web UI.

Use one of the following properties to set the required protocol:

```
java.property.otel.exporter.otlp.traces.protocol=grpc or
java.property.otel.exporter.otlp.traces.protocol=http/protobuf
```

3. To enable tracing for the ActiveMatrix BusinessWorks 5.x applications from a remote machine, use the following property:

```
java.property.otel.exporter.otlp.traces.endpoint=<< Endpoint URL >>
```

On a bare metal on-premises setup with Jaeger UI through ActiveMatrix BusinessWorks 5.x engine use the properties with the sample value.

For HTTP:

```
java.property.otel.exporter.otlp.traces.endpoint=http://localhost:4318/v1/traces
```

For GRPC:

```
java.property.otel.exporter.otlp.traces.endpoint=http://localhost:4317
```

i Note: You can configure the following property to expose the traces and to view these traces under the TIBCO Control Plane UI:

```
java.property.otel.exporter.otlp.traces.endpoint=http://<hostname>/tibco/agent/o11y/dataplane-ID/traces
```

For the non- auto-configuration mode

If you want to disable the default auto-configuration mode, add the following property in the .tra file.

```
java.property.bw.engine.opentelemetry.disableAutoConfiguration=true
```

After disabling the auto-configured mode, with the manual mode add the following properties in the .tra file:

- `java.property.bw.engine.opentelemetry.enable=true`
- `java.property.bw.engine.opentelemetry.span.exporter=OTLP-GRPC`
- `java.property.bw.engine.opentelemetry.span.exporter.endpoint=http://localhost:4317`

OR

- `java.property.bw.engine.opentelemetry.span.exporter=OTLP-HTTP`
- `java.property.bw.engine.opentelemetry.span.exporter.endpoint=http://localhost:4318/v1/traces`

i Note: You can configure the following property to expose the traces and to view these traces under the TIBCO Control Plane UI:

```
Java.property.bw.engine.opentelemetry.span.exporter.endpoint=http://<BareMetal_DP_MachineName>/tibco/agent/o11y/dataplane-ID/traces
```

OpenTelemetry Metrics

This topic shows the metrics supported in ActiveMatrix BusinessWorks.

Auto Configuration mode

It is the default mode in the ActiveMatrix BusinessWorks 5.x engine.

Add the following properties in the configuration file:

- `java.property.bw.engine.opentelemetry.metrics.enable=true`
- `java.property.otel.exporter.otlp.metrics.protocol`

There are two types of protocol supported for the OpenTelemetry metrics - GRPC and HTTP. By default, GRPC is enabled. You can use any of the two protocols in the following ways:

- `java.property.otel.exporter.otlp.metrics.protocol=grpc`
- `java.property.otel.exporter.otlp.metrics.protocol=http/protobuf`
- `java.property.otel.exporter.otlp.metrics.endpoint`

The endpoint is set according to the protocol. Default values are:

For GRPC:

```
java.property.otel.exporter.otlp.metrics.endpoint=http://localhost:4317
```

For HTTP:

```
java.property.otel.exporter.otlp.metrics.endpoint=http://localhost:4318/v1/metrics
```

To configure the Open Telemetry metric export interval, use the following property :

```
java.property.otel.metric.export.interval=<time_interval_in_
milliseconds>=<time_interval_in_milliseconds>
```

Manual Configuration mode

If you want to disable the default auto-configured mode, add the following property in the configuration file:

```
java.property.bw.engine.opentelemetry.metrics.disableAutoConfiguration=true
```

There is no change for enabling the metrics in manual mode. Same property is used.

```
java.property.bw.engine.opentelemetry.metrics.enable=true
```

After disabling the auto configured mode to the manual mode, change the following other required properties:

For setting the protocol, use the following property with one of the values:

- `java.property.bw.engine.opentelemetry.metrics.exporter.protocol=grpc`

- `java.property.bw.engine.opentelemetry.metrics.exporter.protocol=http/protobuf`

By default, GRPC is enabled.

For setting the endpoint, use the following property:

```
java.property.bw.engine.opentelemetry.metrics.exporter.endpoint
```

Default values are the same as that of auto configuration.

Default values are:

For GRPC:

```
java.property.bw.engine.opentelemetry.metrics.exporter.endpoint=http://localhost:4317
```

For HTTP:

```
java.property.bw.engine.opentelemetry.metrics.exporter.endpoint=http://localhost:4318/v1/metrics
```

Configuration file option where changes can be applied

For already deployed applications, the required properties should be added in the TRA file located in the directory: `<TIBCO_HOME>/tra/domain/<Domain_Name>/application/<Application_Name>/`

Once the changes are made, restart the service instance. Redeployment is not required.

To apply the configuration to all the applications deployed and created from a specific TIBCO_HOME, add the properties to the `bwengine.tra` file.

This will be applied for newly deployed applications.

List of Metrics

The ActiveMatrix BusinessWorks 5.x engine currently exports the following metrics:

- `USED_CPU_PERCENT` - CPU usage percentage by service instance.
- `USED_MEMORY_PERCENT` - Memory usage percentage by service instance.
- `COMPLETED_JOB_COUNT` - Success execution count by service instance.
- `FAULTED_JOB_COUNT` - Failure execution count by service instance.
- `SERVICE_INSTANCE_STATE` - Service instance details and its running state. A service

instance can send other service instances' details and their state.

Possible values seen for SERVICE INSTANCE STATE are:

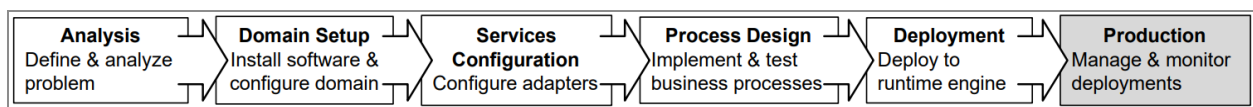
- EMPTY(0, "-"),
- STOPPED(1, "Stopped"),
- SHUTTING_DOWN(2, "Shutting down"),
- UNKNOWN(3, "Unknown"),
- ERROR(4, "Error"),
- STANDBY(5, "Standby"),
- STARTING_UP(6, "Starting up"),
- RUNNING(7, "Running");
- USED_MEMORY - Used memory by service instance in bytes.
- FREE_MEMORY - Free memory in bytes.

The first four of these metrics are used in the ActiveMatrix BusinessWorks 5.x UI.

Phase 6: Production

During the production phase, you monitor and manage ActiveMatrix BusinessWorks deployments. The TIBCO Administration Server and the TIBCO Administrator GUI together support your deployed ActiveMatrix BusinessWorks products at runtime. This section gives an overview of available functionality.

For more information, see TIBCO Administrator documentation.



Introduction

After the integration development team has configured and deployed the integration project, you can use the TIBCO Administrator GUI for monitoring and management.

This section gives an overview of the architecture and of the user management, TIBCO administration domain monitoring and management, and deployment management options. For more information, see *TIBCO Administrator User's Guide*.

Architecture

TIBCO Administrator consists of the TIBCO Administration Server and the TIBCO AdministratorGUI. The TIBCO Administration Server includes an application server component for HTTP communications and a repository server component for data store management. All three server components run as a single process.

The TIBCO Administration Server interacts with each machine in the TIBCO administration domain by way of the TRA running on that machine.

- The TIBCO Administration Server receives information about CPU and memory usage, alerts, and the process instances and components running on each machine and

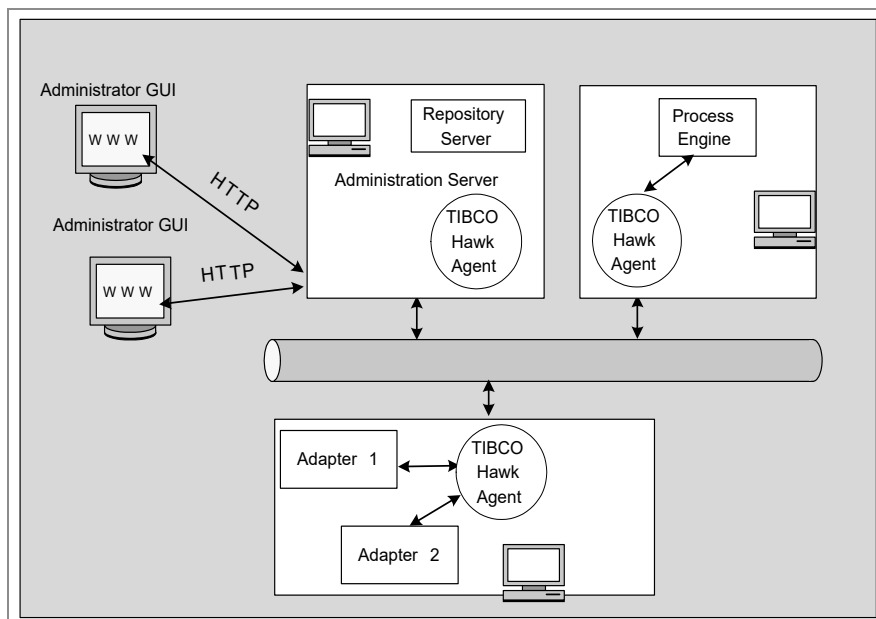
makes them available via the TIBCO Administrator GUI.

- The TIBCO Administrator GUI allows users to start and shut down components. Start and shutdown commands are sent from the TIBCO Administration Server to the appropriate TRA, which in turn starts or stops the process.

The following illustration shows the TIBCO administration domain with 3 machines in the domain.

- The first machine runs the TIBCO Administration Server, which contains an embedded Repository Server for managing data stores. Machines outside the TIBCO administration domain can view the TIBCO Administrator GUI using a web browser. Each user sees only the components for which s/he is authorized.
- The second machine runs two adapters. The adapters are started via the TIBCO Administrator GUI. The information entered in the GUI is sent by the TIBCO Administration Server to the TRA on the machine running the adapter. The TRA starts and stops the adapter and also sends information about component and machine status to the TIBCO Administration Server for access via the GUI.
- The third machine runs a process engine. Just like the adapters, the process engine is managed by the TIBCO Administration Server via the TRA.

Figure 22: Communication inside a TIBCO administration domain



Monitoring and Management Options

At runtime, TIBCO Administrator allows you to remotely access all deployments in your TIBCO administration domain. Depending on your security privileges, you may be able to perform one or more of the following tasks.

- **User Management**—Add users and passwords to the TIBCO administration domain for authentication, then give each user view or execute privileges for TIBCO Administrator GUI elements or project repositories (data stores)
- **Domain Monitoring and Management**—View the status of machines and components running on machines in the domain. Start and stop components as needed.
- **Deployment monitoring**—Monitor the status of each deployment component (process engine and adapter) and process instance and the status of the machines executing them. View trace files and throughput.
- **Deployment management**—View all running components and stop and restart them as needed.

These aspects of monitoring and management are supported by TIBCO Administrator, which includes the TIBCO Administration Server and the TIBCO Administrator GUI.

Glossary

A

ActiveEnterprise

One of the three product families from TIBCO Software Inc. The other two are ActivePortal™ and ActiveExchange™ (see www.tibco.com for details).

Activity

An activity is a specific task in a business process definition. Examples are sending e-mail, writing a file, sending information to an SAP adapter, querying a database. In TIBCO Designer, each activity is represented by a resource and can be added to the process definition from its palette. See also Process Definition.

Adapter

Adapters make it possible to communicate and update business information that originates from diverse sources and resides on diverse host systems within an organization. ActiveMatrix BusinessWorks adapters interact with enterprise applications such as PeopleSoft, SAP R/3, OracleApps and Siebel, as well as databases or files.

TIBCO Administration Domain

Consists of a set of machines on which TIBCO software components are deployed, a set of deployed projects, and a single database for authentication and authorization.

When you install ActiveMatrix BusinessWorks, you specify a TIBCO Administration Domain. The domain initially contains one or more machines. Later, the domain contains the services and process engines running on those machines. Multiple projects can be in the same domain. You can monitor the components of all deployed projects in a domain from the TIBCO Administrator console.

C

Condition

A condition can be used to control the flow of activities in a process diagram. Conditions are specified on transitions to determine whether to take the transition to

the next activity or not. Condition types include: always, XPath which allows you to specify a custom condition using an XPath expression, and otherwise.

Configuration Panel

In TIBCO Designer, the configuration panel allows you to fill in values for the fields in the objects of your project, and to perform mapping.

CRM

Customer Relationship Management system. One of the applications in your enterprise.

D

Deployment

During the deployment phase of your integration project, you assign the various project components (process and services) to the physical locations where they will run for test or production purpose. For example, after production deployment, the adapter service instances, and other process instances will be distributed across the production platform.

Design panel

In TIBCO Designer, you drag resources from the palette panel into the design panel to create instances of that resource for your business process definition.

DTD

Document Type Definition. A non-XML schema file that contains a formal description of the vocabulary and structure of the elements in an associated XML file. DTDs serve the same function as XML schema documents (XSDs). A DTD may also provide some content information. The DTD for an XML document is the combination of the internal and external subsets described by the document type declaration.

E

Endpoint

An endpoint exposes the service to other applications and services. An endpoint is analogous to a port in a WSDL file. Multiple endpoints for each service can be created so that operations within the implementation of the service can be invoked in more than one way.

G

Activity Group

A grouping of activities in a process definition. Some groups specify process control logic.

I

Interface

An interface describes the operations available within a service. The interface is analogous to a portType in a WSDL file. Each interface can contain multiple operations.

M

Machine

A computer on which TIBCO Software components are installed. Each machine may belong to only one TIBCO Administration Domain.

P

Palette

A TIBCO Designer palette is a collection of resources that you can use to populate your project. For example, use an adapter palette to add an adapter and a publisher to a project.

Partner

Partners are other services that your process or service can invoke. Partners are defined by a name and a WSDL portType that describes the operations that can be invoked.

Partner Link Configuration

Partner Link Configuration resources associate partners with endpoints by specifying the actual bindings to endpoints.

Process definition

Specifies the business process flow using activities. A process definition is just a special type of activity; you can therefore use a process definition as an activity.

Process instance

Running instance of a process definition

Process starter

Starts a process based on an external event, e.g. arrival of a file or a message from an adapter.

Project

A ActiveMatrix BusinessWorks project is a collection of all the components of your EAI solution. This includes components like business processes, adapter services, machines, engine instance(s). You create and save projects using TIBCO Designer and later deploy the project.

Project tree panel

One of the panels in TIBCO Designer. The project tree shows a hierarchy of all objects (adapters, activities, etc.) in your project.

R**Resource**

In TIBCO Designer, resources are the objects you can drag and drop into the design panel.

S**Service**

A ActiveMatrix BusinessWorks service retrieves, generates, and processes business data in a decoupled yet reliable manner. The business process receives data from a service and sends data to a service. Associated with a service is a service interface, transport, data syntax, and data schema.

SOAP

Simple Object Access Protocol. SOAP is a lightweight protocol for the exchange of information in a decentralized, distributed environment. It is an XML based protocol that consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing remote procedure calls and responses. (See <http://www.w3.org/TR/SOAP/>)

T**Tab**

A related set of entities and functions that are visible when you select one tab in the TIBCO Administrator GUI. As components are installed, additional tabs become available.

TIBCO Administrator

TIBCO Administrator is a browser-based interface for monitoring and managing TIBCO deployments, their security, and TIBCO administration domains. TIBCO Administrator organizes the information using tabs.

TIBCO Designer

A GUI tool that allows you design your integration project. You use TIBCO Designer for adapter configuration, process design, and deployment.

Transition

Transitions indicate the flow of processing. In TIBCO Designer, a transition is represented by an arrow between activities. Each activity in a process definition must have a transition to it, or the activity is not executed when the process executes.

W

WSDL Web Services Definition Language.

WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). (See <http://www.w3.org/TR/wsdl>)

X

XPath

XPath is a scripting language developed by the World Wide Web Consortium (W3C) for addressing parts of XML documents. It provides basic manipulation functions for strings, numbers and Booleans. TIBCO Designer uses XPath as the language for defining conditions and transformations. A complete description of XPath is available at <http://www.w3.org/TR/xpath>.

XSD

XML Schema Definition. An XSD file defines the structure and elements in a related XML file. The suffix of an XSD document is .xsd.

XSLT

XML Stylesheet Language Transformation. XSL Transformations (XSLT) is a standard way to describe how to transform (change) the structure of an XML (Extensible Markup Language) document into an XML document with a different structure. XSLT is a recommendation of the World Wide Web Consortium

TIBCO Product Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The following documentation for this product is available on the [TIBCO ActiveMatrix BusinessWorks™ Product Documentation](#) page:

- *TIBCO ActiveMatrix BusinessWorks™ Release Notes*
- *TIBCO ActiveMatrix BusinessWorks™ Administration*
- *TIBCO ActiveMatrix BusinessWorks™ Concepts*
- *TIBCO ActiveMatrix BusinessWorks™ Error Codes*
- *TIBCO ActiveMatrix BusinessWorks™ Getting Started*
- *TIBCO ActiveMatrix BusinessWorks™ Installation*
- *TIBCO ActiveMatrix BusinessWorks™ Palette Reference*
- *TIBCO ActiveMatrix BusinessWorks™ Process Design*

To directly access documentation for this product, double-click the following file:

`TIBCO_HOME/release_notes/TIB_<productID>_version_docinfo.html`

where `TIBCO_HOME` is the top-level directory in which TIBCO products are installed. On Windows, the default `TIBCO_HOME` is `C:\tibco`. On UNIX systems, the default `TIBCO_HOME` is `/opt/tibco`.

Other TIBCO Product Documentation

When working with ActiveMatrix BusinessWorks, you may find it useful to read the documentation of the following TIBCO products:

- TIBCO Designer™: *TIBCO Designer is an easy to use graphical user interface for design-time configuration of TIBCO applications. TIBCO Designer includes online help for each palette.*
- TIBCO Runtime Agent™: *TRA supplies a number of TIBCO and third-party libraries used by ActiveMatrix BusinessWorks.*
- TIBCO Administrator™: *TIBCO Administrator is the monitoring and managing interface for new-generation TIBCO products such as ActiveMatrix BusinessWorks.*
- TIBCO Rendezvous® : *TIBCO Rendezvous software uses messages to enable distributed application programs to communicate across a wide variety of hardware platforms and programming languages.*

How to Access Related Third-Party Documentation

When working with ActiveMatrix BusinessWorks, you may find it useful to read the documentation of the following third-party products:

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

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

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

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

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, and are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

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

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

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

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

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

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>.

Copyright © 2001-2025. Cloud Software Group, Inc. All Rights Reserved.