

# **TIBCO ActiveMatrix<sup>®</sup> Adapter for PeopleSoft**

## **Examples**

*Software Release 6.0  
December 2009*

## Important Information

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

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

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

TIB, TIBCO, TIBCO ActiveMatrix Adapter, Predictive Business, Information Bus, The Power of Now, TIBCO ActiveMatrix BusinessWorks, TIBCO Rendezvous, TIBCO Administrator, TIBCO Designer, TIBCO Runtime Agent, TIBCO Hawk, TIBCO Enterprise Message Service, TIBCO Designer Add-in for TIBCO Business Studio, TIBCO ActiveMatrix Service Grid, TIBCO ActiveMatrix Service Bus, TIBCO ActiveMatrix BusinessWorks Service Engine, and TIBCO Business Studio are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

EJB, Java EE, J2EE, and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

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

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. SEE THE README.TXT FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

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

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

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

Copyright © 1999-2009 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

# Contents

<b>Figures</b> .....	<b>vii</b>
<b>Tables</b> .....	<b>ix</b>
<b>Preface</b> .....	<b>xi</b>
Related Documentation .....	xii
TIBCO ActiveMatrix Adapter for PeopleSoft Documentation .....	xii
Other TIBCO Product Documentation .....	xii
Third-Party Documentation .....	xiii
Typographical Conventions .....	xiv
Terminology and Acronyms .....	xvi
How to Contact TIBCO Support .....	xvii
<b>Chapter 1 Introduction</b> .....	<b>1</b>
Overview .....	2
Prerequisites .....	5
Using TIBCO ActiveMatrix BusinessWorks .....	7
Working with dat files in TIBCO Designer .....	7
Location .....	9
<b>Chapter 2 TIBCO ActiveMatrix BusinessWorks: Working with the Customer Component Interface</b> .....	<b>11</b>
Example Description .....	12
Setting up the Example .....	13
Deploying and Running the Example .....	14
Expected Results .....	16
<b>Chapter 3 TIBCO ActiveMatrix BusinessWorks: Working with the SalesOrder Component Interface</b> .....	<b>17</b>
Example Description .....	18
Setting up the Example .....	19
Testing the Example .....	20
Expected Results .....	21

<b>Chapter 4 TIBCO ActiveMatrix BusinessWorks: Working with the PurchaseOrder Component Interface.</b>	<b>23</b>
Example Description	24
Setting up the Example	25
Testing the Example	26
Expected Results	27
 <b>Chapter 5 TIBCO ActiveMatrix BusinessWorks: Working with the Employee Component Interface.</b>	 <b>29</b>
Example Description	30
Setting up the Example	31
Testing the Example	33
Expected Results	36
 <b>Chapter 6 TIBCO ActiveMatrix BusinessWorks: Working with the ItemMaster Component Interface.</b>	 <b>39</b>
Example Description	40
Setting up the Example	41
Testing the Example	42
Expected Results	45
 <b>Chapter 7 TIBCO ActiveMatrix BusinessWorks: Working with the UOM Message</b>	 <b>47</b>
Example Description	48
Setting up the Example	49
Preparing PeopleSoft Integration Broker	50
Integration Scenario Between the Adapter and PeopleSoft Integration Broker	50
Testing the Example	53
Expected Results	54
 <b>Chapter 8 TIBCO ActiveMatrix BusinessWorks: Working with the Contact Message</b>	 <b>56</b>
Example Description	57
Setting up the Example	58
Preparing PeopleSoft Integration Broker	59
Integration Scenario Between the Adapter and PeopleSoft Integration Broker	60
Testing the Example	63
Expected Results	64

<b>Chapter 9 TIBCO ActiveMatrix BusinessWorks: Working with the Country Message</b>	<b>68</b>
Example Description	69
Setting up the Example	70
Preparing PeopleSoft Integration Broker	71
Integration Scenario Between the Adapter and PeopleSoft Integration Broker	71
Testing the Example.	75
Expected Results	76
<b>Index</b>	<b>79</b>



# Figures

Figure 1      Update Operation Using a Publish-Subscribe Scenario. . . . . 3

Figure 2      Update Operation Using an Request-Response Scenario. . . . . 3





# Tables

Table 1      General Typographical Conventions ..... xiv

Table 2      PeopleTools Versions ..... 5



# Preface

TIBCO ActiveMatrix Adapter for PeopleSoft is a bi-directional gateway between the PeopleSoft application and applications configured for the TIBCO environment. It is assumed that you are familiar with the basic concepts of the TIBCO environment and PeopleSoft technology.

## Topics

---

- [Related Documentation, page xii](#)
- [Typographical Conventions, page xiv](#)
- [Terminology and Acronyms, page xvi](#)
- [Terminology and Acronyms, page xvi](#)

## Related Documentation

---

This section lists documentation resources you may find useful.

### TIBCO ActiveMatrix Adapter for PeopleSoft Documentation

The following documents form the TIBCO ActiveMatrix Adapter for PeopleSoft documentation set:

- TIBCO ActiveMatrix Adapter for PeopleSoft *Concepts* Read this manual to familiarize yourself with the product and its uses.
- TIBCO ActiveMatrix Adapter for PeopleSoft *Installation* Read this manual to learn how to install TIBCO ActiveMatrix Adapter for PeopleSoft.
- TIBCO ActiveMatrix Adapter for PeopleSoft *Configuration and Deployment* Read this manual for instructions on how to create, configure, and deploy adapter projects.
- TIBCO ActiveMatrix Adapter for PeopleSoft *Examples* Read this manual to work through the examples provided with the adapter.
- TIBCO ActiveMatrix Adapter for PeopleSoft *Release Notes* Read this document for information about new features, deprecated features, and known and closed issues.

### Other TIBCO Product Documentation

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

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

- TIBCO ActiveMatrix® Service Bus
- TIBCO Business Studio™

## Third-Party Documentation

You may also find it useful to read the following documentation:

- “Integration Tools: PeopleSoft Component Interfaces” in the *PeopleTools PeopleBook*. Topics include an explanation of Component Interfaces, how to create them, how to test them using PeopleTools, and how to validate that a Component Interface is compatible with the underlying component.
- “PeopleSoft Integration Broker” in the *PeopleTools PeopleBook*. Topics include an explanation of Integration Broker, the components involved, and how to use the components in Application Messaging.
- “PeopleSoft Platforms”, a link from the PeopleSoft Customer Connection Website (see <http://www.peoplesoft.com>), for the versions of the databases supported by PeopleSoft.
- “PeopleSoft Internet Architecture Administrating”, a link from the Peoplebook- PeopleTools-Administration Tools website (see <http://www.peoplesoft.com>), for information on configuring a JOLT Listener port.




# Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>TIBCO_HOME</i> <i>ENV_HOME</i> <i>PS_HOME</i>	<p>Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as <i>TIBCO_HOME</i>. The value of <i>TIBCO_HOME</i> depends on the operating system. For example, on Windows systems, the default value is C:\tibco.</p> <p>Other TIBCO products are installed into an installation environment. Incompatible products and multiple instances of the same product are installed into different installation environments. The directory into which such products are installed is referenced in documentation as <i>ENV_HOME</i>. The value of <i>ENV_HOME</i> depends on the operating system. For example, on Windows systems the default value is C:\tibco.</p> <p><i>PS_HOME</i> is the directory where the PeopleSoft Application server is installed. For example, if the PeopleSoft Application server is installed at C:\PT8.49 on Windows systems, the value of <i>PS_HOME</i> is C:\PT8.49</p>
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use MyCommand to start the foo process.</p>
<b>bold code font</b>	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none"><li>• In procedures, to indicate what a user types. For example: Type <b>admin</b>.</li><li>• In large code samples, to indicate the parts of the sample that are of particular interest.</li><li>• In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [<b>enable</b>   disable]</li></ul>

Table 1 General Typographical Conventions (Cont'd)

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

# Terminology and Acronyms

The following acronyms are used in this manual:

Acronym	Meaning
GUI	Graphical User Interface
RV	Refers to TIBCO Rendezvous reliable message quality of service, as opposed to certified message
RPC	Remote Procedural Call
TRA	TIBCO Runtime Agent
JMS	Java Message Service
Component Interface (CI)	A Component Interface is a PeopleTools object that you create in PeopleSoft Application Designer. It exposes a PeopleSoft component for synchronous access from another application. External applications need not be concerned with the details of page structures and component definitions in order to access the underlying data and business logic through Component Interfaces.
TIB Work Page	This page is bundled with the TIB_PS8_ADAPTER project. It implements the logic for capturing desired fields for publication at runtime. The page must be attached to the component from which data is captured.
Inbound	Events coming into the adapter. It refers to the inflow of data into the PeopleSoft application. It is applicable to Subscription and Request-Response services.
Outbound	Events going out from the adapter. It refers to data being captured from the PeopleSoft application and being sent out. It is applicable to the Publication Service.
IB	Integration Broker



## How to Contact TIBCO Support

---

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

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:  
<http://www.tibco.com/services/support>
- If you already have a valid maintenance or support contract, visit this site:  
<https://support.tibco.com>
- Entry to this site requires a user name and password. If you do not have a user name, you can request one.



## Chapter 1 **Introduction**

This chapter contains an overview of the examples and the prerequisites needed to run the examples.

### Topics

---

- [Overview, page 2](#)
- [Prerequisites, page 5](#)

## Overview

---

The examples in this guide demonstrate the integration between the adapter and two TIBCO products, TIBCO ActiveMatrix BusinessWorks and TIBCO IntegrationManager. We have chosen commonly used Component Interfaces to build an end-to-end enterprise wide integration and demonstrate the adapter capabilities.

These Component Interfaces are:

- Customer
- SalesOrder
- PurchaseOrder
- Employee
- ItemMaster
- UOM (Unit of Measure)
- Country
- Contact

The examples provide different operations on each of these Component Interfaces. A brief description of these operations is given next:

### Get

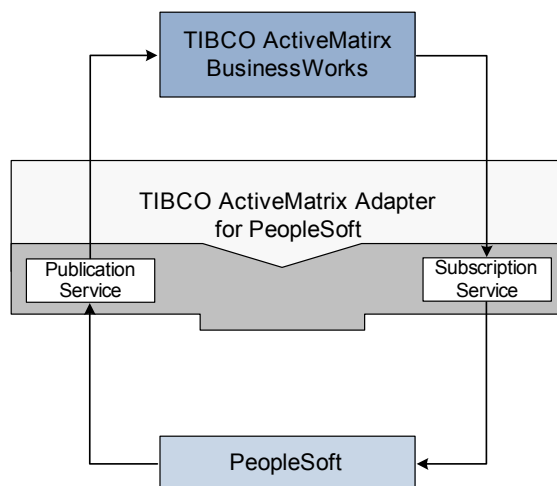
In this operation, TIBCO ActiveMatrix BusinessWorks requests data to be fetched from the PeopleSoft application. The Request-Response Service is used to simulate this. TIBCO ActiveMatrix BusinessWorks reads the request from an XML file and sends the request. The corresponding Request-Response Service is executed on the adapter and the result is returned to TIBCO ActiveMatrix BusinessWorks. TIBCO ActiveMatrix BusinessWorks maps the results according to business requirements and the data is written to an XML file.

### Update

Another important requirement in an integration scenario is to update data in an existing record. The Update operation is simulated in two ways.

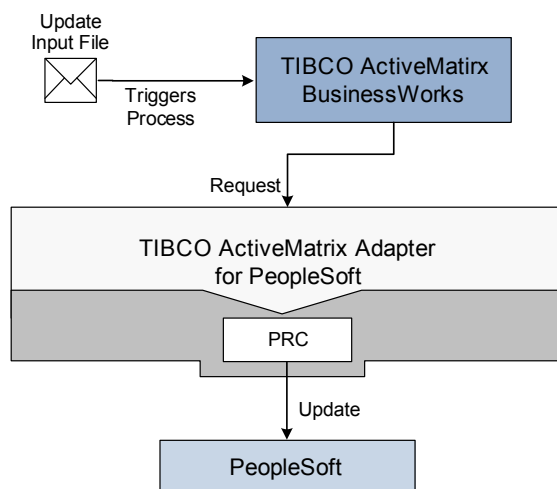
- The first is a publish-subscribe scenario. Data is published by the adapter's Publication service; the values that need to be updated are modified using a TIBCO ActiveMatrix BusinessWorks process. The updated message is sent to the adapter's Subscription Service and the service updates the values in PeopleSoft.

Figure 1 Update Operation Using a Publish-Subscribe Scenario



- The second is a Request-Response scenario. The values are modified in the mapper class in TIBCO ActiveMatrix BusinessWorks and the request is sent to the adapter. The Request-Response service of the adapter receives the request and updates the values in PeopleSoft.

Figure 2 Update Operation Using an Request-Response Scenario



The publish-subscribe scenario is used for the Customer, PurchaseOrder and ItemMaster Component Interfaces. The Request-Response scenario is used for the Employee and SalesOrder Component Interfaces.

**Sync**

In this operation, TIBCO ActiveMatrix Adapter for PeopleSoft publishes a message if there are any changes in the PeopleSoft target application. TIBCO ActiveMatrix BusinessWorks receives this message, does the necessary mapping of the fields and writes them to an XML file.

For publishing an event from PeopleSoft, refer to *TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*.

## Prerequisites

The following table shows the Component Interface, the corresponding PeopleSoft module and the PeopleTools version used to develop the respective Component Interface.

*Table 2 PeopleTools Versions*

Component Interface/ Application Message	Module	PeopleTools Version
Customer	Financials/SCM 8	8.18
SalesOrder	Financials/SCM 8	8.41
PurchaseOrder	Financials/SCM 8	8.18
Employee	HRMS 8.9	8.45
ItemMaster	CRM 8.9	8.45
UOM	CRM 8.9	8.45
Contact	CRM 8.9	8.45
Country	HRMS /CRM 8.9	8.45

The Component Interfaces developed with PeopleTools 8.18 can be used with PeopleTools versions 8.1x and the Component Interfaces developed using PeopleTools 8.40 and 8.41/8.45 can be used with PeopleTools version 8.4x subject to the condition that the adapter supports the PeopleTools version.



To make the examples work with later versions of PeopleTools, you may need to make changes to the Component Interfaces.

The adapter software includes two pre-built PeopleSoft projects (TIB\_EXAMPLES) that contain the Component Interfaces to be used in PeopleTools 8.1x and PeopleTools 8.4x. You must set up the projects in your PeopleSoft database before running the examples.

The TIB\_EXAMPLES project to be imported into PeopleTools 8.1x is located in `<adapter_home>\examples\TIB_EXAMPLES_PT81x` folder.

The TIB\_EXAMPLES project to be imported into PeopleTools 8.4x is located in `<adapter_home>\examples\TIB_EXAMPLES_PT84x` folder.

To import the project into PeopleTools 8.1x/8.4x:

1. Open PeopleTools Application Designer and log into the server where you want to add the project.
2. Verify that there is no existing TIB\_EXAMPLES project in the database.
  - a. Select **File>Open**.
  - b. In the Open Object dialog box, select Object Type = Project and then enter **TIB\_EXAMPLES**. If there is no existing TIB\_EXAMPLES project, skip to [step 3](#).
  - c. Open the existing TIB\_EXAMPLES project and delete all the Component Interfaces.
  - d. Delete the TIB\_EXAMPLES project itself.
  - e. If necessary, also delete the cache.
  - f. Save the project.
  - g. Exit PeopleTools Application Designer and then login again.
3. Import the project.
  - a. In PeopleTools 8.1x, select **File>Copy Project from File ...** .  
In PeopleTools 8.4x, select **Tools>Copy Project>From File ....**
  - b. In the dialog box that appears, browse to locate the directory or enter the name of the directory in which the project is located. If you are using PeopleTools 8.1x, the directory will be `<adapter_home>\examples\TIB_EXAMPLES_PT81x`. If you are using PeopleTools 8.4x, the directory will be `<adapter_home>\examples\TIB_EXAMPLES_PT84x`. The TIB\_EXAMPLES project appears in the list.
  - c. Select the TIB\_EXAMPLES project to open it.
  - d. In the dialog box, make sure all the object types are selected, then click **Copy**.
4. Go to **PeopleTools > Maintain Security** and grant permissions to work on the Component Interfaces and menu you have just imported. Make sure full access is given to all the Component Interfaces for the account being used by the adapter. Set permissions from **PeopleTools > Security > Permissions & Roles > Permissions Lists**. Choose the appropriate permission list for the account used by the adapter.



For additional information, refer to the *Loading the Adapter PeopleSoft Project* section, in the *Installation* chapter of the *TIBCO ActiveMatrix Adapter for PeopleSoft Installation*.





While running the examples with a particular version of PeopleTools, make sure that the `classpath` in the `.tra` file of the example is pointing to the correct version of `PSJOA.JAR`.

## Using TIBCO ActiveMatrix BusinessWorks

If you are using the adapter with TIBCO ActiveMatrix BusinessWorks, the following software must be installed to run the examples:

- TIBCO ActiveMatrix BusinessWorks
- TIBCO Administrator
- TIBCO ActiveMatrix Adapter for PeopleSoft
- Appropriate PeopleSoft Application with corresponding version of PeopleTools listed in [Table 2 on page 5](#).
- TIBCO Runtime Agent
- TIBCO Enterprise Message Service (EMS)



The EMS server must be running and accessible to the machine on which the adapter is installed.

The TIBCO ActiveMatrix BusinessWorks examples use TIBCO Designer to create an Enterprise Archive File (EAR) and TIBCO Administrator to deploy the EAR file.

In TIBCO Administrator, make sure all software components needed by the adapter instance are installed on one or more machines that are part of a TIBCO Administration Domain and that the software is registered in the domain.

Use the TIBCO Domain Utility to add a machine to a TIBCO Administration Domain.

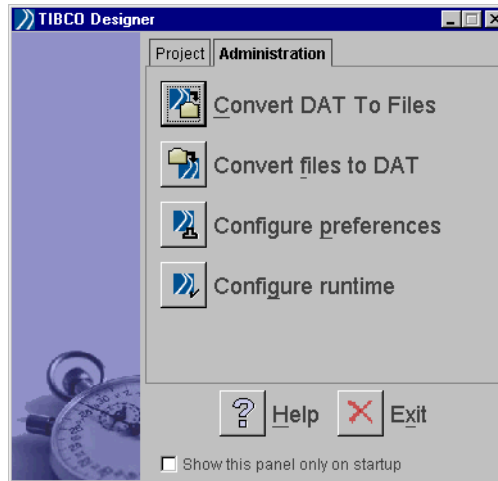
These topics are explained in the TIBCO Administrator documentation set.

## Working with dat files in TIBCO Designer

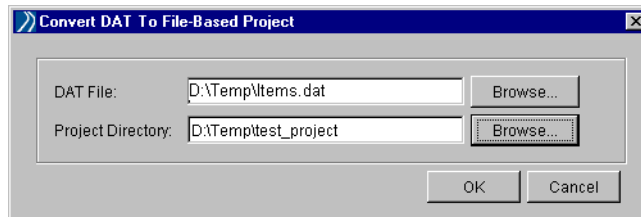
The dat files included in the adapter installation are used by all examples except the TIBCO ActiveMatrix BusinessWorks examples.

You cannot directly open a dat file in TIBCO Designer and make modifications to the configurations. To be able to do this, you need to carry out the following steps:

1. Convert the dat file to a multi-file project.
  - a. Open Designer. In the first screen that is displayed, click **Administration**. The screen is shown next.



- b. Then click the **Convert DAT to Files** icon. In the window that is displayed, browse and select the dat file you wish to convert to a multi-file project. Click **OK**.



2. Click the **Open existing project** icon from the first TIBCO Designer screen. Browse to the directory where the converted multi-file project is saved.
3. Make configuration changes as per your requirements.
4. Export the multi-file project to a dat.
5. Select **Project>Export Full Project** from the menu. Browse and select the location of the directory you wish to save the dat file to. Ensure that the directory is different from the multi-file project. Enter the name of the project and click **OK**.

## Location

The files for the examples are in the following location:

For TIBCO ActiveMatrix BusinessWorks examples:

`<adapter_home>\examples\BusinessWorks\<ComponentInterfaceName>`



## Chapter 2

# **TIBCO ActiveMatrix BusinessWorks: Working with the Customer Component Interface**

This example shows how to use several adapter services within a TIBCO ActiveMatrix BusinessWorks process to get, update, and synchronize a Customer record stored in PeopleSoft. The example is deployed and run using the TIBCO Administrator GUI.

## Topics

---

- [Example Description, page 12](#)
- [Setting up the Example, page 13](#)
- [Deploying and Running the Example, page 14](#)
- [Expected Results, page 16](#)

## Example Description

---

This example shows how the adapter's Publication, Subscription and Request-Response services can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has three processes, `GetProcess`, `UpdateProcess` and `SyncProcess`.

- `GetProcess` uses the adapter's Request-Response service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`GetProcess` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

- `UpdateProcess` uses the adapter's Publication and Subscription services to update PeopleSoft.

In a typical business scenario, an adapter to another application (for example, SAP or Siebel) publishes a message into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Subscription service to update the data in PeopleSoft. In Customer example provided, the publisher is also a PeopleSoft adapter.

Alternatively, you may specify key values in the input XML file.

`UpdateProcess` queries PeopleSoft based on the key values. The Customer details are updated in PeopleSoft.

- `SyncProcess` uses the adapter's Publication service to send a message out of PeopleSoft whenever Customer Component Interface details are modified or updated in the target PeopleSoft application. Any change to the data in PeopleSoft triggers the `SyncProcess`.

On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `Customer.zip` file. The file is located in the `<adapter_home>/examples/BusinessWorks/Customer` folder.
4. Go to the Publication service instance.
5. Click the **Design-time Connection** tab.
6. Go to **TIB\_CUSTOMER** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **Customer**.
8. Select **Customer** in the project panel.
9. Click **Build Archive** in the **Configuration** tab. This creates an archive file.
10. Exit TIBCO Designer.

## Deploying and Running the Example

---

Perform the following tasks to deploy and run the example.

### Task A Deploy the Example

In TIBCO Administrator:

1. Click on the **Installed Software** link on the left panel. Check if TIBCO ActiveMatrix Adapter for PeopleSoft is registered. If the software is not registered, register by clicking the **Add Custom Software** button. Please refer to the TIBCO Administrator documentation for more details on adding custom software.
2. Go to **Application Management**. Click **New Application**. Upload the EAR file created in the previous section and click **OK**. On the following screen, clear the **Quick Configure** check box and then click **Save**.
3. Click **Configuration** in the newly created application. Click on the top level application name in the configuration view.
  - a. Go to the **Advanced** tab and provide the values for:
 

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```
  - b. Click **Save**. The Configuration screen appears.
  - c. Expand the Enterprise archive link, so that adapter instances, `pubCI.aar`, `sub.aar` and `rpc.aar` are visible.
  - d. Click **pubCI.aar** and then click the **Add to Additional Machines** button.
  - e. Select the machine and click **OK**. In the following screen, click **Save**.
  - f. Repeat steps d and e for `sub.aar` and `rpc.aar`.
  - g. In the Configuration screen click **Deploy**, which will take you to the next screen. The **Start successfully deployed services** check box is selected by default. Click **OK**. The `.tra` files are created in `<adapter_home>/bin/domain/<DomainName>`.
4. Once the deployment is complete, click **Service Instances** under the application. The adapter is listed. Select the adapter and click **Load Selected**. Both the customer creation and customer fetching process is started. The adapter instance is also started.



## Task B Run the Example

To run each example:

1. To trigger `GetProcess`, you must enter the key values in the `Get_Input.xml` file located in the `<adapter_home>/examples/BusinessWorks/Customer` folder. For example key values for the following parameters can be entered:

```
SETID
TIB_CUSTOMER
```

This fetches the customer details from PeopleSoft based on the details as given in the XML file. After the customer details are fetched from PeopleSoft, the `Get_Output.xml` file is written to the location `<adapter_home>/examples/BusinessWorks/Customer`

2. To trigger `UpdateProcess`, you must enter the data in the `input_render.xml` file located in the `<adapter_home>/examples/BusinessWorks/Customer` folder. For example:

```
<PROPERTIES>
  <SETID>CRM01</SETID>
  <PRODUCT_ID>10000</PRODUCT_ID>
  <PRODUCT_ID_O>10000</PRODUCT_ID_O>
  <DESCR>CRM01</DESCR>
  <TIB_PROD_PRICE__KIT_COMPS_PRICE>
  <OPRN_CODE>U</OPRN_CODE>
  <PROD_COMPONENT_ID>WE</PROD_COMPONENT_ID>
  <DATE_IN_EFFECT>U</DATE_IN_EFFECT>
  <PROD_KIT_COMP_PRICE>12.78</PROD_KIT_COMP_PRICE>
  <DATE_OBSOLUTE>12</DATE_OBSOLUTE>
  </TIB_PROD_PRICE__KIT_COMPS_PRICE>
  <TIB_PROD_PRICE__PROD_UOM>
  <OPRN_CODE>U</OPRN_CODE>
  <UNIT_OF_MEASURE>VINIBA</UNIT_OF_MEASURE>
  <TIB_PROD_PRICE__PROD_PRICE_BU>
  <OPRN_CODE>U</OPRN_CODE>
  <BUSINESS_UNIT_IN>QUOT</BUSINESS_UNIT_IN>
  <CURRENCY_CD>U</CURRENCY_CD>
  <TIB_PROD_PRICE__PROD_PRICE>
  <OPRN_CODE>U</OPRN_CODE>
  <EFFDT>U</EFFDT>
  <EFF_STATUS>QUOT</EFF_STATUS>
  <LIST_PRICE>1.1</LIST_PRICE>
  <UNIT_CAST>1.1</UNIT_CAST>
  <MSG_SUG_RTL_PRC>1.1</MSG_SUG_RTL_PRC>
  </TIB_PROD_PRICE__PROD_PRICE>
  </TIB_PROD_PRICE__PROD_PRICE_BU>
  </TIB_PROD_PRICE__PROD_UOM>
</PROPERTIES>
```

This creates a customer record in PeopleSoft.

3. To trigger `SyncProcess`, you must modify the data in PeopleSoft. For example you may modify any customer record. This triggers the `SyncProcess` and writes the data to the `Sync_output.xml` file.

## Expected Results

---

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/BusinessWorks/Customer`

- GetProcess data is fetched based on the SETID and TIB\_CUSTOMER parameters specified in the `Get_Input.xml` file and is written into the `Get_output.xml` file. The `Get_output.xml` file should display the following customer record:

```
<Customers>
<PROPERTIES>
  <SETID>SHARE</SETID>
  <CUST_ID>1000</CUST_ID>
  <NAME1>Alliance & Group12</NAME1>
  <NAME_SHORT>Alliance</NAME_SHORT>
  <ADDRESS_SEQ_NO>1</ADDRESS_SEQ_NO>
  <TIB_CUSTOMER__CUST_ADDR_SEQ>
    <OPRN_CODE>U</OPRN_CODE>
    <SETID>SHARE</SETID>
    <CUST_ID>1000</CUST_ID>
    <ADDRESS_SEQ_NO>1</ADDRESS_SEQ_NO>
    <TIB_CUSTOMER__CUST_ADDRESS>
      <OPRN_CODE>U</OPRN_CODE>
      <EFF_DT>07/19/2002</EFF_DT>
      <EFF_STATUS>A</EFF_STATUS>
      <ADDRESS1>14410 Union Ave</ADDRESS1>
      <ADDRESS2></ADDRESS2>
      <CITY>San Jose</CITY>
      <STATE>CA</STATE>
      <POSTAL>95124</POSTAL>
    </TIB_CUSTOMER__CUST_ADDRESS>
  </TIB_CUSTOMER__CUST_ADDR_SEQ>
</PROPERTIES>
</Customers>
```

- UpdateProcess data is updated based on the SETID and TIB\_CUSTOMER parameters specified in the `input_render.xml` file and is updated in PeopleSoft.
- SyncProcess output is stored in the `Sync_output.xml` file.

## Chapter 3

# **TIBCO ActiveMatrix BusinessWorks: Working with the SalesOrder Component Interface**

This example shows how to use several adapter services within a TIBCO ActiveMatrix BusinessWorks process to get, update, and synchronize a SalesOrder record stored in PeopleSoft.

## Topics

---

- [Example Description, page 18](#)
- [Setting up the Example, page 19](#)
- [Testing the Example, page 20](#)
- [Expected Results, page 21](#)

## Example Description

---

This example shows how the adapter's Publication, Subscription and Request-Response services can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has three processes, `GetProcess`, `UpdateProcess` and `SyncProcess`.

- `GetProcess` uses the adapter's Request-Response service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`GetProcess` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

- `UpdateProcess` uses the adapter's Request-Response service to update PeopleSoft.

In a typical business scenario, an adapter to another application (for example, SAP or Siebel) publishes a message into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Request-Response service to update the data in PeopleSoft and sends a reply to TIBCO ActiveMatrix BusinessWorks.

Alternatively, you may specify key values in the input XML file.

`UpdateProcess` queries PeopleSoft based on the key values. The SalesOrder details are updated in PeopleSoft.

- `SyncProcess` uses the adapter's Publication service to send a message out of PeopleSoft whenever SalesOrder Component Interface details are modified or updated in the target PeopleSoft application. Any change to the data in PeopleSoft triggers the `SyncProcess`.

On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `SalesOrder.zip` file. The file is located in the folder  
`<adapter_home>/examples/BusinessWorks/SalesOrder`

4. Substitute the values of the following global variables for design-time connection:

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.

5. Go to the Publication service instance.
6. Go to **TIB\_SalesOrder** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **SalesOrder** in a new folder.
8. Exit TIBCO Designer.

### Start the TIBCO EMS Server:

This example uses JMS as the transport type. Ensure that the TIBCO EMS Server is running and accessible to the machine on which the adapter is installed before running any of the processes.

## Testing the Example

---

In TIBCO Designer:

1. From the project panel, select the process you want to test. For example, `GetProcess`.
2. Click the **Set Breakpoints** icon.
3. In the window that appears, choose **Select All**, then click **OK**.
4. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
5. Click the **Start testing viewed process** button.
6. In the process selection window that appears, the `GetProcess` is selected by default. Click **Load Selected**. The process is now in Test mode.
7. Click the **File Poller** icon. In the configuration panel, `File Name` field, click **Browse** and select the input XML file. Click **Apply**. Any change to the input XML file starts the process.
8. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
9. Step through the process once more.
10. Click the **Stop Testing** icon to return to design mode.

The above steps describe the testing for `GetProcess`. Follow the same steps to run the `SyncProcess` and `UpdateProcess`. The `SyncProcess` does not have an input XML file. Any change to the data in PeopleSoft will invoke the `SyncProcess`.

## Expected Results

---

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/BusinessWorks/SalesOrder`

- GetProcess data is fetched based on the SETID and TIB\_SalesOrder parameters specified in the `Get_Input.xml` file and is written into the `Get_output.xml` file. The `Get_output.xml` file should display the following SalesOrder record:

```
<SalesOrder>
  <PROPERTIES>
    <BUSINESS_UNIT>US001</BUSINESS_UNIT>
    <ORDER_NO>CEN0001</ORDER_NO>
    <DESCR_0></DESCR_0>
    <ADDRESS1></ADDRESS1>
    <COUNTRY_SHIP_FROM></COUNTRY_SHIP_FROM>
    <SOLD_TO_CUST_ID_0>US011</SOLD_TO_CUST_ID_0>
    <BILL_TO_CUST_ID_1>US011</BILL_TO_CUST_ID_1>
    <ORDENT_HDR_NAV>0000</ORDENT_HDR_NAV>
    <COMMSN_LEVEL>1</COMMSN_LEVEL>
    <COMMSN_METHOD>C</COMMSN_METHOD>
    <EXPORT>N</EXPORT>
    <ORDER_DATE>09/30/2002</ORDER_DATE>
    <ORDER_GRP>STD</ORDER_GRP>
    <ORDER_STATUS>0</ORDER_STATUS>
    <PROD_ID_SRC>S</PROD_ID_SRC>
  <TIB_SALESORDER>
    <OPRN_CODE>U</OPRN_CODE>
    <COPY_LINE_SEL></COPY_LINE_SEL>
    <ORDER_NO></ORDER_NO>
    <ORDER_DATE></ORDER_DATE>
    <DESCR></DESCR>
    <PROD_ID_ENTERED></PROD_ID_ENTERED>
    <QTY_ORDERED>0</QTY_ORDERED>
    <UNIT_OF_MEASURE></UNIT_OF_MEASURE>
  </TIB_SALESORDER>
</PROPERTIES>
</SalesOrder>
```

- UpdateProcess data is updated based on the SETID and TIB\_SalesOrder parameters specified in the `input_render.xml` file and is updated in PeopleSoft.
- SyncProcess output is stored in the `Sync_output.xml` file.





## Chapter 4

# **TIBCO ActiveMatrix BusinessWorks: Working with the PurchaseOrder Component Interface**

This example shows how to use several adapter services within a TIBCO ActiveMatrix BusinessWorks process to get, update, and synchronize a PurchaseOrder record stored in PeopleSoft.

## Topics

---

- [Example Description, page 24](#)
- [Setting up the Example, page 25](#)
- [Testing the Example, page 26](#)
- [Expected Results, page 27](#)

## Example Description

---

This example shows how the adapter's Publication, Subscription and Request-Response services can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has three processes, `GetProcess`, `UpdateProcess` and `SyncProcess`.

- `GetProcess` uses the adapter's Request-Response service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`GetProcess` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

- `UpdateProcess` uses the adapter's Request-Response service to update PeopleSoft.

In a typical business scenario, an adapter to another application (for example, SAP or Siebel) publishes a message into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Request-Response service to update the data in PeopleSoft and sends a reply to TIBCO ActiveMatrix BusinessWorks.

Alternatively, you may specify key values in the input XML file. `UpdateProcess` queries PeopleSoft based on the key values. The PurchaseOrder details are updated in PeopleSoft.

- `SyncProcess` uses the adapter's Publication service to send a message out of PeopleSoft whenever PurchaseOrder Component Interface details are modified or updated in the target PeopleSoft application. Any change to the data in PeopleSoft triggers the `SyncProcess`.

On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `PurchaseOrder.zip` file. The file is located in the folder  
`<adapter_home>/examples/BusinessWorks/PurchaseOrder`

4. Substitute the values of the following global variables for design-time connection:

```
adpsft8.connection.appserver  
adpsft8.connection.login  
adpsft8.connection.password  
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.

5. Go to the Publication service instance.
6. Go to **TIB\_Purchase\_Order** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **PurchaseOrder** in a new folder and exit TIBCO Designer.

### Start the TIBCO EMS Server:

This example uses JMS as the transport type. Ensure that the TIBCO EMS Server is running and accessible to the machine on which the adapter is installed before running any of the processes.

## Testing the Example

---

In TIBCO Designer:

1. From the project panel, select the process you want to test. For example, `GetProcess`.
2. Click the **Set Breakpoints** icon.
3. In the window that appears, choose **Select All**, then click **OK**.
4. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
5. Click the **Start testing viewed process** button.
6. In the process selection window that appears, the `GetProcess` is selected by default. Click **Load Selected**. The process is now in Test mode.
7. Click the **File Poller** icon. In the configuration panel, `File Name` field, click **Browse** and select the input XML file. Click **Apply**. Any change to the input XML file starts the process.
8. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
9. Step through the process once more.
10. Click the **Stop Testing** icon to return to design mode.
11. The above steps describe the testing for `GetProcess`. Follow the same steps to run the `SyncProcess` and `UpdateProcess`. The `SyncProcess` does not have an input XML file. Any change to the data in PeopleSoft will invoke the `SyncProcess`.

## Expected Results

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/BusinessWorks/PurchaseOrder`

- GetProcess data is fetched based on the SETID and TIB\_Purchase\_Order parameters specified in the Get\_Input.xml file and is written into the Get\_output.xml file. The Get\_output.xml file should display the following PurchaseOrder record:

```
<PurchaseOrders>
  <PROPERTIES>
    <BUSINESS_UNIT_2>US001</BUSINESS_UNIT_2>
    <PO_ID_2>0000000001</PO_ID_2>
    <CHNG_ORD_BATCH>0</CHNG_ORD_BATCH>
    <PO_TYPE>GEN</PO_TYPE>
    <HOLD_STATUS>N</HOLD_STATUS>
    <DISP_ACTION>Y</DISP_ACTION>
    <DISP_METHOD>PRN</DISP_METHOD>
    <PO_DT>08/10/2000</PO_DT>
    <TIB_PURCHASE_ORDER_PO_LINE>
      <OPRN_CODE>U</OPRN_CODE>
      <UNIT_OF_MEASURE>EA</UNIT_OF_MEASURE>
      <QTY_TYPE>L</QTY_TYPE>
      <PRICE_DT_TYPE>P</PRICE_DT_TYPE>
      <MFG_ID></MFG_ID>
      <CURRENCY_CD>USD</CURRENCY_CD>
      <CURRENCY_CD_BASE>USD</CURRENCY_CD_BASE>
      <DELETING_LINE></DELETING_LINE>
      <TIB_PURCHASE_ORDER_PO_LINE_MISC>
        <OPRN_CODE>U</OPRN_CODE>
        <MISC_CHARGE_CODE></MISC_CHARGE_CODE>
        <MISC_CHARGE_ORIGN></MISC_CHARGE_ORIGN>
        <CHARGE_AMOUNT>0</CHARGE_AMOUNT>
        <CURRENCY_CD></CURRENCY_CD>
        <LC_COMP_CALC></LC_COMP_CALC>
        <UNIT_PRICE>0</UNIT_PRICE>
        <FLAT_AMOUNT>0</FLAT_AMOUNT>
        <PCT_VALUE>0</PCT_VALUE>
        <UNIT_OF_MEASURE></UNIT_OF_MEASURE>
      </TIB_PURCHASE_ORDER_PO_LINE_MISC>
    </TIB_PURCHASE_ORDER_PO_LINE>
  </PROPERTIES>
</PurchaseOrders>
```

- UpdateProcess data is updated based on the SETID and TIB\_Purchase\_Order parameters specified in the input\_render.xml file and is updated in PeopleSoft.
- SyncProcess output is stored in the Sync\_output.xml file.



## Chapter 5

# **TIBCO ActiveMatrix BusinessWorks: Working with the Employee Component Interface**

This example shows how to use several adapter services within a TIBCO BusinessWorks process to get, update, and synchronize an Employee record stored in PeopleSoft.

## Topics

---

- [Example Description, page 30](#)
- [Setting up the Example, page 31](#)
- [Testing the Example, page 33](#)
- [Expected Results, page 36](#)

## Example Description

---

This example shows how the adapter's Publication, Subscription and Request-Response services can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has three processes, `GetProcess`, `UpdateProcess` and `SyncProcess`.

- `GetProcess` uses the adapter's Request-Response service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`GetProcess` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

- `UpdateProcess` uses the adapter's Publication and Subscription services to update PeopleSoft.

In a typical business scenario, an adapter publishes a message to another application (for example, SAP or Siebel) into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Subscription service to update the data in PeopleSoft.

In the Employee example provided, the publisher triggers the `UpdateProcess`. The data to be updated is specified in the input XML file. `UpdateProcess` queries PeopleSoft based on the key values specified in the input XML file and the Employee details are updated in PeopleSoft.

- `SyncProcess` uses the adapter's Publication service to send a message out of PeopleSoft whenever Employee Component Interface details are modified or updated in the target PeopleSoft application. Any change to the data in PeopleSoft triggers the `SyncProcess`.

On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.



## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `Employee.zip` file. The file is located in the folder `<adapter_home>/examples/BusinessWorks/Employee`
4. Substitute the values of the following global variables for design time connection for the pub, rpc, and sub services.

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```

You can also specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.

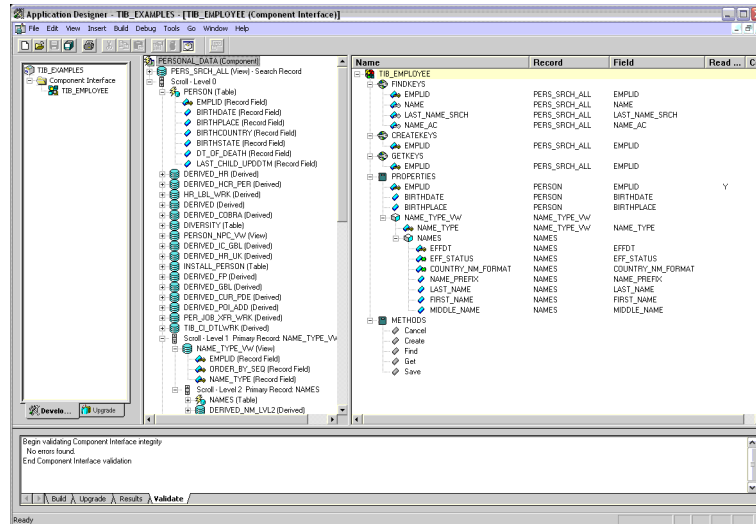
5. Ensure that the global variable `adpsft8DTARvService` is blank.

### Peoplesoft Application Setup:

In PeopleSoft Application Designer:

1. Open the `TIB_EXAMPLES` project.
2. Select **File>New>Component Interface**.
3. In the **Select Source Component for Component Interface** dialog, type `PERSONAL_DATA` in the **Name** field and click **Select**. Double click on the `PERSONAL_DATA` component.
4. The next dialog box asks "*Do you want to default the properties based on the underlying Component definition, PERSONAL\_DATA?*". Select **Yes**.

5. Delete unwanted fields and records from the component interface definition as shown below and save it as TIB\_EMPLOYEE.



6. Insert the TIB\_EMPLOYEE component interface into the project TIB\_EXAMPLES by pressing F7. Save the project.
7. Open the TIB\_EMPLOYEE component interface.
8. Double-click on the PERSONAL\_DATA component.
9. Insert the TIB\_CI\_ADAPTER\_WRK page in the component and save it.
10. Give permissions to the TIB\_CI\_ADAPTER\_WRK page for the PERSONAL\_DATA component. For more information on setting permissions, refer to *PeopleBooks*.
11. In TIBCO Designer:
  - a. Select the pub adapter instance. Go to **Design-time Connection** and connect to the DTA. Click **Save To Peoplesoft**.
  - b. Save as a multi-file project by the name **Employee** in a new folder and exit TIBCO Designer.

## Testing the Example

---

Perform the following task to test the example.

### Task A Test the Example (GetProcess)

Test the example for `GetProcess`, `UpdateProcess`, and `SyncProcess`.

In TIBCO Designer:

1. From the project panel, select `GetProcess`.
2. Click the **File Poller** icon. Select the location of the input XML file `get_input.xml`. The default location is `<adapter_home>\examples\businessworks\EMPLOYEE\Get_input.xml`. This XML file conforms to `<adapter_home>\examples\businessworks\EMPLOYEE\Get_input_schema.xsd`.
3. Specify the desired EMPID value in the `Get_input.xml` file.
4. Click the **Write File** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `Get_output.xml` file (This file is generated in case of a success).
5. Click the **Write-File-1** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `get_output_err.xml` (This file is generated in case of an error).
6. Select the adapter tester and start the `rpc` adapter instance.
7. Click the **Set Breakpoints** icon.
8. In the window that appears, choose **Select All**, then click **OK**.
9. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
10. Click the **Start testing viewed process** button. (Make sure that `GetProcess` is selected.)
11. In the process selection window that appears, click **Load Selected**. The process is now in the Test mode.
12. The tester picks up the values from the `Get_input.xml` file. In case of a success, it retrieves the employee details and generates the `Get_output.xml` file. In case of an error, it prints the error message in the `get_output_err.xml` file. Any change to the input XML file starts the process.

13. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
14. Step through the process once more.
15. Click the **Stop Testing** icon to return to design mode.
16. Stop the `rpc` adapter instance in the adapter tester.

### Task B Test the Example (SyncProcess)

In TIBCO Designer:

1. From the project panel, select the `SyncProcess` process.
2. Click the **Write File** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `Sync_output.xml` file.
3. Select the adapter tester and start the `pub` adapter instance.
4. Click the **Set Breakpoints** icon.
5. In the window that appears, choose **Select All**, then click **OK**.
6. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
7. Click the **Start testing viewed process** button. (Make sure that the `SyncProcess` is selected.)
8. In the process selection window that appears, click **Load Selected**. The process is now in the Test mode.
9. The `SyncProcess` does not have an input XML file. Any change to the employee data in PeopleSoft will invoke the `SyncProcess`. `Sync_output.xml` will have the corresponding message published to the adapter.
10. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
11. Step through the process once more.
12. Click the **Stop Testing** icon to return to design mode.
13. Stop the `pub` instance in the adapter tester.

### Task C Test the Example (UpdateProcess)

In TIBCO Designer:

1. From the project panel, select `UpdateProcess`.

2. Click the **Read File** icon. Select the **Input** tab, in the activity input expand the `ReadActivityInputClass`. In the **fileName** field, specify the entire path or location of the input `Update_input.xml` file. This XML file conforms to `<adapter_home>\examples\businessworks\EMPLOYEE\Update_input_schema.xsd`.
3. Select the adapter tester and start the pub and sub adapter instance.
4. Click the **Set Breakpoints** icon.
5. In the window that appears, choose **Select All**, then click **OK**.
6. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
7. Click the **Start testing viewed process** button. (Make sure that the `UpdateProcess` is selected.)
8. In the process selection window that appears, click **Load Selected**. The process is now in the Test mode.
9. Any change to the data in PeopleSoft will invoke the `UpdateProcess`. The tester picks up the value from the `Update_input.xml` file and updates the corresponding data in PeopleSoft.
10. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
11. Step through the process once more.
12. Click the **Stop Testing** icon to return to design mode.
13. Stop the pub and sub adapter instances in the adapter tester.

## Expected Results

---

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/BusinessWorks/EMPLOYEE`

- GetProcess data is fetched based on the EMPID parameter specified in the `Get_Input.xml` file and is written into the `Get_output.xml` file. The `Get_output.xml` file should display the following Employee record:

```
<?xml version="1.0" encoding=""?>
<EMPLOYEE>
  <EMPLID>FA1058</EMPLID>
  <BIRTHDATE>08/06/1982</BIRTHDATE>
  <BIRTHPLACE>New York City</BIRTHPLACE>
  <TIB_EMPLOYEE__NAME_TYPE_VW>
    <NAME_TYPE>PRI</NAME_TYPE>
    <TIB_EMPLOYEE__NAMES>
      <EFFDT>09/23/1999</EFFDT>
      <EFF_STATUS>A</EFF_STATUS>
      <COUNTRY_NM_FORMAT>001</COUNTRY_NM_FORMAT>
      <NAME_PREFIX>Mr</NAME_PREFIX>
      <LAST_NAME>WOODROW</LAST_NAME>
      <FIRST_NAME>AARON</FIRST_NAME>
      <MIDDLE_NAME/>
    </TIB_EMPLOYEE__NAMES>
  </TIB_EMPLOYEE__NAME_TYPE_VW>
  <TIB_EMPLOYEE__NAME_TYPE_VW>
    <NAME_TYPE>PRF</NAME_TYPE>
    <TIB_EMPLOYEE__NAMES>
      <EFFDT>09/23/1999</EFFDT>
      <EFF_STATUS>A</EFF_STATUS>
      <COUNTRY_NM_FORMAT>001</COUNTRY_NM_FORMAT>
      <NAME_PREFIX/>
      <LAST_NAME/>
      <FIRST_NAME/>
      <MIDDLE_NAME/>
    </TIB_EMPLOYEE__NAMES>
  </TIB_EMPLOYEE__NAME_TYPE_VW>
</EMPLOYEE>
```

- UpdateProcess data is updated based on the EMPID parameter specified in the `Update_input.xml` file and is updated in PeopleSoft.
- SyncProcess output is stored in the `Sync_output.xml` file. The `Sync_output.xml` file must look similar to the following:

```
<?xml version="1.0" encoding=""?>
<EMPLOYEE>
  <EMPLID>FA1058</EMPLID>
  <BIRTHDATE>1982-08-06</BIRTHDATE>
  <BIRTHPLACE>N Y City</BIRTHPLACE>
  <TIB_EMPLOYEE__NAME_TYPE_VW>
    <NAME_TYPE>PRI</NAME_TYPE>
```

```

<TIB_EMPLOYEE__NAMES>
  <EFFDT>1999-09-23</EFFDT>
  <EFF_STATUS>A</EFF_STATUS>
  <COUNTRY_NM_FORMAT>001</COUNTRY_NM_FORMAT>
  <NAME_PREFIX/>
  <LAST_NAME>WOODROW</LAST_NAME>
  <FIRST_NAME>AARON</FIRST_NAME>
  <MIDDLE_NAME/>
</TIB_EMPLOYEE__NAMES>
</TIB_EMPLOYEE__NAME_TYPE_VW>
<TIB_EMPLOYEE__NAME_TYPE_VW>
  <NAME_TYPE>PRF</NAME_TYPE>
  <TIB_EMPLOYEE__NAMES>
    <EFFDT>1999-09-23</EFFDT>
    <EFF_STATUS>A</EFF_STATUS>
    <COUNTRY_NM_FORMAT>001</COUNTRY_NM_FORMAT>
    <NAME_PREFIX/>
    <LAST_NAME/>
    <FIRST_NAME/>
    <MIDDLE_NAME/>
  </TIB_EMPLOYEE__NAMES>
</TIB_EMPLOYEE__NAME_TYPE_VW>
</EMPLOYEE>

```





## Chapter 6

## **TIBCO ActiveMatrix BusinessWorks: Working with the ItemMaster Component Interface**

This example shows how to use several adapter services within a TIBCO ActiveMatrix BusinessWorks process to get, update, and synchronize an ItemMaster record stored in PeopleSoft.

### Topics

---

- [Example Description, page 40](#)
- [Setting up the Example, page 41](#)
- [Testing the Example, page 42](#)
- [Expected Results, page 45](#)

## Example Description

---

This example shows how the adapter's Publication, Subscription and Request-Response services can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has three processes, `GetProcess`, `UpdateProcess` and `SyncProcess`.

- `GetProcess` uses the adapter's Request-Response service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`GetProcess` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

- `UpdateProcess` uses the adapter's Publication and Subscription services to update PeopleSoft.

In a typical business scenario, an adapter publishes a message to another application (for example, SAP or Siebel) into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Subscription service to update the data in PeopleSoft.

In the ItemMaster example provided, the publisher triggers the `UpdateProcess`. The data to be updated is specified in the input XML file. `UpdateProcess` queries PeopleSoft based on the key values specified in the input XML file and the ItemMaster details are updated in PeopleSoft.

- `SyncProcess` uses the adapter's Publication service to send a message out of PeopleSoft whenever ItemMaster Component Interface details are modified or updated in the target PeopleSoft application. Any change to the data in PeopleSoft triggers the `SyncProcess`.

On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `ItemMaster.zip` file. The file is located in the folder  
`<adapter_home>/examples/BusinessWorks/ItemMaster`

4. Substitute the values of the following global variables for design-time connection for the pub, sub, and rpc services:

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.

5. Ensure that the global variable `adpsft8DTARvService` is blank.
6. Select the pub adapter instance. Go to **Design-time Connection** and connect to the DTA. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **ItemMaster** in a new folder and exit TIBCO Designer.

### Peoplesoft Application Setup:

In PeopleSoft Application Designer:

1. Open `TIB_EXAMPLES` project.
2. Open the `TIB_INV_PROD_FAMILIES` component interface.
3. Double-click on the component `INV_PROD_FAMILIES`.
4. Insert the `TIB_CI_ADAPTER_WRK` page in the component and save it.
5. Give permissions to the `TIB_CI_ADAPTER_WRK` page for the `INV_PROD_FAMILIES` component. For more information on setting permissions, refer to *PeopleBooks*.

## Testing the Example

---

Perform the following task to test the example.

### Task A Test the Example (GetProcess)

In TIBCO Designer:

1. From the project panel, select `GetProcess`.
2. Click the **File Poller** icon. Select the location of the input XML file `get_input.xml`. The default location is `<adapter_home>\examples\businessworks\ITEMMASTER\Get_input.xml`. This XML file conforms to `<adapter_home>\examples\businessworks\ITEMMASTER\Get_input_schema.xsd`.
3. Specify the desired SETID and INV\_PROD\_FAM\_CD value in the `Get_input.xml` file.
4. Click the **Write File** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `Get_output.xml` file (This file is generated in case of a success).
5. Click the **Write-File-1** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `get_output_err.xml` (This file is generated in case of an error).
6. Select the adapter tester and start the rpc adapter instance.
7. Click the **Set Breakpoints** icon.
8. In the window that appears, choose **Select All**, then click **OK**.
9. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
10. Click the **Start testing viewed process** button. (Make sure that `GetProcess` is selected.)
11. In the process selection window that appears, click **Load Selected**. The process is now in Test mode.
12. The tester picks up the values from the `Get_input.xml` file, and in case of a success, it retrieves the item master details and generate the `Get_output.xml` file. In case of an error, it prints the error message in the `get_output_err.xml` file. Any change to the input XML file starts the process.

13. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
14. Step through the process once more.
15. Click the **Stop Testing** icon to return to design mode.
16. Stop the `rpc` adapter instance in the adapter tester.

### Task B Test the Example (SyncProcess)

In TIBCO Designer:

1. From the project panel, select the `SyncProcess` process.
2. Click the **Write File** icon. Select the **Input** tab and in the activity input expand the `WriteActivityInputTextClass`. In the **fileName** field, specify the entire path or location of the `Sync_output.xml` file.
3. Select the adapter tester and start the `pub` adapter instance.
4. Click the **Set Breakpoints** icon.
5. In the window that appears, choose **Select All**, then click **OK**.
6. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
7. Click the **Start testing viewed process** button. (Make sure that the `SyncProcess` is selected.)
8. In the process selection window that appears, click **Load Selected**. The process is now in the Test mode.
9. The `SyncProcess` does not have an input XML file. Any change to the data in PeopleSoft will invoke the `SyncProcess`. `Sync_output.xml` will have the corresponding message published to the adapter.
10. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
11. Step through the process once more.
12. Click the **Stop Testing** icon to return to design mode.
13. Stop the `pub` instance in the adapter tester.

### Task C Test the Example (UpdateProcess)

In TIBCO Designer:

1. From the project panel, select `UpdateProcess`.

2. Click the **Read File** icon. Select the **Input** tab, in the activity input expand the `ReadActivityInputClass`. In the **fileName** field, specify the entire path or location of the input `Update_input.xml` file. This XML file conforms to `<adapter_home>\examples\businessworks\ITEMMASTER\Schema.xsd`.
3. Select the adapter tester and start the pub and sub adapter instance.
4. Click the **Set Breakpoints** icon.
5. In the window that appears, choose **Select All**, then click **OK**.
6. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
7. Click the **Start testing viewed process** button. (Make sure that the `UpdateProcess` is selected.)
8. In the process selection window that appears, click **Load Selected**. The process is now in the Test mode.
9. Any change to the data in PeopleSoft will invoke the `UpdateProcess`.
10. The tester picks up the value from the `Update_input.xml` file and updates the corresponding data in PeopleSoft.
11. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
12. Step through the process once more.
13. Click the **Stop Testing** icon to return to design mode.
14. Stop the pub and sub adapter instances in the adapter tester.

## Expected Results

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/BusinessWorks/ItemMaster`

- GetProcess data is fetched based on the SETID and INV\_PROD\_FAM\_CD parameters specified in the Get\_Input.xml file and is written into the Get\_output.xml file. The Get\_output.xml file should display the following ItemMaster record:

```
<?xml version="1.0" encoding="UTF-8"?>
<MapperOutput>
  <PROPERTIES
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:ae="http://www.tibco.com/xmlns/ae2xsd/2002/05"
    xmlns:ns="http://www.tibco.com/xmlns/ae2xsd/2002/05/ae/
      PeopleSoft/rpc">
    <SETID>CRM01</SETID>
    <INV_PROD_FAM_CD>COMPACTOR</INV_PROD_FAM_CD>
    <TIB_INV_PROD_FAMILIES_caret_INV_ITEM_FAM>
      <item>
        <OPRN_CODE>U</OPRN_CODE>
        <EFFDT>01/01/2000</EFFDT>
        <EFF_STATUS>A</EFF_STATUS>
        <DESCR>With Compactor 11</DESCR>
        <DESCRSHORT>Compactor</DESCRSHORT>
      </item>
    </TIB_INV_PROD_FAMILIES_caret_INV_ITEM_FAM>
  </PROPERTIES>
</MapperOutput>
```

- UpdateProcess data is updated based on the SETID and INV\_PROD\_FAM\_CD parameters specified in the Update\_input.xml file and is updated in PeopleSoft.
- SyncProcess output is stored in the Sync\_output.xml file. The Sync\_output.xml file must look similar to the following:

```
<?xml version="1.0" encoding=""?>
<ns0:TIB_INV_PROD_FAMILIES_PROPERTIES
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ae="http://www.tibco.com/xmlns/ae2xsd/2002/05"
  xmlns:ns="http://www.tibco.com/xmlns/ae2xsd/2002/05/ae/
    PeopleSoft/pub"
  xmlns:ns0="http://www.tibco.com/xmlns/ae2xsd/2002/05/ae/
    PeopleSoft/pub/businessObjects">
  <SETID>CRM01</SETID>
  <INV_PROD_FAM_CD>COMPACTOR</INV_PROD_FAM_CD>
  <TIB_INV_PROD_FAMILIES_caret_INV_ITEM_FAM>
    <item>
      <OPRN_CODE>U</OPRN_CODE>
      <EFFDT>2000-01-01</EFFDT>
      <EFF_STATUS>A</EFF_STATUS>
```

```
        <DESCR>With Compactor 12</DESCR>
        <DESCRSHORT>Compactor</DESCRSHORT>
    </item>
</TIB__INV__PROD__FAMILIES_caret__INV__ITEM__FAM>
</ns0:TIB__INV__PROD__FAMILIES__PROPERTIES>
```



## Chapter 7

# **TIBCO ActiveMatrix BusinessWorks: Working with the UOM Message**

This example shows how to use the Message Publication service within a TIBCO ActiveMatrix BusinessWorks process to trace changes in the PeopleSoft environment and publish them to the TIBCO environment. The message is transferred to the PeopleSoft environment using Integration Broker.

## Topics

---

- [Example Description, page 48](#)
- [Setting up the Example, page 49](#)
- [Preparing PeopleSoft Integration Broker, page 50](#)
- [Testing the Example, page 53](#)
- [Expected Results, page 54](#)

## Example Description

---

This example shows how the adapter's Message Publication Service can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has one process, `ProcessMessagePublication`, that publishes the `UOM_SYNC` message whenever `UnitOfMeasure` details are modified or updated in the PeopleSoft application.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `Contact.zip` file. The file is located in the folder `<adapter_home>/examples/IntegrationBroker/UOM`.
4. Substitute the values of the following global variables for design-time connection:

```
adpsft8.connection.appserver  
adpsft8.connection.login  
adpsft8.connection.password  
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in the project file will be overridden.

5. Go to the **Message Publication** service instance.
6. Go to **MessagePublisher** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **UOM** in a new folder.
8. Exit TIBCO Designer.

### Start the TIBCO EMS Server:

This example uses TIBCO Enterprise Message Server (EMS) as the transport type. Ensure that the EMS server is running and accessible to the machine on which the adapter is installed before running any of the processes.

# Preparing PeopleSoft Integration Broker

After setting up the example, there are a few settings that you need to take care of in the PeopleSoft environment.

## Integration Scenario Between the Adapter and PeopleSoft Integration Broker

Use the following steps to create an integration scenario between the adapter and PeopleSoft Integration Broker:

- 1. Define a Gateway File, Message, Message Channel, and Node as listed in the procedure, *Setting up the Integration Broker in Chapter 2 Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*.
- 2. In the **Connectors** tab, Select **ConnectorID** to be TIBCOTarget.
- 3. Enter the following Connection Properties:

**JMSQueue:** TEST\_PSFT\_PUB\_UOM\_SYNC

**JMSProvider:** <JMS\_Server\_Name> : <Port\_Number> Enter the URL of the TIBCO Enterprise Message Server (EMS) and the port number. The default port number is 7222.

**Service Type:** Publisher

- 4. Define a Transaction as listed in the procedure, *Defining a Transaction in Chapter 2 Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*. Modify the following properties:

**Transaction Type:** Outbound Asynchronous.

Node Transactions

Find an Existing Value

Add a New Value

Node Name:

MSG\_PUB\_NODE

Effective Date:

12/04/2006

Transaction Type:

Outbound Asynchronous

Request Message:

UOM\_SYNC

Request Message Version:

VERSION1

Add

5. Perform the following steps to include People Code in the `SavePostChange` method.
  - a. Open PeopleSoft Application Designer.
  - b. Go to **File>Open** and in `Definition Type` choose `Component`.
  - c. Open the definition of the component you want to publish data from.
  - d. With the component definition open, go to **View>View PeopleCode**.
  - e. When the PeopleCode Editor comes up, leave the component name as is in the left hand side drop-down and select the `SavePostChange` event in the right hand side drop-down.
  - f. In the PeopleCode Editor, paste the PeopleCode given below. In case there is any existing PeopleCode, append the code given below.

Replace the `%message_name%` with the corresponding message name without any quotes.

#### **For PeopleTools 8.1x**

(This PeopleCode will go into the `SavePostChange` event of the PeopleCode for the required Component.)

```
Local Message &msg;
Local Rowset &rowSet;
&rowSet = GetLevel0();
&msg = CreateMessage(Message.UOM_SYNC);
&msg.CopyRowset(&rowSet);
&msg.Publish();
```

#### **For PeopleTools 8.4x**

(This PeopleCode will go into the `SavePostChange` event of the PeopleCode for the required Component.)

```
Local Message &msg;
Local Message &msg1;
Local Rowset &rowSet;
Local XmlDoc &xmlDoc;
Local string &xmlString;
Local XmlNode &rootNode;
Local XmlNode &actionNode;
&rowSet = GetLevel0();
&msg = CreateMessage(Message.UOM_SYNC);
&msg1 = CreateMessage(Message.UOM_SYNC);
&msg.CopyRowset(&rowSet);
&xmlString = &msg.GenXMLString();
```

```

&xmlDoc = CreateXmlDoc(&xmlString);
&rootNode = &xmlDoc.DocumentElement;
&actionNode =
&rootNode.FindNode("MsgData/Transaction/PSCAMA/AUDIT_ACTN");
&actionNode.NodeValue = %Mode;
&xmlString = &xmlDoc.GenXmlString();
&msg1.LoadXMLString(&xmlString);
&msg1.Publish();

```

6. Save the component definition.

7. Update PeopleSoft.

Open the UOM component in the browser by navigating to **Setup CRM>Common Definitions>UnitsOfMeasure**. Open an existing UOM definition, update and save.

8. Monitor the Message in PeopleSoft Message Monitor.

Go to the Message Monitor by navigating to **PeopleTools>Integration Broker>Monitor>Monitor Message**. Choose the **Publication Contract** tab and click **Refresh**. The status of the latest message published will be displayed. The published message will also be displayed on the adapter console.



If you use PeopleTools 8.1x, the status of the latest message published will be displayed as **Retry**.

9. Stop the adapter.

You can stop the adapter by broadcasting a stop message using the `tibrvsend` command-line tool available from `TIBCO_HOME\TIBRV\bin`. The convention of the subject is `<hostname><instanceName>.STOPAdapter` with message content as now. Also if you have TIBCO Hawk installed on your machine, you can stop the adapter using the `StopAdapterInstance` method of the TIBCO Hawk microagent.

## Testing the Example

---

In TIBCO Designer:

1. From the project panel, select the process you want to test. For example, `ProcessMessagePubliser`.
2. Click the **Set Breakpoints** icon.
3. In the window that appears, choose **Select All**, then click **OK**.
4. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
5. Click the **Start testing viewed process** button.
6. In the process selection window that appears, the `ProcessMessagePubliser` is selected by default. Click **Load Selected**. The process is now in Test mode.
7. Click the **Adapter Publisher** icon.
8. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
9. Any change to the data in PeopleSoft will trigger the generation of an `xml` file in the adapter.
10. Step through the process once more.
11. Click the **Stop Testing** icon to return to design mode.

The above steps describe the testing for `ProcessMessagePublication`.

## Expected Results

---

The example results can be viewed in the respective XML files written to:  
<adapter\_home>/examples/IntegrationBroker/UOM

- ProcessMessagePubliser looks for changes on UOM\_SYNC\_VERSION1 message. Based on the UOM message, the process generates an xml file that is published to the TIBCO environment. The xml file should display the following UOM record:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <ns0:uoms
  xmlns:ns0="http://xmlns.example.com/unique/default/namespace/
  1158039121783">
  - <ns0:uom>
    <ns0:unit_of_measure>AVC    </ns0:unit_of_measure>
    <ns0:Desc>Average Cost Dollars    </ns0:Desc>
    <ns0:DescShort>AvgCost    </ns0:DescShort>
  </ns0:uom>
</ns0:uoms>
```





## Chapter 8

## TIBCO ActiveMatrix BusinessWorks: Working with the Contact Message

This example shows how to use the Message Subscription service within a TIBCO ActiveMatrix BusinessWorks process to get a request from the TIBCO environment, process it, and send it to PeopleSoft. The message is transferred to the PeopleSoft environment using Integration Broker.

### Topics

---

- [Example Description, page 57](#)
- [Setting up the Example, page 58](#)
- [Preparing PeopleSoft Integration Broker, page 59](#)
- [Testing the Example, page 63](#)
- [Expected Results, page 64](#)

## Example Description

---

This example shows how the adapter's Message Subscription Service can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has one process, `ProcessMessageSubscriber`, that uses the adapter's Message Subscription service to subscribe for PeopleSoft messages from the TIBCO environment and send them to the PeopleSoft environment.

`ProcessMessageSubscriber` subscribes for PeopleSoft messages from the TIBCO environment, maps them to PeopleSoft schemas, and inserts data into the PeopleSoft database.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box, click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `Contact.zip` file. The file is located in the folder  
`<adapter_home>/examples/IntegrationBroker/Contact`
4. Substitute the values of the following global variables for design-time connection:
 

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.
5. Go to the **Message Subscription** service instance.
6. Go to **MessageSubscriber** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **Contact** in a new folder.
8. Exit TIBCO Designer.

### Start the TIBCO EMS Server:

This example uses TIBCO Enterprise Message Server (EMS) as the transport type. Ensure that the EMS server is running and accessible to the machine on which the adapter is installed before running any of the processes.

## Preparing PeopleSoft Integration Broker

---

After setting up the example, there are a few settings that you need to take care of in the PeopleSoft environment.

## Integration Scenario Between the Adapter and PeopleSoft Integration Broker

1. Modify the `integrationGateway.Properties` file located in `PeopleSoftHOME\PT8.46\webserve\peoplesoft\applications\peoplesoft\PSIGW\WEB-INF`.

Replace the `%user_name%` and `%server_name:port_number%` with the corresponding user name without any quotes.

Search for the string :

`## JOLT connect string setting for optional Default Application Server. Do NOT specify a NODENAME.`

Add the following lines below the string:

```
ig.isc.serverURL=%server_name:port_number%. For example,
//10.97.97.222:9000.
```

```
ig.isc.userid=%user_name%. For example User01.
```

`#Use the supplied "Password Encryption Utility" to generate an encrypted password for the next entry.`

```
ig.isc.password=Ez6NDsqOkxI=
```

```
ig.isc.toolsRel=8.46
```

At the end of the file, add the following lines:

```
ig.jms.Queue1=TEST_PSFT_SUB_CONTACT
```

```
ig.jms.Queue1.Provider=Tibco
```

```
ig.jms.Queue1.JMSFactory=QueueConnectionFactory
```

```
ig.jms.Queue1.MessageSelector=
```

```
ig.jms.Queue1.Url=%server_name:port_number%. For example,
tcp://localhost:7222.
```

```
ig.jms.Queue1.User=%user_name%
```

```
ig.jms.Queue1.Password=kzqhmUL2ras=
```



Ensure that `ig.isc.serverURL` has the same value as the one specified in the JNDI URL field in the **SubscriberOptions** tab in the adapter. Also ensure that the subject name contained in `ig.jms.Queue1` is the same as the one entered in the IB Subscriber Subject field.

To generate the password, run the utility program `PSCipher` located in `PeopleSoftHOME\PT8.46\webserve\peoplesoft`.

2. Define a Gateway File, Message, Message Channel, and Node as listed in the procedure, *Setting up the Integration Gateway in Chapter 2 Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*.
3. In the **Connectors** tab, select **ConnectorID** to be `JMSTarget`.

4. Enter the following Connection Properties:

**JMSFactory:** QueueConnectionFactory

**JMSProvider:** TIBCO

**JMSURL:** Enter the name of the TIBCO Enterprise Message Server (EMS).

5. Define a Transaction as listed in the procedure, *Defining a Transaction* in Chapter 2 *Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*. Modify the following properties:

- a. **Transaction Type:** Inbound Asynchronous.

**Node Transactions**

Find an Existing Value Add a New Value

Node Name: MSG\_SUB\_NODE

Effective Date: 12/04/2006

Transaction Type: Inbound Asynchronous

Request Message: CONTACT\_SYNC

Request Message Version: VERSION\_1

Add

6. In the PeopleCode Editor, paste the PeopleCode given below. In case there is any existing PeopleCode, append the code given below.

```
Local Message &MSG;
Local string &XML_STRING;
Local File &SUCCESS_FILE;
Local XmlDoc &xmlDoc;
&MSG = GetMessage();
/* get an xmldoc object loaded with the content data.
*/
&XML_STRING = &MSG.GenXMLString();
&SUCCESS_FILE = GetFile("c:\temp\async_sub_country.txt", "w",
"a", %FilePath_Absolute);
&SUCCESS_FILE.WriteLine(&XML_STRING);
&SUCCESS_FILE.Close();
```

7. Start the JMSListeningConnector Administrator. To start the JMSListeningConnector Administrator, access [http://<PeopleSoft\\_Application\\_Server\\_Name>:<Port\\_Number>/PSGIW/JMSListen](http://<PeopleSoft_Application_Server_Name>:<Port_Number>/PSGIW/JMSListen)

`ingConnector Administrator`. Ensure that the Listener is started before running the Message Subscription Service.

You can stop the adapter by broadcasting a stop message using the `tibrvsend` command-line tool available from `TIBCO_HOME\TIBRV\bin`. The convention of the subject is `<hostname><instanceName>.STOPAdapter` with message content as `now`. Also if you have TIBCO Hawk installed on your machine, you can stop the adapter using the `StopAdapterInstance` method of the TIBCO Hawk microagent.



## Testing the Example

---

In TIBCO Designer:

1. From the project panel, select the process you want to test. For example, `ProcessMessageSubscriber`.
2. Click the **Set Breakpoints** icon.
3. In the window that appears, choose **Select All**, then click **OK**.
4. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
5. Click the **Start testing viewed process** button.
6. In the process selection window that appears, the `ProcessMessageSubscriber` is selected by default. Click **Load Selected**. The process is now in Test mode.
7. Click the **File Poller** icon. In the configuration panel, `File Name` field, click **Browse** and select the input XML file. Click **Apply**. Any change to the input XML file starts the process.
8. Once the process starter is highlighted (indicating a process has started), click the **Step to next activity** icon to step through the process.
9. Step through the process once more.
10. Click the **Stop Testing** icon to return to design mode.

The above steps describe the testing for `ProcessMessageSubscriber`.

## Expected Results

---

The example results can be viewed in the respective XML files written to:  
`<adapter_home>/examples/IntegrationBroker/Contact`

- ProcessMessageSubscriber data is subscribes for changes on CONTACT\_SYNC\_VERSION1 message. Based on the Contact message, the process generates an xml file that is published to the PeopleSoft environment. The PeopleSoft environment processes the message and generates output xml file. The xml file should display the following Contact record:

```
<?xml version="1.0" ?>
- <CONTACT_SYNC>
- <FieldTypes>
- <CONTACT class="R">
  <SETID type="CHAR" />
  <CONTACT_ID type="CHAR" />
  <EFF_STATUS type="CHAR" />
  <CONTACT_FLAG type="CHAR" />
  <NAME1 type="CHAR" />
  <TITLE type="CHAR" />
  <LANGUAGE_CD type="CHAR" />
  <EMAILID type="CHAR" />
  <COMM_METHOD type="CHAR" />
  <SALUTATION_CD type="CHAR" />
  <SALUTATION type="CHAR" />
  <CREATE_CUSTOMER type="CHAR" />
  <LAST_MAINT_OPRID type="CHAR" />
  <DATE_LAST_MAINT type="DATE" />
  <AUTHORIZATION_ID type="CHAR" />
  <PERSON_ID type="CHAR" />
</CONTACT>
- <CONTACT_CUST class="R">
  <SETID type="CHAR" />
  <CONTACT_ID type="CHAR" />
  <CUSTOMER_SETID type="CHAR" />
  <CUST_ID type="CHAR" />
  <EFF_STATUS type="CHAR" />
  <CNTCT_SEQ_NUM type="NUMBER" />
  <ADDRESS_SEQ_NUM type="NUMBER" />
  <BILL_TO_FLG type="CHAR" />
  <SOLD_TO_FLG type="CHAR" />
  <SHIP_TO_FLG type="CHAR" />
  <LAST_MAINT_OPRID type="CHAR" />
  <DATE_LAST_MAINT type="DATE" />
</CONTACT_CUST>
- <CONTACT_PHN class="R">
  <SETID type="CHAR" />
  <CONTACT_ID type="CHAR" />
  <PHONE_TYPE type="CHAR" />
  <COUNTRY_CODE type="CHAR" />
  <PHONE type="CHAR" />
  <EXTENSION type="CHAR" />
```

```

        <LAST_MAINT_OPRID type="CHAR" />
        <DATE_LAST_MAINT type="DATE" />
    </CONTACT_PHN>
- <CONTACT_PAGER class="R">
    <SETID type="CHAR" />
    <CONTACT_ID type="CHAR" />
    <PHONE_TYPE type="CHAR" />
    <COUNTRY_CODE type="CHAR" />
    <PHONE type="CHAR" />
    <EXTENSION type="CHAR" />
    <PASSWORD type="CHAR" />
    <LAST_MAINT_OPRID type="CHAR" />
    <DATE_LAST_MAINT type="DATE" />
</CONTACT_PAGER>
- <CONTACT_CARD class="R">
    <SETID type="CHAR" />
    <CONTACT_ID type="CHAR" />
    <CUSTOMER_SETID type="CHAR" />
    <CUST_ID type="CHAR" />
    <CR_CARD_NBR type="CHAR" />
    <CR_CARD_TYPE type="CHAR" />
    <PRIMARY_CARD type="CHAR" />
    <CR_CARD_FNAME type="CHAR" />
    <CR_CARD_LNAME type="CHAR" />
    <CR_CARD_EXPMO type="CHAR" />
    <CR_CARD_EXPYR type="CHAR" />
    <CR_CARD_DIGITS type="CHAR" />
    <ADDR_SEQ_NUM type="NUMBER" />
    <LAST_MAINT_OPRID type="CHAR" />
    <DATE_LAST_MAINT type="DATE" />
</CONTACT_CARD>
- <PSCAMA class="R">
    <LANGUAGE_CD type="CHAR" />
    <AUDIT_ACTN type="CHAR" />
    <BASE_LANGUAGE_CD type="CHAR" />
    <MSG_SEQ_FLG type="CHAR" />
    <PROCESS_INSTANCE type="NUMBER" />
    <PUBLISH_RULE_ID type="CHAR" />
    <MSGNODENAME type="CHAR" />
</PSCAMA>
</FieldTypes>
- <MsgData>
- <Transaction>
- <CONTACT class="R">
    <SETID>site id </SETID>
    <CONTACT_ID>cx </CONTACT_ID>
    <EFF_STATUS>cx </EFF_STATUS>
    <CONTACT_FLAG>cx </CONTACT_FLAG>
    <NAME1>cx </NAME1>
    <TITLE>cx </TITLE>
    <LANGUAGE_CD>cx </LANGUAGE_CD>
    <EMAILID>cx </EMAILID>
    <COMM_METHOD>cx </COMM_METHOD>
    <SALUTATION_CD>cx </SALUTATION_CD>
    <SALUTATION>cx </SALUTATION>
    <CREATE_CUSTOMER>cx </CREATE_CUSTOMER>
    <LAST_MAINT_OPRID />

```

```

        <DATE_LAST_MAINT />
        <AUTHORIZATION_ID>sa</AUTHORIZATION_ID>
        <PERSON_ID>ds</PERSON_ID>
    </CONTACT>
- <CONTACT_CUST class="R">
    <SETID />
    <CONTACT_ID />
    <CUSTOMER_SETID />
    <CUST_ID />
    <EFF_STATUS />
    <CNTCT_SEQ_NUM />
    <ADDRESS_SEQ_NUM />
    <BILL_TO_FLG />
    <SOLD_TO_FLG />
    <SHIP_TO_FLG />
    <LAST_MAINT_OPRID />
    <DATE_LAST_MAINT />
</CONTACT_CUST>
- <CONTACT_PHN class="R">
    <SETID />
    <CONTACT_ID />
    <PHONE_TYPE />
    <COUNTRY_CODE />
    <PHONE />
    <EXTENSION />
    <LAST_MAINT_OPRID />
    <DATE_LAST_MAINT />
</CONTACT_PHN>
- <CONTACT_PAGER class="R">
    <SETID />
    <CONTACT_ID />
    <PHONE_TYPE />
    <COUNTRY_CODE />
    <PHONE />
    <EXTENSION />
    <PASSWORD />
    <LAST_MAINT_OPRID />
    <DATE_LAST_MAINT />
</CONTACT_PAGER>
- <CONTACT_CARD class="R">
    <SETID />
    <CONTACT_ID />
    <CUSTOMER_SETID />
    <CUST_ID />
    <CR_CARD_NBR />
    <CR_CARD_TYPE />
    <PRIMARY_CARD />
    <CR_CARD_FNAME />
    <CR_CARD_LNAME />
    <CR_CARD_EXPMO />
    <CR_CARD_EXPYR />
    <CR_CARD_DIGITS />
    <ADDR_SEQ_NUM />
    <LAST_MAINT_OPRID />
    <DATE_LAST_MAINT />
</CONTACT_CARD>
- <PSCAMA class="R">

```

```
<LANGUAGE_CD>ENG</LANGUAGE_CD>
<AUDIT_ACTN />
<BASE_LANGUAGE_CD>ENG</BASE_LANGUAGE_CD>
<MSG_SEQ_FLG />
<PROCESS_INSTANCE>0</PROCESS_INSTANCE>
<PUBLISH_RULE_ID />
<MSGNODENAME />
</PSCAMA>
</Transaction>
</MsgData>
</CONTACT_SYNC>
```

## Chapter 9

# TIBCO ActiveMatrix BusinessWorks: Working with the Country Message

This example shows how to use the Request-Response Invocation service within a TIBCO ActiveMatrix BusinessWorks process to get a request from PeopleSoft, process it, and respond back with a message. The message is transferred to the PeopleSoft environment using Integration Broker.

## Topics

---

- [Example Description, page 69](#)
- [Setting up the Example, page 70](#)
- [Preparing PeopleSoft Integration Broker, page 71](#)
- [Testing the Example, page 75](#)
- [Expected Results, page 76](#)

## Example Description

---

This example shows how the adapter's Request-Response Invocation service can be used in a TIBCO ActiveMatrix BusinessWorks process. The example has one process, `ProcessMessageRPCClient`, that uses the adapter's Request-Response Invocation service to fetch data from PeopleSoft for key values that you specify in an input XML file.

`ProcessMessageRPCClient` queries PeopleSoft based on the given key values and fetches the data. The data is sent back by the adapter to TIBCO ActiveMatrix BusinessWorks and is written into an output XML file.

In a typical business scenario, an adapter to another application (for example, SAP or Siebel) publishes a message into the TIBCO environment. TIBCO ActiveMatrix BusinessWorks receives the messages and writes them to an input XML file. The content of this XML file will contain the changes to be updated in PeopleSoft. TIBCO ActiveMatrix Adapter for PeopleSoft uses the Request-Response Invocation service to update the data in PeopleSoft and sends a reply to TIBCO ActiveMatrix BusinessWorks.

In this case, you may specify key values in the input XML file.

`ProcessMessageRPCClient` queries PeopleSoft based on the key values. The COUNTRY details are updated in PeopleSoft. On receiving the modified data, TIBCO ActiveMatrix BusinessWorks logs it to an XML file.

## Setting up the Example

---

Before starting the example you must import the sample project zip file and save it in a new project.

In TIBCO Designer:

1. In the initial dialog box click **Open New Project** and specify a name for the project.
2. Click **Project>Import Full Project**.
3. Click the **Zip Archive** tab and browse to locate the `Country.zip` file. The file is located in the folder  
`<adapter_home>/examples/IntegrationBroker/Country`
4. Substitute the values of the following global variables for design-time connection:
 

```
adpsft8.connection.appserver
adpsft8.connection.login
adpsft8.connection.password
adpsft8.connection.port
```

Or, you can specify these values in the `examples.tra`. In this case, the values given in project file will be overridden.
5. Go to the **Request-Response Invocation** service instance.
6. Go to **MessageRPCClient** and connect to the design-time adapter. Click **Save To Peoplesoft**.
7. Save as a multi-file project by the name **Country** in a new folder.
8. Exit TIBCO Designer.

### Start the TIBCO EMS Server:

This example uses TIBCO Enterprise Message Server (EMS) as the transport type. Ensure that the EMS server is running and accessible to the machine on which the adapter is installed before running any of the processes.



## Preparing PeopleSoft Integration Broker

---

After setting up the example, there are a few settings that you need to take care of in the PeopleSoft environment.

### Integration Scenario Between the Adapter and PeopleSoft Integration Broker

1. Define a Gateway File, Message, Message Channel, and Node as listed in the procedure, *Setting up the Integration Gateway in Chapter 2 Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*.
2. In the **Connectors** tab, select **ConnectorID** to be **TIBCOTarget**, if you have selected JMS transport while configuring the adapter.
3. Enter the following Connection Properties:

**JMSQueue:** TEST.CLIENT

**JMSProvider:** Enter the URL of the TIBCO Enterprise Message Server (EMS)

**Service Type:** RPCCLIENT

- 4. Define a Transaction as listed in the procedure, in *Defining a Transaction, Chapter 2 Preparing PeopleSoft of TIBCO ActiveMatrix Adapter for PeopleSoft Configuration and Deployment*. Modify the following properties:
  - a. **Transaction Type:** Outbound Synchronous.

Node Transactions

Find an Existing Value

Add a New Value

Node Name:

RPCCLIENT\_JMS

Effective Date:

12/04/2006

Transaction Type:

Outbound Synchronous

Request Message:

COUNTRY\_SYNC

Request Message Version:

VERSION\_1

Add

Find an Existing Value

Add a New Value

- b. Click **Messages**. In the Response Message group, enter the following:  
**Message Name:** COUNTRY\_SYNC  
**Message Version:** VERSION\_1

5. Perform the following steps to include People Code in the `SavePostChange` method.
  - a. Open PeopleSoft Application Designer
  - b. Go to **File>Open** and in `Definition Type` choose `Component`.
  - c. Open the definition of the component you want to publish data from.
  - d. With the component definition open, go to **View>View PeopleCode**.
  - e. When the PeopleCode Editor comes up, leave the component name as is in the left hand side drop-down and select the `SavePostChange` event in the right hand side drop-down.
  - f. In the PeopleCode Editor, paste the PeopleCode given below. In case there is any existing PeopleCode, append the code given below.

Replace the `%message_name%` with the corresponding message name without any quotes.

#### For PeopleTools 8.4x

(This PeopleCode will go into the `SavePostChange` event of the PeopleCode for the required Component)

```
Local string &XML_STRING;
Local Message &request_MSG, &response_MSG;
Local Rowset &request_RS;
Local Rowset &response_RS;
Local Rowset &MSG_RS;
Local Record &REC;
&request_RS = GetLevel0();
&request_MSG = CreateMessage(Message.COUNTRY_SYNC);
&request_MSG.CopyRowset(&request_RS);
/* publish request and wait for the response = &XML_STRING; */
&response_MSG = &request_MSG.SyncRequest();
If (&response_MSG.ResponseStatus = 0) Then
    &MSG_RS = &response_MSG.GetRowset();
    &REC =
    &response_MSG.GetRowset().GetRow(1).GetRecord(Record.COUNTRY
    );
    &XML_STRING = &response_MSG.GenXMLString();
End-If
```

Save the component definition.

6. Start the adapter.

Run the Request-Response Invocation instance from the TIBCO Designer.

7. Update PeopleSoft. Access the PeopleSoft Application and navigate to **Setup CRM>Common Definitions>Location>Country TBL**. Open an existing Country definition, update and save. You can see the generated output as an XML file on the PeopleSoft machine.
8. Stop the adapter.

You can stop the adapter by broadcasting a stop message using the `tibrvsend` command-line tool available from `TIBCO_HOME\TIBRV\bin`. The convention of the subject is `<hostname><instanceName>.STOPAdapter` with message content as `now`. Also if you have TIBCO Hawk installed on your machine, you can stop the adapter using the `StopAdapterInstance` method of the TIBCO Hawk microagent.

## Testing the Example

---

In TIBCO Designer:

1. From the project panel, select the process you want to test. For example, `ProcessMessageRPCClient`.
2. Click the **Set Breakpoints** icon.
3. In the window that appears, choose **Select All**, then click **OK**.
4. Click the **Tester** tab to the left of the project panel. The test panel replaces the project tree.
5. Click the **Start testing viewed process** button.
6. In the process selection window that appears, the `ProcessMessageRPCClient` is selected by default. Click **Load Selected**. The process is now in Test mode.
7. Click the **Adapter Request-Response Server** icon.
8. Click the **Step to next activity** icon to step through the process.
9. Step through the process once more.
10. Click the **Stop Testing** icon to return to design mode.

The above steps describe the testing for `ProcessMessageRPCClient`.

## Expected Results

---

The example results can be viewed in the respective XML files written to:

`<adapter_home>/examples/IntegrationBroker/Country`

- `ProcessMessageRPCClient` data is fetched based on the `COUNTRY_SYNC_VERSION1` message. The adapter receives a request whenever the `COUNTRY_SYNC_VERSION1` message is modified. The adapter processes the request and sends a response back to PeopleSoft in an xml file. The xml file should display the following Country record:

```
<?xml version="1.0" ?>
- <COUNTRY_SYNC>
- <FieldTypes>
- <COUNTRY_TBL class="R">
  <ADDR1_AVAIL type="CHAR" />
  <ADDR1_LBL type="CHAR" />
  <ADDR2_AVAIL type="CHAR" />
  <ADDR2_LBL type="CHAR" />
  <ADDR3_AVAIL type="CHAR" />
  <ADDR3_LBL type="CHAR" />
  <ADDR4_AVAIL type="CHAR" />
  <ADDR4_LBL type="CHAR" />
  <ADDR_FIELD1_AVAIL type="CHAR" />
  <ADDR_FIELD1_LBL type="CHAR" />
  <ADDR_FIELD2_AVAIL type="CHAR" />
  <ADDR_FIELD2_LBL type="CHAR" />
  <ADDR_FIELD3_AVAIL type="CHAR" />
  <ADDR_FIELD3_LBL type="CHAR" />
  <CITY_AVAIL type="CHAR" />
  <CITY_LBL type="CHAR" />
  <COUNTRY type="CHAR" />
  <COUNTRY_2CHAR type="CHAR" />
  <COUNTY_AVAIL type="CHAR" />
  <COUNTY_LBL type="CHAR" />
  <DESCR type="CHAR" />
  <DESCRSHORT type="CHAR" />
  <EU_MEMBER_STATE type="CHAR" />
  <GBSYS_CFGPATH_UK type="CHAR" />
  <GBSYS_NRPATH_UK type="CHAR" />
  <GEO_CODE_AVAIL type="CHAR" />
  <GEO_CODE_LBL type="CHAR" />
  <HOUSE_TYPE_AVAIL type="CHAR" />
  <HOUSE_TYPE_LBL type="CHAR" />
  <IN_CITY_LIM_AVAIL type="CHAR" />
  <IN_CITY_LIM_LBL type="CHAR" />
  <NUM1_AVAIL type="CHAR" />
  <NUM1_LBL type="CHAR" />
  <NUM2_AVAIL type="CHAR" />
  <NUM2_LBL type="CHAR" />
  <POSTAL_AVAIL type="CHAR" />
  <POSTAL_LBL type="CHAR" />
  <POST_SRCH_AVAIL type="CHAR" />
  <STATE_AVAIL type="CHAR" />
```

```

        <STATE_LBL type="CHAR" />
        <SYNCDTTM type="CHAR" />
        <SYNCID type="NUMBER" />
    </COUNTRY_TBL>
- <PSCAMA class="R">
    <LANGUAGE_CD type="CHAR" />
    <AUDIT_ACTN type="CHAR" />
    <BASE_LANGUAGE_CD type="CHAR" />
    <MSG_SEQ_FLG type="CHAR" />
    <PROCESS_INSTANCE type="NUMBER" />
    <PUBLISH_RULE_ID type="CHAR" />
    <MSGNODENAME type="CHAR" />
</PSCAMA>
</FieldTypes>
- <MsgData>
- <Transaction>
- <COUNTRY_TBL class="R">
    <ADDR1_AVAIL />
    <ADDR1_LBL />
    <ADDR2_AVAIL />
    <ADDR2_LBL />
    <ADDR3_AVAIL />
    <ADDR3_LBL />
    <ADDR4_AVAIL />
    <ADDR4_LBL />
    <ADDR_FIELD1_AVAIL />
    <ADDR_FIELD1_LBL />
    <ADDR_FIELD2_AVAIL />
    <ADDR_FIELD2_LBL />
    <ADDR_FIELD3_AVAIL />
    <ADDR_FIELD3_LBL />
    <CITY_AVAIL />
    <CITY_LBL />
    <COUNTRY>India</COUNTRY>
    <COUNTRY_2CHAR />
    <COUNTY_AVAIL />
    <COUNTY_LBL />
    <DESCR>India</DESCR>
    <DESCRSHORT>IND</DESCRSHORT>
    <EU_MEMBER_STATE />
    <GBSYS_CFGPATH_UK />
    <GBSYS_NRPATH_UK />
    <GEO_CODE_AVAIL />
    <GEO_CODE_LBL />
    <HOUSE_TYPE_AVAIL />
    <HOUSE_TYPE_LBL />
    <IN_CITY_LIM_AVAIL />
    <IN_CITY_LIM_LBL />
    <NUM1_AVAIL />
    <NUM1_LBL />
    <NUM2_AVAIL />
    <NUM2_LBL />
    <POSTAL_AVAIL />
    <POSTAL_LBL />
    <POST_SRCH_AVAIL />
    <STATE_AVAIL />
    <STATE_LBL />

```

```

        <SYNCDTTM />
        <SYNCID />
    </COUNTRY_TBL>
- <PSCAMA class="R">
    <LANGUAGE_CD>ENG</LANGUAGE_CD>
    <AUDIT_ACTN />
    <BASE_LANGUAGE_CD>ENG</BASE_LANGUAGE_CD>
    <MSG_SEQ_FLG />
    <PROCESS_INSTANCE>0</PROCESS_INSTANCE>
    <PUBLISH_RULE_ID />
    <MSGNODENAME />
</PSCAMA>
</Transaction>
</MsgData>
</COUNTRY_SYNC>

```



# Index

## C

commands  
    TIB\_PS8\_ADAPTER [6](#)  
conventions used in this manual [xiii](#)  
customer support [xvi](#)

## D

Deploy and Run the Example [14](#)  
documents, related [xiii](#)

## E

Example Description [12, 18, 24, 30, 40, 48, 57, 69](#)  
Expected Results [16, 21, 27, 36, 45, 54, 64, 76](#)

## G

Get [2](#)

## I

Integration Scenario Between the Adapter and PeopleSoft Integration Broker [50, 60](#)

## L

Location [9](#)

## O

Overview [2](#)

## P

Prerequisites [5](#)

## R

related documents [xiii](#)  
RV [xvi, xvi](#)

## S

Setup the Example [13, 19, 25, 39, 41, 49, 58, 70](#)  
support, contacting [xvi](#)  
Sync [4](#)

## T

technical support [xvi](#)  
Test the Example [20, 26, 33, 42, 53, 63, 75](#)  
TIB\_PS8\_ADAPTER PeopleSoft project [5](#)  
TIBCO\_HOME [xiv](#)

## U

Update [2](#)

Update Operation Using a Publish-Subscribe  
Scenario [3](#)  
Update Operation Using an Request-Response  
Scenario [3](#)  
Using TIBCO ActiveMatrix BusinessWorks [7](#)

## W

Working with dat files in TIBCO Designer [7](#)