TIBCO ActiveMatrix BusinessWorks™ Getting Started

Software Release 6.1.0
May 2014
Document Update: June 2014

Two-Second Advantage®

Important Information

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

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

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

TIBCO, TIBCO ActiveMatrix BusinessWorks, TIBCO Rendezvous, TIBCO Enterprise Message Service, TIBCO Business Studio, TIBCO Enterprise Administrator, TIBCO ActiveSpaces, TIBCO Runtime Agent, TIBCO Designer, and Two-Second Advantage are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Enterprise Java Beans (EJB), Java Platform Enterprise Edition (Java EE), Java 2 Platform Enterprise Edition (J2EE), and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle Corporation 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 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 © 2001-2014 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Figures 4
TIBCO Documentation and Support Services 5
Changing Help Preferences 6
About the Getting Started Guide 7
TIBCO ActiveMatrix BusinessWorks Orientation 8
TIBCO Business Studio 8
Debugger 8
Process Development 8
Beginning Process Development 8
Developing Web Services 9
Process Development Features 10
Shared Resources and Mapping 11
REST Support 11
REST Documenter and Tester 12
Building an Archive File 13
File Poller Module Tutorial 16
Creating a New Project 16
Configuring the File Poller Project 16
Testing the FilePoller Application in the Debugger 18
Building and Testing a REST Service 20
Creating a New BusinessWorks Process 21
Building a REST Service 24
Testing the REST Service 27
Testing the POST and GET Operation 31
Troubleshooting 32
Enterprise Administration 34
Running the Admin Sample 34
Deploying the Application 37
Defining and Deploying Multiple AppSpaces and AppNodes 40
Core Admin Sample Scripts 41

Figures

Utils Package 35
Net Package 35

TIBCO Documentation and Support Services

All TIBCO documentation is available in the TIBCO Documentation Library, which can be found here:

https://docs.tibco.com

Product-Specific Documentation

The following documents for TIBCO ActiveMatrix BusinessWorks can be found in the TIBCO Documentation Library:

- Concepts
- Installation
- Getting Started
- Application Development
- Administration
- Bindings and Palettes Reference
- Samples
- Migration

The following documents provide additional information and can found be in the TIBCO Documentation Library:

- *TIBCO Enterprise Administrator User's Guide*

How to Contact TIBCO Support

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

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

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

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

  https://support.tibco.com

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

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCOmmunity offers forums, blogs, and access to a variety of resources. To register, go to:

http://www.tibcommunity.com

Changing Help Preferences

By default, TIBCO ActiveMatrix BusinessWorks documentation access from TIBCO Business Studio is through **https://docs.tibco.com/**. However, if you do not have an all time Internet access, you can download the full TIBCO ActiveMatrix BusinessWorks 6.x documentation and configure TIBCO Business Studio to this local copy.

TIBCO ActiveMatrix BusinessWorks documentation can be accessed in the following ways:

- **Online mode**: The online documentation is available on the TIBCO Product Documentation site at https://docs.tibco.com/products/tibco-activematrix-businessworks. The benefit of using the online mode is that the Doc site contains the latest version of the documentation. Check the Doc site frequently for updated documentation.
- **Offline mode**: For environments that do not have Internet access, the product documentation can be downloaded from the TIBCO Product Documentation site using the **Download All** button, or from https://download.tibco.com/tibco/, and made available offline either on a local or a network drive.

By default, TIBCO Business Studio accesses the documentation from the TIBCO Product Documentation site. To access the product documentation using the offline mode, change the help preferences in TIBCO Business Studio to access help from a local directory or from an internal web server.

Prerequisites

Before changing help preferences you must first download TIBCO ActiveMatrix BusinessWorks 6.x documentation from https://docs.tibco.com/.

1. Click https://docs.tibco.com/.
2. In the search field, type `ActiveMatrix BusinessWorks` and press return.
3. Select the TIBCO ActiveMatrix BusinessWorks 6.x product from the list.
4. Scroll to the bottom of the page and click the **Download All** button.
5. A zip file containing the latest documentation downloads to your web browser's default download location. Copy the zip file to a local directory, or to an internal web server and then unzip the file.

Procedure

1. In TIBCO Business Studio, click **Window > Preferences**. On Mac OS X, click **TIBCO Business Studio > Preferences**.
2. In the **Preferences** dialog, click **BusinessWorks > Help**.
3. Click **Custom Location** and click **Browse** to select the `html` directory where you unzipped the documentation, or provide the URL to the `html` directory on your internal web server.
4. Click **Apply** and then click **OK**.

About the Getting Started Guide

This Getting Started guide includes a guided tour of creating a process, testing it in the Debugger, and deploying this developed TIBCO ActiveMatrix BusinessWorks application using the TIBCO®® Enterprise Administrator. The tutorials or samples are fairly simple so that you focus on how to use TIBCO ActiveMatrix BusinessWorks within each phase of the project lifecycle.

This guide is not intended to illustrate every aspect of TIBCO ActiveMatrix BusinessWorks, so only a small subset of the available activities will be used.

This guide will familiarize you with the following:

- TIBCO ActiveMatrix BusinessWorks Orientation: aims to familiarize you with the BusinessWorks development and deployment environment. Before proceeding, it is recommended to read the *Concepts* guide to familiarize yourself with TIBCO ActiveMatrix BusinessWorks terminology and basic concepts.
- File Poller Module Tutorial: guides you through creating a simple process and running it.
- Building and Testing a REST Service: familiarizes you to build and test a simple REST Service using TIBCO Business Studio™ and the Swagger UI.
- Enterprise Administration: gives information about the Enterprise Administration framework, the user can deploy their application using **`bwadmin`** command line utility or through Admin Web UI.

TIBCO ActiveMatrix BusinessWorks Orientation

TIBCO ActiveMatrix BusinessWorks 6.x includes an Eclipse-based design time. You can create new services, orchestrate business processes and integrate applications in a shortest time. TIBCO ActiveMatrix BusinessWorks 6.x offers a model-driven development approach to process design, supported by a rich set of palettes that allows you to visually create and test business processes that connect to various technologies such as database, messaging servers, and so on.

One or more processes and their associated resources are then packaged into enterprise archive files. These archive files are then deployed in BWAdmin or TIBCO Enterprise Administrator, which is used to administer these applications.

TIBCO Business Studio

TIBCO Business Studio is the design-time IDE (based on Eclipse) where you can user can create and test ActiveMatrix BusinessWorks processes.

Debugger

TIBCO ActiveMatrix BusinessWorks offers a Debugger which can be used from TIBCO Business Studio. You can use this to test processes during the process development stage. Starting the **Debugger** brings up the **Debug** perspective. Use this perspective to set breakpoints, steps through processes, and examine job variables and activity input and output at each step. The Console view also displays the messages and errors returned by ActiveMatrix BusinessWorks.

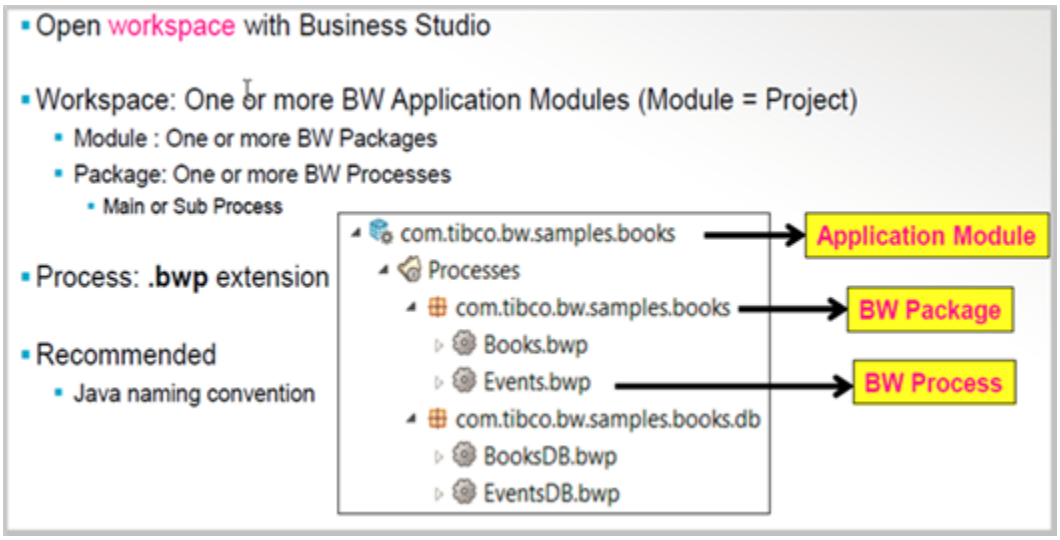
Process Development

Process development makes use of palettes consisting of several activities. ActiveMatrix BusinessWorks process development *typically involves* creating the process using activities, and implementing and invoking the service.

For details about the ActiveMatrix BusinessWorks palettes and activities, refer to*Bindings and Palettes Reference*.

Beginning Process Development

TIBCO Business Studio is used to develop processes. TIBCO Business Studio uses a workspace to save projects. A workspace contains one or more application modules. Each application module is equivalent to a project and it is stored as a folder on the disk.



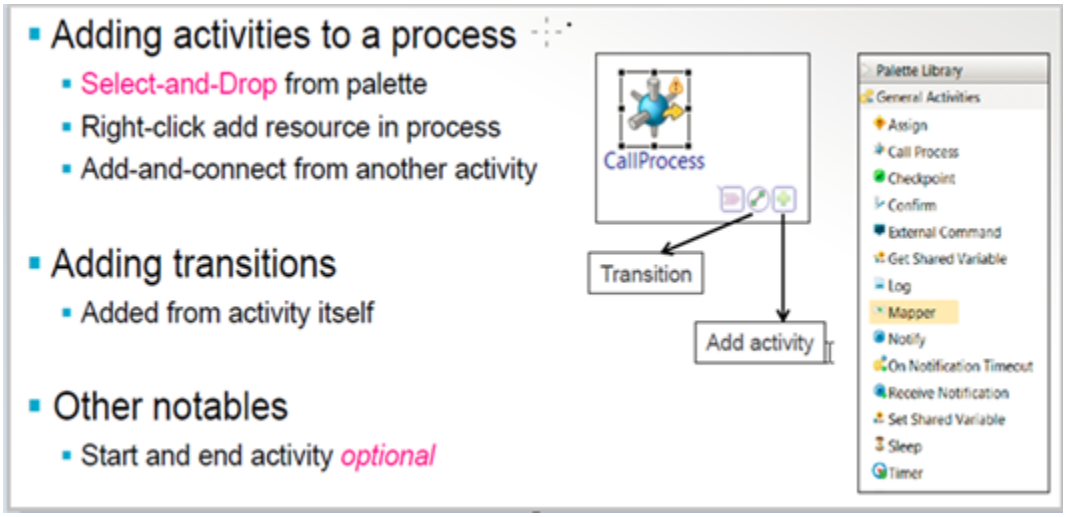
The default workspace location on Windows is `C:\users\<UID>\workspace`.

An application module contains a special folder called **Processes**. The **Processes** folder contains user created processes that are fully qualified using packages. The packages can be viewed in the **Processes** folder. In addition, an application module also contains special folders to store WSDL files, schemas, and shared resources.

It is recommended that a package follows the Java naming convention. The ActiveMatrix BusinessWorks process is stored as a single file with a `.bwp` extension.

Process Development Details

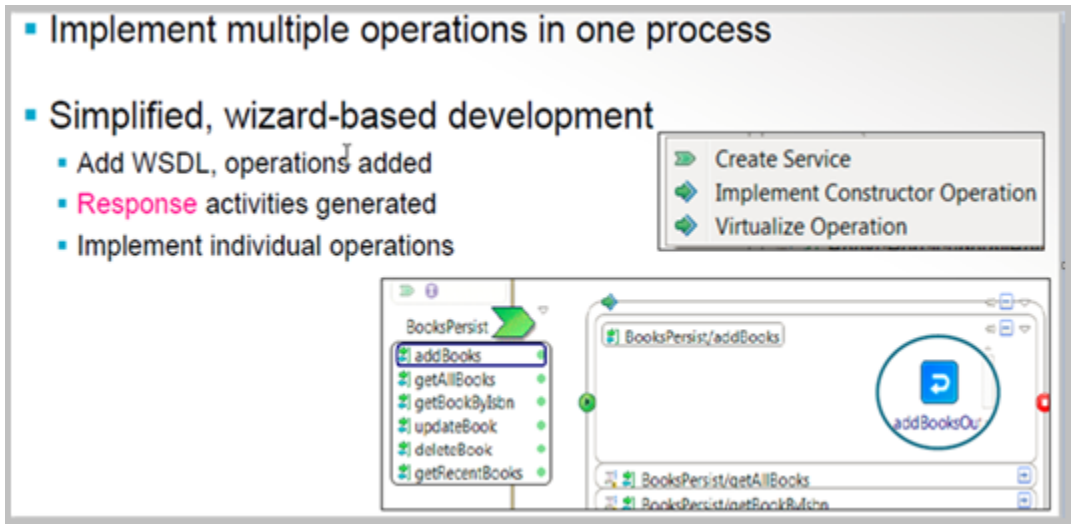
To add an activity to a process, select it from the palette window and drop (For example, click to select-and-drop. Click to drag-and-drop does not work.) it onto the right place in the process.



Developing Web Services

Web services are application components that communicate using open standard protocols. You can develop SOAP based web services with the help of a new wizard in ActiveMatrix BusinessWorks.

You can select and drop a WSDL file on to the Process Editor to implement the web services.

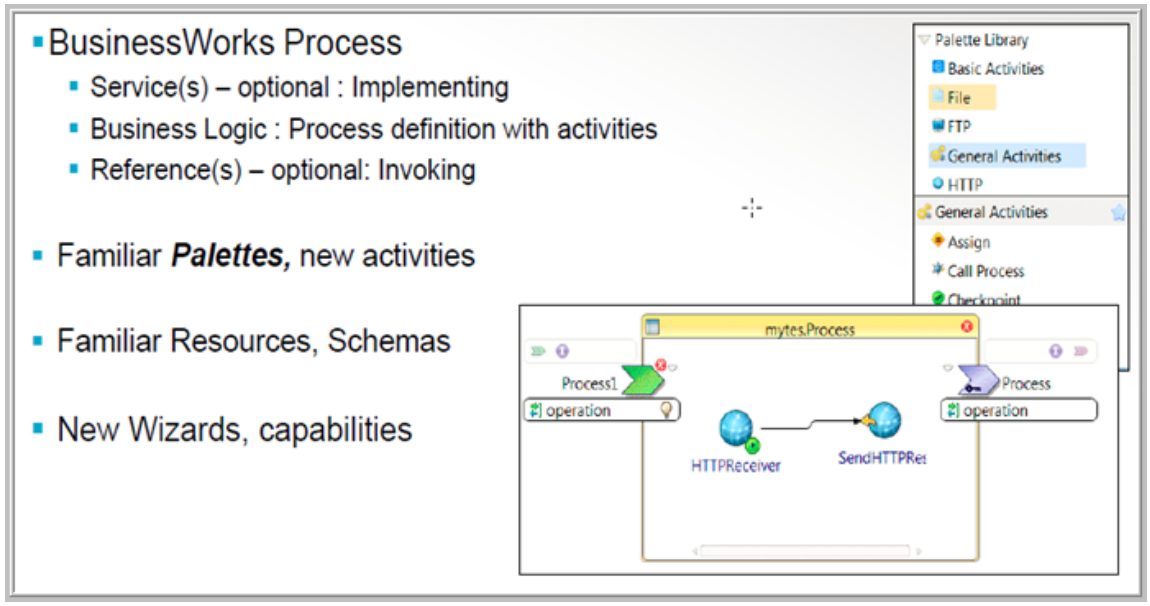


Select the **Create Service** option to define all the operations as input to the process. Next, select **Implement Constructor Operation**. This creates the starter process with the response activity. After this, you can complete the process to implement individual operations for the service.

Process Development Features

TIBCO ActiveMatrix BusinessWorks provides activities and shared resources to rapidly design business processes.

The following image describes the enhanced methodology of process development features.

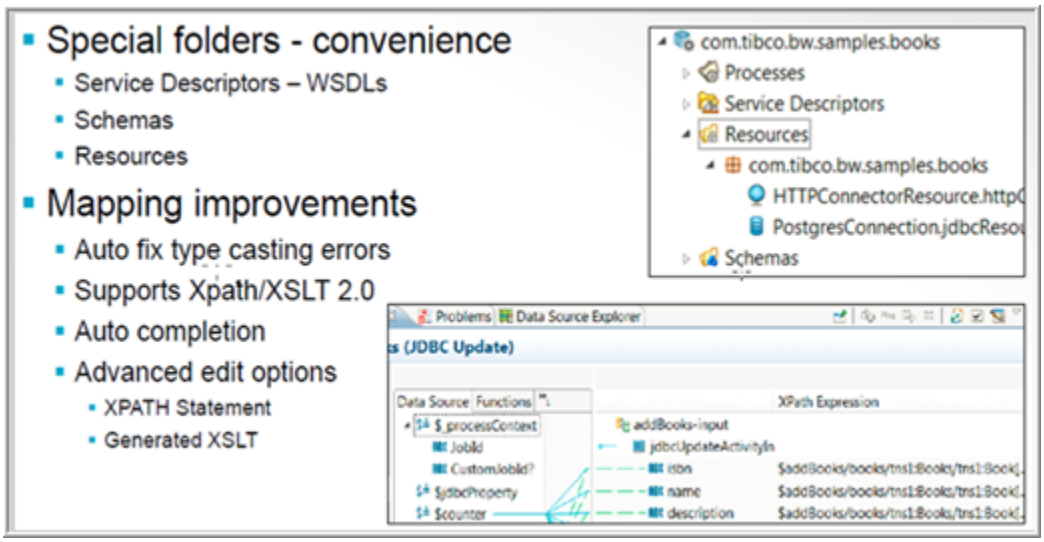


Implemented services are shown as chevrons on the left side of the **Process Editor**. Any references that are invoked are shown on the right side of the **Process Editor**. You may invoke the references depending on your use case.

For a simple process, **Services** and **References** are **optional**. However if needed, you can convert the process to implement or invoke a service.

Shared Resources and Mapping

In the **Project Explorer**, certain folders are reserved for storing specific types of assets. For example, the **Processes** folder is the default special folder that stores processes and the packages containing those processes.



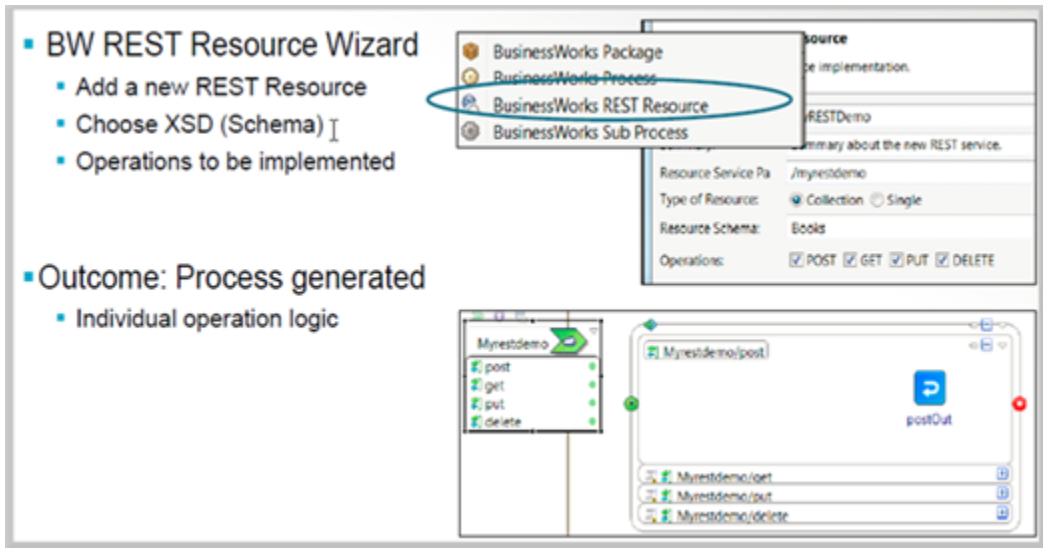
Shared resources are configurations that are shared among activities. These are resources, such as database connections, JMS and HTTP connections, and connections to other servers. For example, to connect to a database, you can create a JDBC Connection in the Resources folder. This describes the username, password, JDBC URL and other information about the connection. You can then use this connection with JDBC Palette Activities to execute queries against the database.

The following four types of special folders can exist in a TIBCO ActiveMatrix BusinessWorks project.

- **Processes**: contain TIBCO ActiveMatrix BusinessWorks business processes.
- **Resources**: contain Shared Resources used by activities in a process.
- **Schemas**: to store XSD (schema) files.
- **Service Descriptors**: to store WSDL files.

REST Support

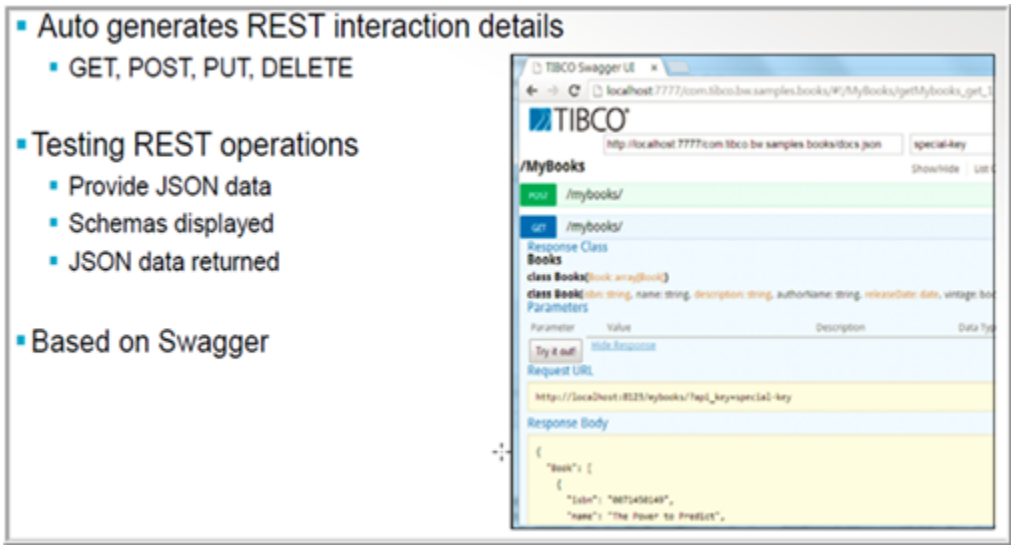
TIBCO ActiveMatrix BusinessWorks provides a new wizard for building RESTful based services. To create a new resource, select an XSD (schema) element and pick the REST operations to be implemented (POST, GET, PUT, DELETE).



You can use wizard to generate a single process by implementing a REST service with the operations picked in the wizard by using the **Constructor**. The input and output messages for the operations are automatically generated along with a Response Activity. An HTTP Shared Resource is also generated with default configuration. Now you can use other activities and add the business logic to each operation in the process.

REST Documenter and Tester

To debug a REST service, ActiveMatrix BusinessWorks automatically generates a REST documentation and tester.



The documentation is based on the Swagger specification and is rendered using the Swagger UI. This Swagger based interface provides a convenient and quick way to:

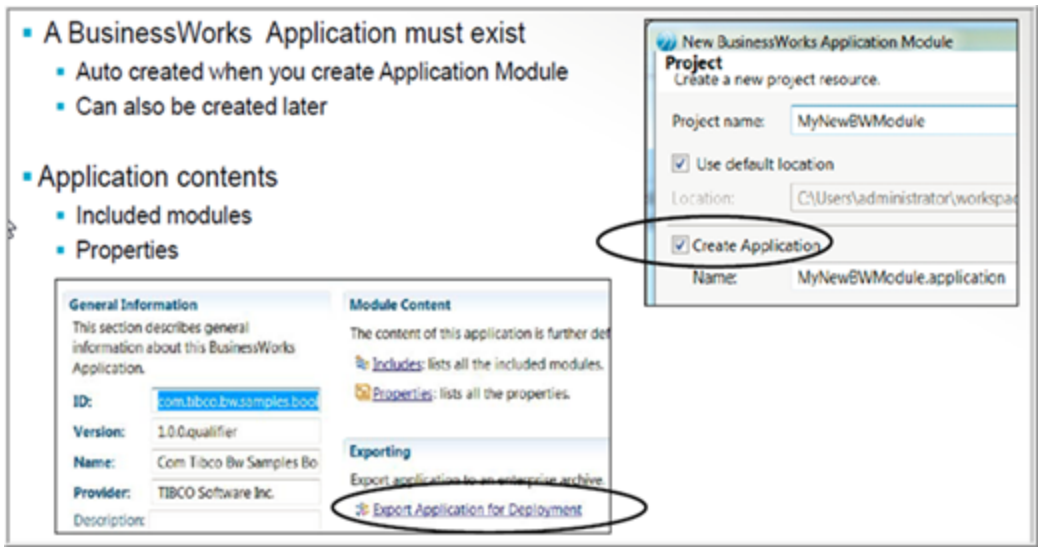
- Identify and list out all REST endpoints and operations implemented by the REST resource service.
- Examine the inputs and outputs for each operation in JSON format.
- Input fields to specify JSON or XML input for each operation and a button to invoke the operation and receive a live response for the input supplied.

Building an Archive File

After completing an application module, you must define an application to build a deployment archive file. An application defines all the processes, properties, and resources that must be included in the archive file. By default, all processes are included.

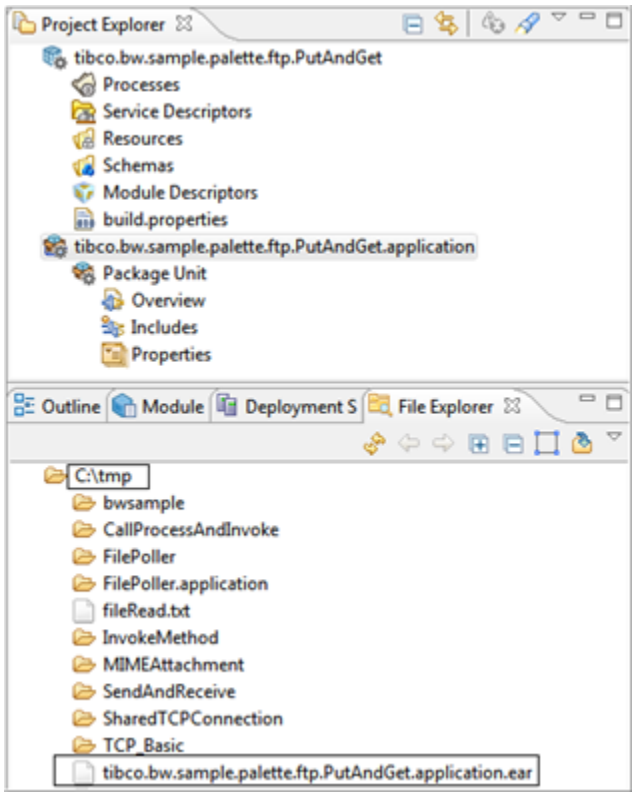
You can create an archive file using any one of following two ways:

- From TIBCO Business Studio, click **Project Application > Overview > Export Application for Deployment**.



  You can provide a custom name, if required, for the archive file by clearing the **Use Default EAR file name** check box.

- Changing the default location in the **File Explorer** using the **Open Directory to Browse** option and point it to any custom folder location, for example `c:/tmp`.



  Select the **Project Application** in the **Project Explorer** and drop it in the `c:/tmp` folder. An archive file is created.

An application module is complete when:

- All processes and subprocesses are complete.
- The module has no errors.
- All shared resources are defined.

File Poller Module Tutorial

This tutorial guides you through creating a simple FilePoller project. This simple project allows you to perform many of the same tasks as those required for the project with a larger scope and more complexity. The FilePoller project polls a directory for a specified file, and writes a new file to the same directory each time the file is modified.

This tutorial involves:

- Creating a New Project
- Configuring the File Poller Project
- Testing the FilePoller Application in the Debugger

Creating a New Project

This section guides you through creating a simple TIBCO ActiveMatrix BusinessWorks project.

Prerequisites

TIBCO ActiveMatrix BusinessWorks 6.x

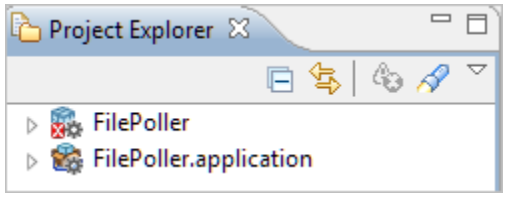
TIBCO Enterprise Administrator

Procedure

1. Start `$TIBCO_HOME\studio\3.6\eclipse\TIBCOBusinessStudio.exe`.
2. Launch the **BusinessWorks Application Module** wizard from **File > New > BusinessWorks Resources**.
   The **BusinessWorks Resource Wizard** window displays.
3. In the **Select a wizard** dialog box, choose **BusinessWorks Application Module** and click **Next**.
4. In the **Project name** field, specify the project name.
   Keep the **Use default location**, **Create empty process**, and **Create Application** check boxes selected.
5. Click **Finish**.

Result

Two projects are created and are visible in the Project Explorer.



Configuring the File Poller Project

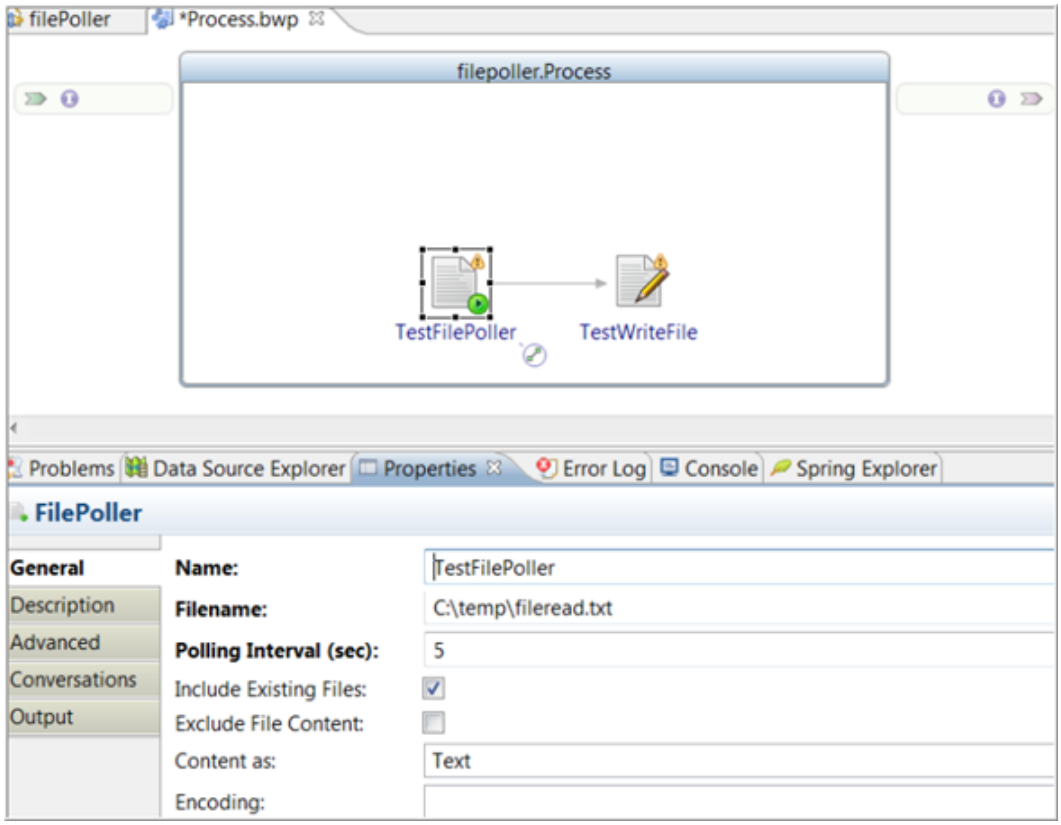
The FilePoller project uses a **File Poller** activity and a **File Write** activity. This project creates a simple process with a specified file at, for example, `c:\tmp\fileread.txt`. This file is polled periodically to determine if the file was changed.
By default, the file gets created if it does not exist in the specified location. The **File Poller** activity and the **Write File** activity from the **File** palette are used in this process.

Prerequisites

- Your computer has sufficient disk storage and you have file write permission.
- You have created a text file, for example, `c:\tmp\fileread.txt` . Add a few lines to the file and save.

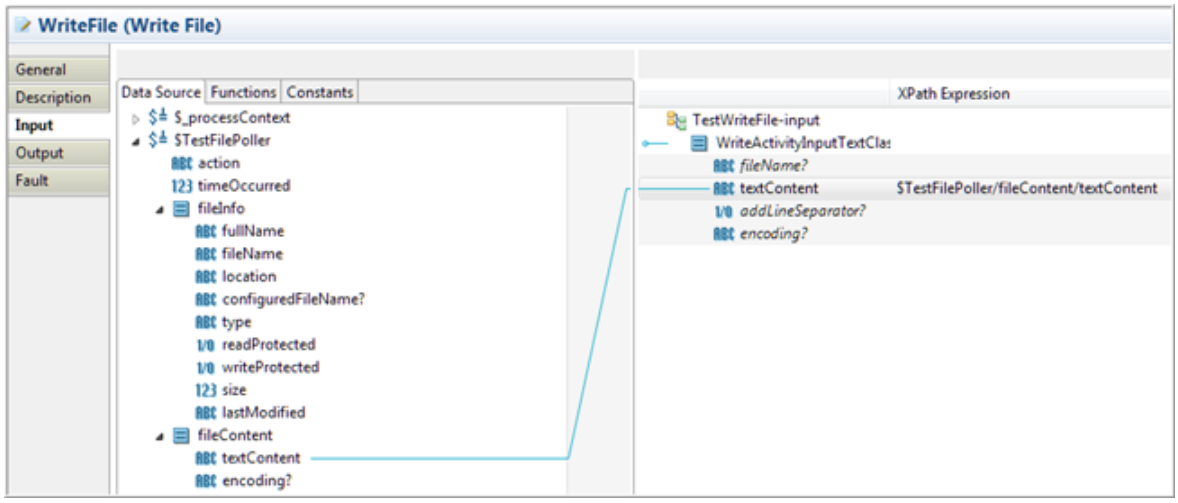
Procedure

1. Select and drop a **File Poller** activity from the **File** palette to the **Process Editor** window.
   To add an activity to the Process Editor, click the activity and drop it on the Process Editor. Do not drag and drop the activity.
2. Select the **Write File** activity from the **File** palette and drop next to the **File Poller** activity.
3. Create a transition from the **File Poller** activity to the **Write File** activity using the icon.
   You will find this icon below the Palette heading in the Palette window.
4. Select the **File Poller** activity and in the **Properties** tab, select the **General** tab on the left.
5. In the **General** tab, specify the values as shown below.



   If the **Include Existing Files** check box is selected, the File Poller activity polls the existing file regardless of the changes made. The specified file is periodically polled at the specified interval even if the file has not changed. These changes may be for modifying, creating, or removing the file.
6. To save your project, click **File > Save** or the **Save** button on the tool bar.
7. Select the **Write File** activity and click **General** tab.
8. Click on button in the **Filename** field and specify the output location, such as, `c:\tmp\FilePoller\output.log`.

9. Click the **Input** tab to specify input to the **WriteFile**. Drag the `fileContent\textContent` from the **Data Source** pane into the **`textContent`** field in the **XPath Expression** pane. This will write the content of the polled file to the output file.



10. Save your project.
11. Test this project in the debugger. For details about the testing procedure, refer to Testing the FilePoller in Debugger.

Result

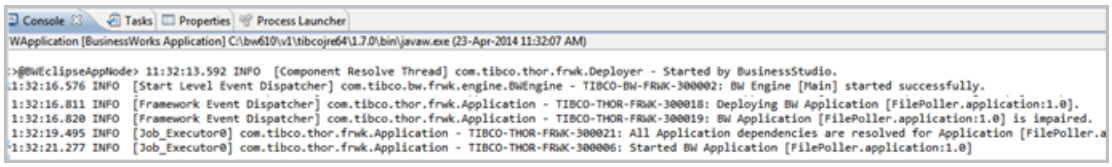
The write file is automatically created with the updated content of the read file after the change.

Testing the FilePoller Application in the Debugger

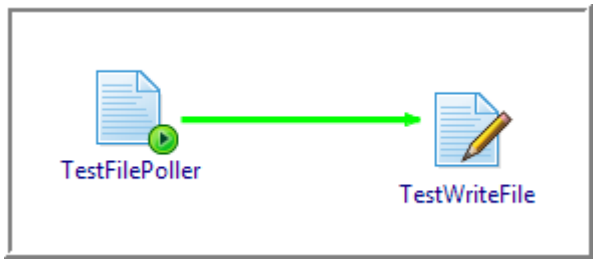
The debugger provides a simple and fast way of debugging one or more ActiveMatrix BusinessWorks applications in a local runtime environment. The applications must be in the workspace and selected or deployed before launching the debugger. After starting the debugging session, the debugger does not provide tooling support for deploying and debugging on the same runtime instance. The runtime starts when the debugger is started and stops when the debugger is stopped.

Procedure

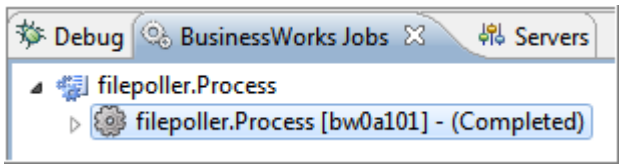
1. Open the **FilePoller** project in TIBCO Business Studio.
2. In the **Project Explorer,** select `FilePoller\Processes\filepoller\Process.bwp` and open it in the Process Editor. Click **Run > Run** in the menu bar.
3. Right-click in the Process Editor and select **Debug BusinessWorks Applications** option from the pop-up window. You can also click to start debug the application.
You will see the following console message.



4. Now you are running the instance of the FilePoller application. When the debugger is launched, the perspective changes from **Modeling** to **Debug**. You will see the Console view in the Debug perspective.
5. Modify the `c:\tmp\fileread.txt` that you have defined as the polled file in Creating a New Project. For example, open the `c:\tmp\fileread.txt` and write `"Hello BusinessWorks!"` and save the changes.
   This will create an output file in `C:\tmp\FilePoller`. This output contains the updated text matching the updated information of the `fileread.txt`.
6. Open the file `c:\tmp\FilePoller\output.log` and verify the updated information, for example, `"Hello BusinessWorks!"` printed in this file.
7. You will see the path taken by the engine for executing the process. Transitions turn to green in color to specify that a path was executed.



8. Click on the **Businessworks Jobs** view on the top left to see the jobs created for the process.



9. Click on a particular activity, then click the **Job Data** view on the top right to see the input and output data of the activity.
10. To stop the current job, click the **Stop** button on the console toolbar.

Building and Testing a REST Service

The REST Bookstore sample lets you explore the REST tooling in TIBCO Business Studio interface for ActiveMatrix BusinessWorks. You can import this sample in TIBCO Business Studio through File Explorer and examine the project and the solution implemented by it.

The BooksDB process implements different aspects of a Book Store service, such as adding books, deleting a book, and getting a list of books, or a single book by ISBN. You can build a REST Web Service, using this BooksDB web service, and then test it using the built in debugger. Using the Swagger UI provided by TIBCO ActiveMatrix BusinessWorks, you can invoke the operations for the REST resource.

This sample is about REST services. You can also use the other TIBCO ActiveMatrix BusinessWorks activities and services to create BusinessWorks processes. For other binding, core, and palettes samples, refer to *Samples*. For detailed information about the TIBCO ActiveMatrix BusinessWorks palettes and activities, refer to *Bindings and Palettes Reference*.

This sample walks you through the complete start to end process.

Prerequisites

Before working with this sample you must have the following software installed and configured on your machine.

- TIBCO ActiveMatrix BusinessWorks 6.x
- Install PostgreSQL:
  1. Download and install PostgreSQL from http://www.postgresql.org/download/
  2. Go to SQL Shell (psql) from the installed program.
  3. The PostgreSQL command line window starts.
  4. Open the DB and table creation script, `dbsetup.sql` from the scripts folder under `TIBCO_HOME\bw\n.n\samples\binding\rest\BookStore`
  5. Open the database and the table creation script, `dbsetup.sql`
  6. Run the script from the command line as per the instructions mentioned in the readme file under the `scripts` folder.

     You will see the database and the tables created from the **pgAdmin UI** from the installed program.
- Google Chrome: Install latest of version of Google Chrome at the standard default location from https://www.google.com/intl/en/chrome/browser/

Sample Flow

You must follow the steps mentioned in the following sections for successfully creating and testing the REST service.

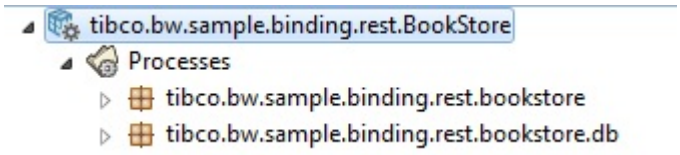
- Building a REST Service
- Testing the REST Service

Creating a New BusinessWorks Process

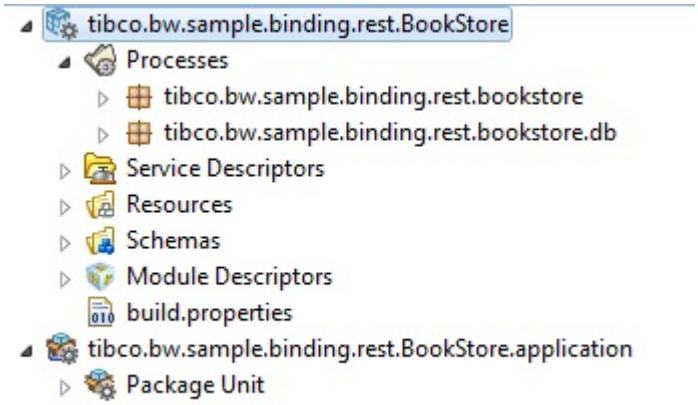
This sample shows how to create a new BusinessWorks Process.

Procedure

1. Start `$TIBCO_HOME\studio\3.6\eclipse\TIBCOBusinessStudio.exe`.
2. In the upper-right tab, click the **Modeling** perspective.
3. Click the **File Explorer** tab. If the tab is not visible, click **Window > Show View > Other > FileSystem > File Explorer** and click **OK**.
4. Click **File > Switch Workspace** and select or open a clean new workspace.
5. In the samples directory, select **binding > rest > Bookstore > apps** and double-click **tibco.bw.sample.binding.rest.BookStore**.
6. In the **Project Explorer** expand the **tibco.bw.sample.binding.rest.BookStore** project. Do not close the project.
7. You can also import the sample using the **File > Import > General > Existing Studio Projects into Workspace > Select archive File > Browse** option.
8. You will see the project displayed in the Project Explorer panel on the left.



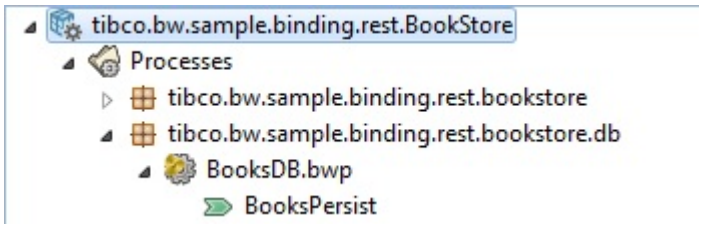
9. Expand the `tibco.bw.sample.binding.rest.BookStore` project and verify its components.



In this window you see:

- `BusinessWorks Application Module: tibco.bw.sample.binding.rest.BookStore`
- `BusinessWorks Application: tibco.bw.sample.binding.rest.BookStore.application`

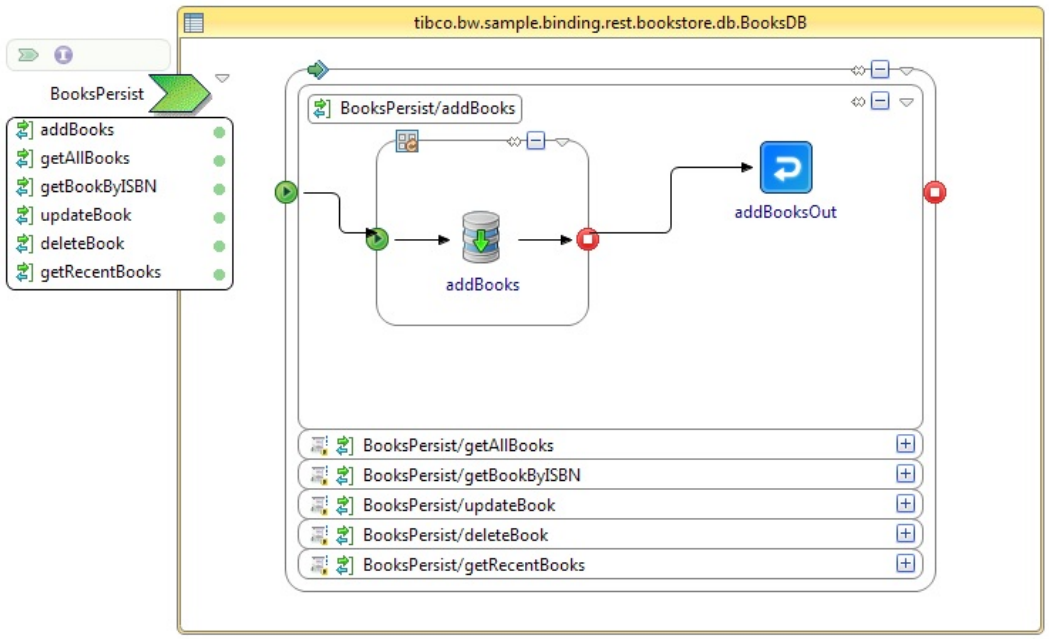
10. Expand **Processes** and then expand **tibco.bw.sample.binding.rest.bookstore.db**.
    You will see **BooksDB.bwp** (*bwp: BusinessWorks Process*).



11. Double-click **BooksDB.bwp**.
    The BusinessWorks process comprises:

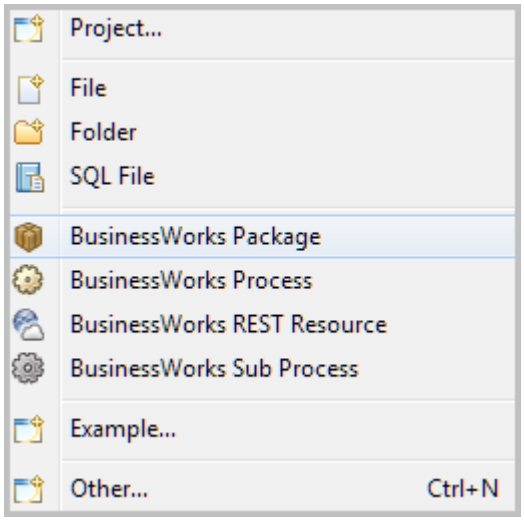
    - Green chevron on the left indicates the service details.
    - addBooks, getAllBooks, and so on indicate the operations implemented by this process.
    - Each operation is implemented separately.

    The BusinessWorks Process opens. Verify the BusinessWorks Process details.

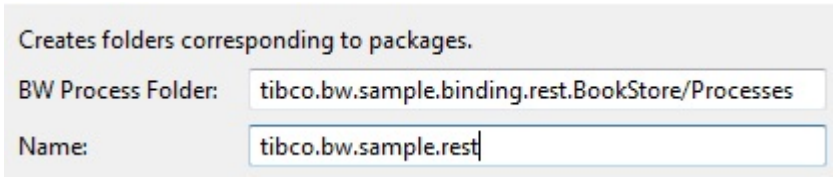


12. Double-click on the operation to display the process for example, **BookPersist > addBooks**.
    a) In addBooks, you can see a JDBC activity.
    b) The activity is repeated using a ForEach group.
    c) addBooksOut represents the **Reponse** to the web service request.

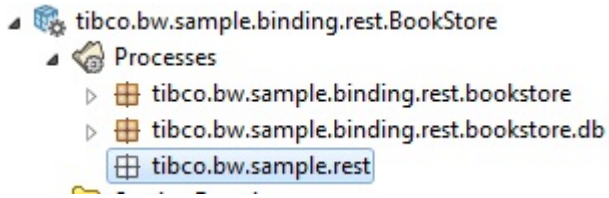
13. To add a new process package named `tibco.bw.sample.rest`, right-click on **Processes** in the Project Explorer and select **New > BusinessWorks Package**.



14. In the **BusinessWorks Package** screen, specify `tibco.bw.sample.rest` in the **Name** field.



15. Click **Finish** and verify the new package `tibco.bw.sample.rest` in the Project Explorer (same as the image).

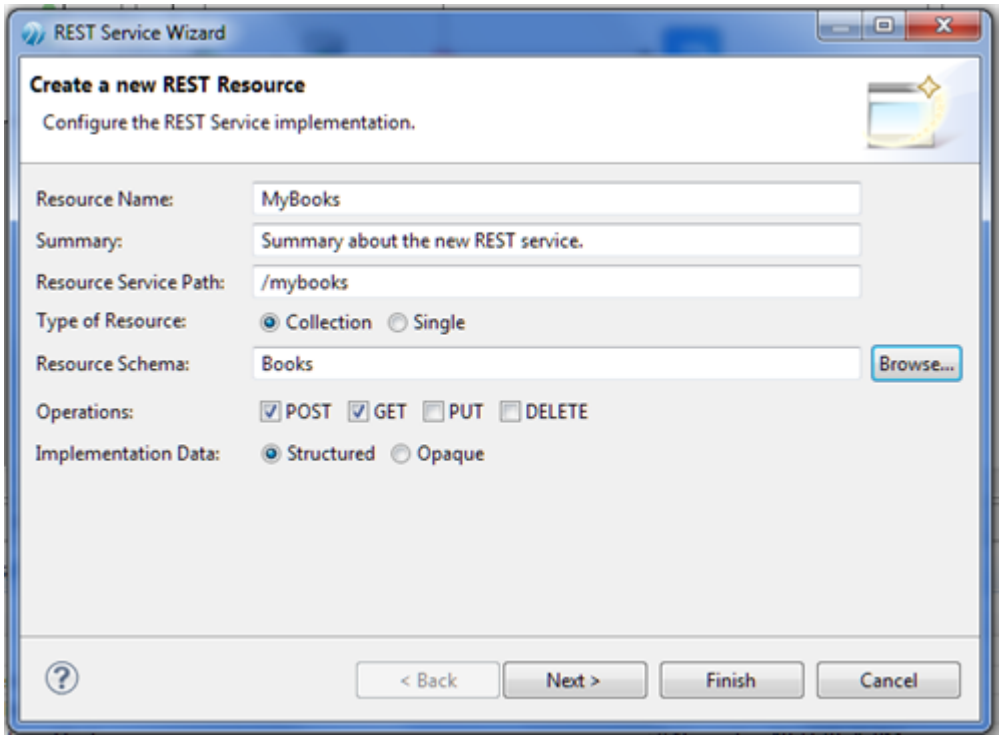


Building a REST Service

Here you will learn how to build a REST service.

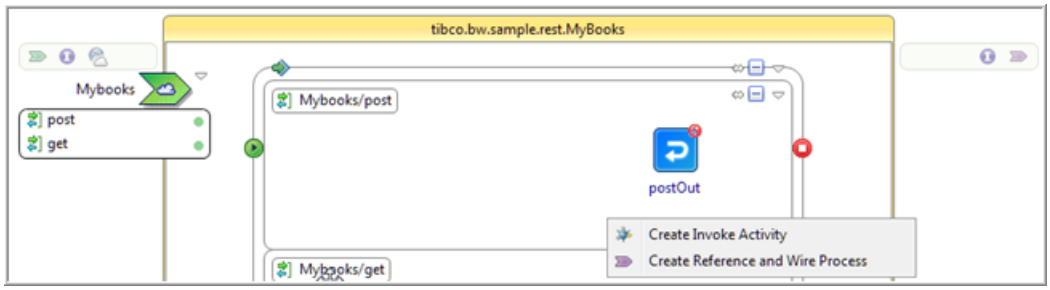
Procedure

1. To define a BusinessWorks REST Resource named **MyBooks**, select **tibco.bw.sample.rest > New > BusinessWorks Rest Resource**.
   The **REST Service Wizard** window opens.

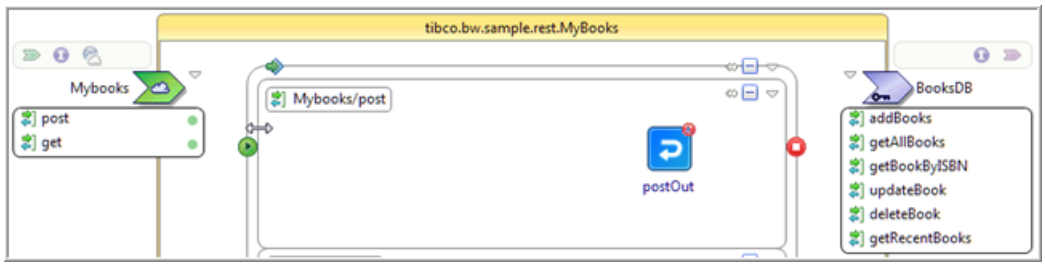


2. Specify the following values in the **REST Service Wizard** window.
   a) **Resource Name**: MyBooks
   b) **Summary**: Add the summary about your REST Service.
   c) **Resource Service Path**: Auto filled
   d) **Type of Resource**: Collection (default)
   e) **Resource Schema**: Select **Browse > Books** in the **Select Schema Element Declaration** window.
   f) **Operations**: Select POST and GET check boxes.
   g) **Implementation Data**: Structured (default)
3. Click **Finish**.
   This will create a new ActiveMatrix BusinessWorks process.

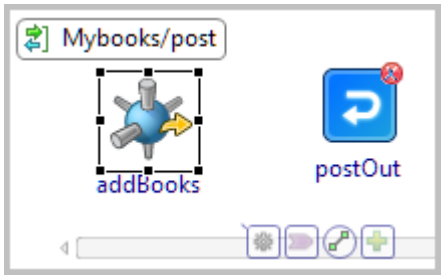
4. From Project Explorer, select the **BooksDB.bwp** and drop it close to the implemented operation. A wizard pops up with `Create Invoke Activity` and `Create Reference and Wire Process` options.



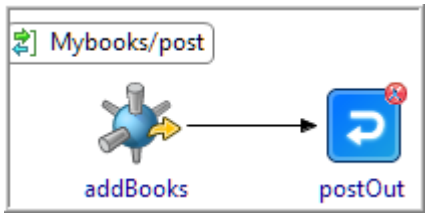
5. Select **Create References and Wire Process**.
The references are added to the process. The purple chevron indicates the service and its operations, that can be referenced by the process.



6. To update the POST process to invoke the appropriate external service operation:
   a) Click on **addBooks** operations from the reference activity.
   b) Select and drop the operation to the left of **postOut** activity. An invoke process activity is created.

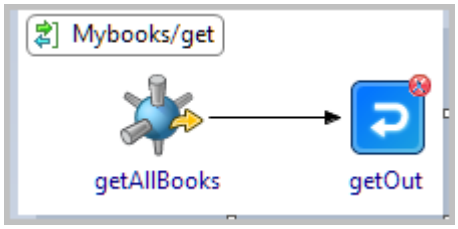


7. Click on the newly added activity. Select the icon and connect **addBooks** to **postOut**.

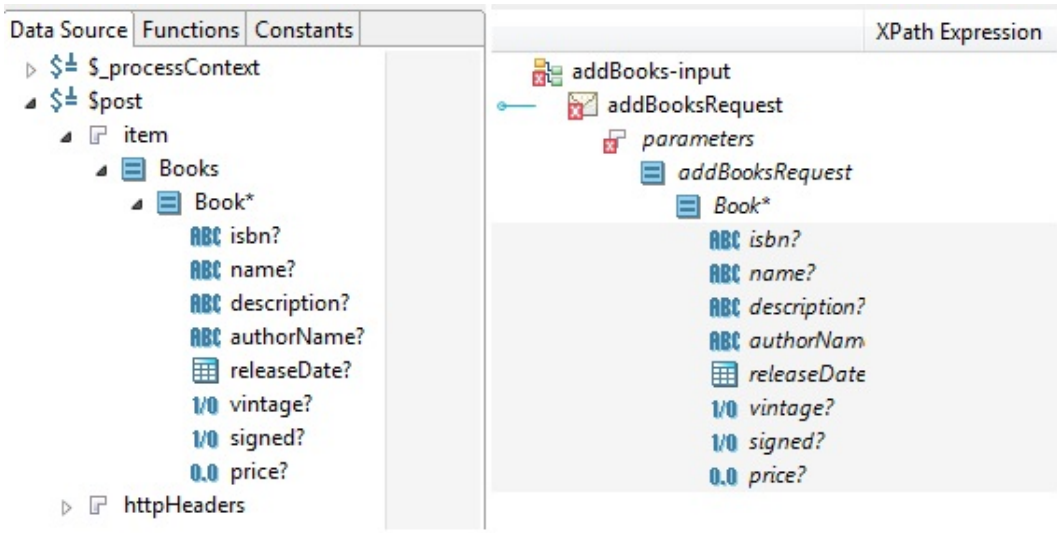


8. Click on **getAllBooks** operations and drag and drop the operation to the left of **getOut** activity.

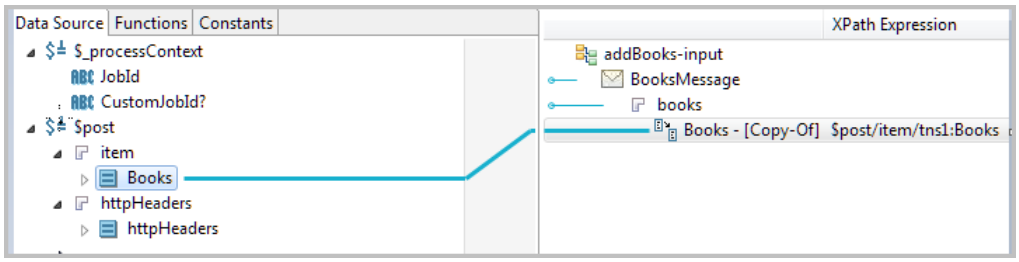
9. Connect **getAllBooks** to **getOut**.



10. Save your changes.
11. Click on **addBooks** activity and select **Properties > Input**.
12. Expand the data tree in the **Data Source** tab to locate Books.

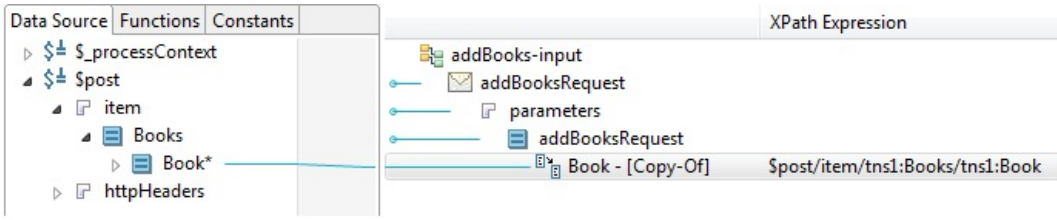


13. Map (drag) the Book* element from the left to the Book* element on the right.
14. In the pop-up window, select **Make a Copy of Books** and click **Finish**.

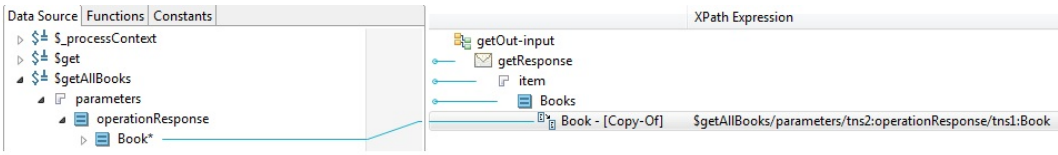


15. Save your changes.

16. Click on **postOut** activity, and select **Properties > Input** to map the **Book*** element from left to right.



17. Click `getAllBooks` and select **Properties > Input**.
18. Put any dummy value in the input such as, "Get All Books".
19. Click the **getOut** activity, and select **Properties > Input** to map the Book* element from left to right.



Result

Your project will show no errors.

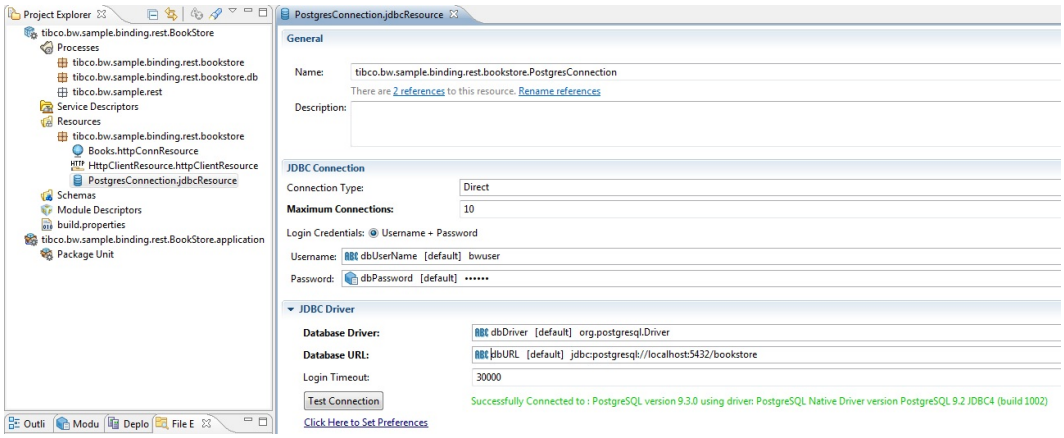
Testing the REST Service

After creating a ActiveMatrix BusinessWorks process and building a REST Service, you test the REST service using the built-in tester and the Swagger UI.

Procedure

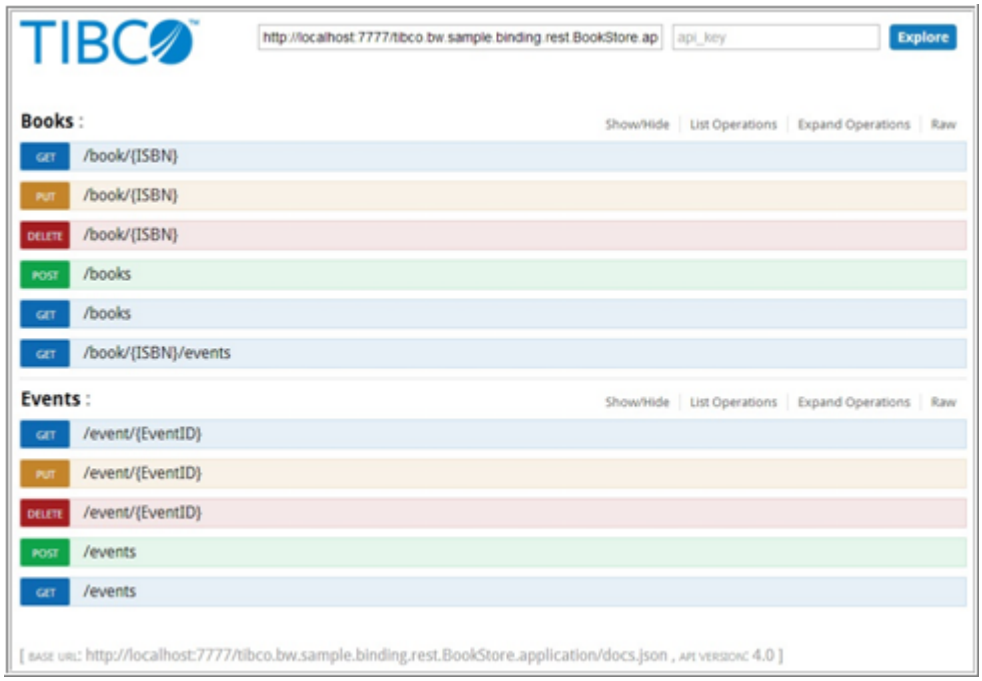
1. Select the **Project Explorer** tab and double-click **tibco.bw.sample.binding.rest.BookStore**.
2. Set the default ApplicationProfile to match the OS system you are running on.
3. Expand **tibco.bw.sample.binding.rest.Bookstore.application > Package Unit** folder. Provide valid values for the application properties. Provide a valid username, password and database URL to connect to your PostgreSQL database in the module properties, if different from the default setting.

4. Verify your JDBC connection.
   a) Expand the **Resources** directory.
   b) Double-click **PostgresConnection.jdbcResource**.
   c) In JDBC Driver, click **Click Here to Set Preferences**.
   d) Set the JDBC driver folder directory preference to the folder where you have downloaded PostgreSQL JDBC Driver as mentioned in the Prerequisites section in Building and Testing a REST Service. Click **Apply** and then click **OK**.
   e) Click the **Test Connection** button to verify the connection.



5. Click **File > Save**.
6. Fully expand the **Processes** directory and double-click **MyBooks.bwp**.
7. Click **Run > Debug Configurations**.
8. At the left hand tree of **Debug Configuration** wizard, expand **BusinessWorks Application** and select **BWApplication**
9. Click **Applications** tab and then click the **Deselect All** button if you have multiple applications. Select the check box next to **tibco.bw.sample.binding.rest.BookStore**.
10. Click **Debug**. This will run the sample in Debug mode.
    The console window shows engine messages similar to: `Started BW Application [tibco.bw.sample.binding.rest.BookStore.application:1.0]`. The OSGI command to list REST and Swagger URLs is **`lrestdoc`**, which lists the discovery URL: `[Application Name]: tibco.bw.sample.binding.rest.BookStore.application [Discovery Url]: http://localhost:7777/tibco.bw.sample.binding.rest.BookStore.application`.

11. Launch Google Chrome browser and open http://localhost:7777/tibco.bw.sample.binding.rest.BookStore.application. The Google Chrome page displays as follows.



12. Click on any of the operations, such as POST, GET, PUT, DELETE as displayed on the web page.
13. After you have finished with the operations on the page, in TIBCO Business Studio, click the **Terminate** icon to stop the process.
14. OSGI Command to list REST and Swagger URL's is "lrestdoc " which lists the discovery URL as follows:

    Application Name]: tibco.bw.sample.binding.rest.BookStore.application

    [Discovery Url]:
    http://localhost:7777/tibco.bw.sample.binding.rest.BookStore.application

Result

The web page lists the following operations for Books and Events:

Books

- Post books
- GET books
- GET book by ISBN
- PUT book by ISBN
- DELETE book by ISBN

Events

- POST Events
- GET Events
- GET Event by EventID

- PUT Event by EventID
- DELETE Event by EventID

GET books returns an output similar to the following:

```
{
  "Book": [
    {
      "isbn": "0061122416",
      "name": "The Alchemist",
      "description": "Every few decades a book is published that changes the lives
of its readers forever. The Alchemist is such a book",
      "authorName": "Paul Coelho",
      "releaseDate": "2006-04-25",
      "vintage": true,
      "signed": true,
      "price": 11.9
    },
    {
      "isbn": "0071450149",
      "name": "The Power to Predict",
      "description": "How Real Time Businesses Anticipate Customer Needs, Create
Opportunities, and Beat the Competition",
      "authorName": "Vivek Ranadive",
      "releaseDate": "2006-01-26",
      "vintage": false,
      "signed": true,
      "price": 15.999
    }
 ]
}
```

GET books by ISBN returns an output similar to the following for the ISBN 0061122416:

```
  {
      "isbn": "0061122416",
      "name": "The Alchemist",
      "description": "Every few decades a book is published that changes the lives
of its readers forever. The Alchemist is such a book",
      "authorName": "Paul Coelho",
      "releaseDate": "2006-04-25",
      "vintage": true,
      "signed": true,
      "price": 11.9
    }
```

The `books.log` file is generated with the following information:

```
POST Books------->{"Book":[{"isbn":"1451648537","name":"Steve
Jobs","description":"Biography of Apple Co-Founder Steve Jobs","authorName":"Walter
Isaacson","releaseDate":"2012-10-24","vintage":false,"signed":false,"price":21},
{"isbn":"0385537859","name":"Inferno","description":"Robert Langdon returns in Dan
Brown's latest fast paced action thirller","authorName":"Dan
Brown","releaseDate":"2013-05-14","vintage":false,"signed":true,"price":14.09},
{"isbn":"0399103421","name":"The Godfather","description":"The Godfather is an epic
story of a New York's top mafia family, loyalty, and how men of honor live in their
own world, and die by their own laws.","authorName":"Mario
Puzo","releaseDate":"1969-03-10","vintage":true,"signed":true,"price":
50}]}************************************************************

GET Books------->{"Book":[{"isbn":"1451648537","name":"Steve
Jobs","description":"Biography of Apple Co-Founder Steve Jobs","authorName":"Walter
Isaacson","releaseDate":"2012-10-24+05:30","vintage":false,"signed":false,"price":
21},{"isbn":"0385537859","name":"Inferno","description":"Robert Langdon returns in
Dan Brown's latest fast paced action thirller","authorName":"Dan
Brown","releaseDate":"2013-05-14+05:30","vintage":false,"signed":true,"price":
14.09},{"isbn":"0399103421","name":"The Godfather","description":"The Godfather is
an epic story of a New York's top mafia family, loyalty, and how men of honor live
in their own world, and die by their own laws.","authorName":"Mario
Puzo","releaseDate":"1969-03-10+05:30","vintage":true,"signed":true,"price":
50}]}************************************************************

GET Book By ISBN------->{"isbn":"1451648537","name":"Steve
```

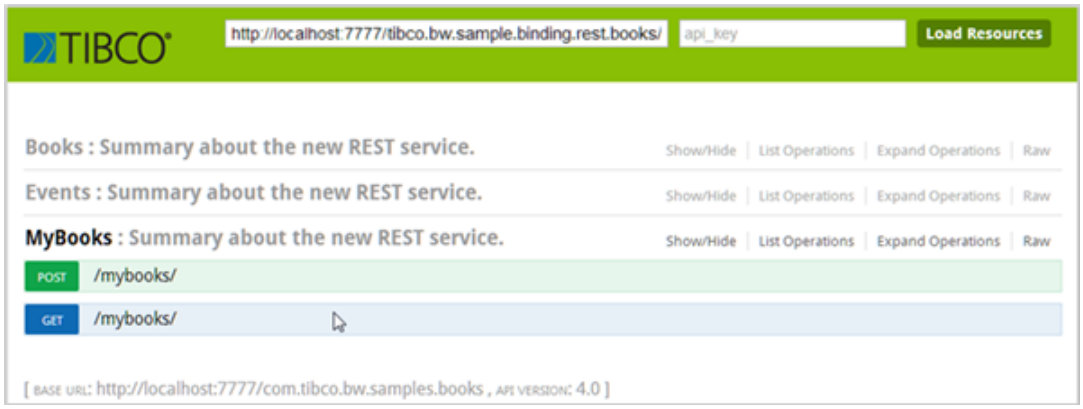
```
Jobs","description":"Biography of Apple Co-Founder Steve Jobs","authorName":"Walter
Isaacson","releaseDate":"2012-10-24+05:30","vintage":false,"signed":false,"price":
21}************************************************************

DELETE Book By ISBN-------->"Deleted book with ISBN -
1451648537"************************************************************

GET Events By ISBN----
>{}************************************************************
```

Testing the POST and GET Operation

An available RESTful service displays the GET operation in the Swagger UI. The POST operation is testing using the JSON service. It is important to test these operations by doing some simple tasks.

Procedure

1. Click on **/MyBooks** hyperlink. It expands and displays the POST and GET operations.



2. Click on the **POST** icon to display its details.
3. Provide values in the Books Parameter. You can use the json payload from `books.json` file which is under `c:/tmp` .
4. Click the **Try it out!** button.
5. Now click on the **GET** icon to display its details.

6. Click the **Try it out!** button.
   The response displays a list of books returned by the REST service from the database.

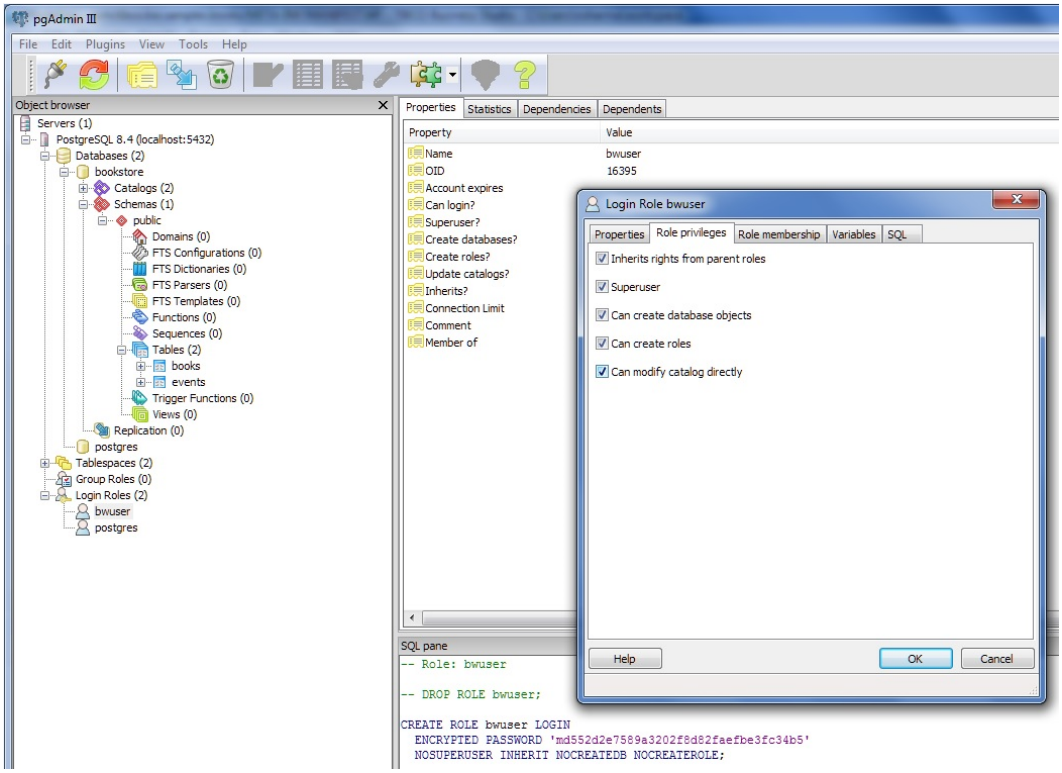
7. After you have finished with the operations on the page in TIBCO Business Studio, click the Terminate icon to stop the process.

Troubleshooting

Your may encounter some errors while execution or running the process. By verifying simple details these errors can be resolved.

Following are some of the possible errors you may encounter and their resolutions.

Error Encountered	Resolution
Unable to insert rows into the database using the **`dbsetup.sql`** script in the scripts folder.	Use the sample json payload from the sample json folder to post the data.
The REST Swagger API page is not visible.	Verify whether the application has started.
Problem markers are visible in the project.	Clean the project by invoking **Project > Clean** or try switching to a clean new workspace.
Getting `File was not found` exception.	Ensure that `books.json` and `book_put.json` file is present at the location as described in the Input_File and Input_File_1 Module Property.
Getting an `unregistered user` error message while running the process.	Select all the check boxes in the **Role Privileges** tab in the **pgAdmin UI** and run the process again. See image below.


pgAdmin III
File Edit Plugins View Tools Help
Object browser
Servers (1)
PostgreSQL 8.4 (localhost:5432)
Databases (2)
bookstore
Catalogs (2)
Schemas (1)
public
Domains (0)
FTS Configurations (0)
FTS Dictionaries (0)
FTS Parsers (0)
FTS Templates (0)
Functions (0)
Sequences (0)
Tables (2)
books
events
Trigger Functions (0)
Views (0)
Replication (0)
postgres
Tablespaces (2)
Group Roles (0)
Login Roles (2)
bwuser
postgres
Properties Statistics Dependencies Dependents
Property Value
Name bwuser
OID 16395
Account expires
Can login?
Superuser?
Create databases?
Create roles?
Update catalogs?
Inherits?
Connection Limit
Comment
Member of
Login Role bwuser
Properties Role privileges Role membership Variables SQL
Inherits rights from parent roles
Superuser
Can create database objects
Can create roles
Can modify catalog directly
Help OK Cancel
SQL pane
-- Role: bwuser
-- DROP ROLE bwuser;
CREATE ROLE bwuser LOGIN
ENCRYPTED PASSWORD 'md552d2e7589a3202f8d82faefbe3fc34b5'
NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;

Enterprise Administration

TIBCO ActiveMatrix BusinessWorks supports an Enterprise Administration framework that is hosted by TIBCO Enterprise Administrator (TEA). Through this Enterprise Administration framework, the user can deploy their application using **`bwadmin`** command line utility or through Admin Web UI.

In this Getting Started guide, we will demonstrate:

1. Creation of Domains, AppSpaces, AppNodes, uploading and deploying Enterprise Archive files to become running Applications through **`bwadmin`** command line utility, and
2. Navigation of these created assets (Domains, AppSpaces, AppNodes, Enterprise Archive files, and Applications) through Admin Web UI.

For more information on the concepts of the above entities, refer to *Administration Concepts* section in *Concepts* guide.

For more information on the commands used in this sample, refer to *Getting Started* section in *Administration* guide.

Running the Admin Sample

This sample walks you through running the Bookstore and Acme administration scripts.

Prerequisites

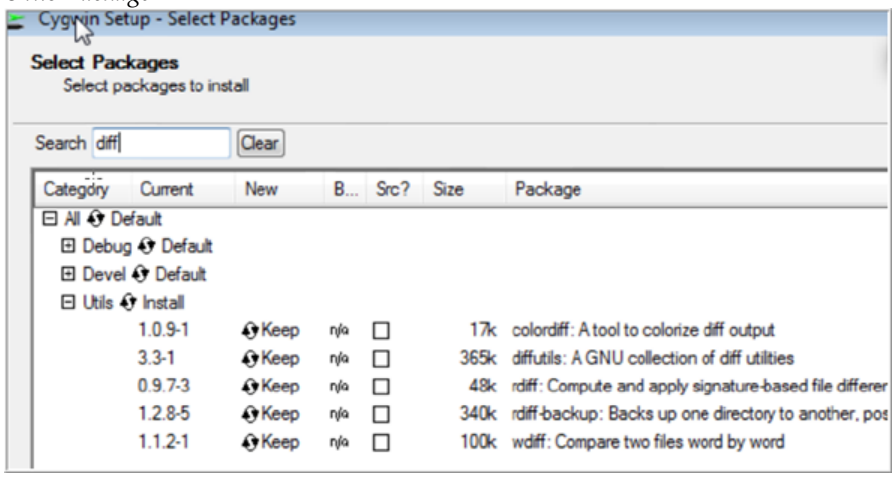
1. Install TIBCO ActiveMatrix BusinessWorks.
   - On Windows: `c:\<tibco_home>\bw\n.n`
   - On Mac: `/Users/<user_name>/bw/n.n`

   Ensure that the installation directory does not contain a space if you want to run the Bookstore sample using the bundled script **`./runBookStore.sh`**.
2. Install TIBCO® Enterprise Administrator. Follow the instructions in the installer.

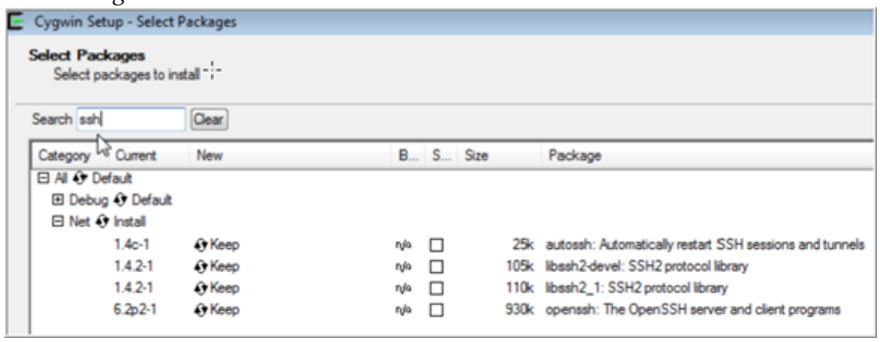
   When prompted for JAVA_HOME, you can use `$TIBCO_HOME/tibcojre64/1.7.0`
3. Optional. Install TIBCO Enterprise Message Service™ 8.x or higher if you want to run the Acme.com applications with **`./runAcme.sh`**
4. Install the latest version of Google Chrome at the standard default location from https://www.google.com/intl/en/chrome/browser/
5. Optional. Install PostgreSQL 9.3.x if you want to run the Bookstore sample with **`./runBookStore.sh`**
   - Download and install PostgreSQL 9.3.x from http://www.postgresql.org/download/
6. On Windows, it is mandatory to install the latest Cygwin 64-bit version from http://www.cygwin.com/. Install the **Utils** and **Net** package which contains the `diff.exe` and `ssh.exe` executables.

7. Select the **Utils** and **Net** packages and change the option from `Default` to `Install`. Refer to the following screenshots that show the **Utils** and **Net** packages.

Utils Package



Net Package



Procedure

1. Set up `.bashrc`. For details, refer to Core Admin Sample Scripts.

   - On Windows: Copy *TIBCO_HOME*`\bw\n.n\samples\core\admin\bashrc.bw6.windows` to `c:\cygwin64\home\<username>\`
   - Startup the Cygwin terminal from **All Programs > Cygwin64 Terminal**.
   - Rename `bashrc.bw6.windows` file using the following commands:

     1. **`rm .bashrc`**
     2. **`mv bashrc.bw6.windows .bashrc`**

   - On UNIX: Copy *TIBCO_HOME*`\bw\n.n\samples\core\admin\bashrc.bw6.unix` to `~/<username>`
   - Rename `bashrc.bw6.unix` file using the following commands:

     1. **`rm .bashrc`**
     2. **`mv bashrc.bw6.unix .bashrc`**

2. Edit `~/.bashrc` file and make sure the TIBCO_HOME configuration points to the ActiveMatrix BusinessWorks installation home. This **`.bashrc`** script assumes that TIBCO Enterprise Message Service and TIBCO Enterprise Administrator are installed in the same TIBCO_HOME as ActiveMatrix BusinessWorks. Do customize if you have installed TIBCO Enterprise Message Service and TIBCO Enterprise Administrator in a different TIBCO_HOME from that of ActiveMatrix BusinessWorks.
3. After saving the ~/`.bashrc` file, close the command line terminal and restart a new instance.

This step is to ensure the newly started instance of the command line terminal shows it has displayed "**sourced ~/.bashrc**" message. This indicates that `~/.bashrc` settings are now in effect.

4. Change to the `admin` folder by typing: `admin`
5. To create all Domains contained in this sample, issue the following command: **`./runAll.sh`**

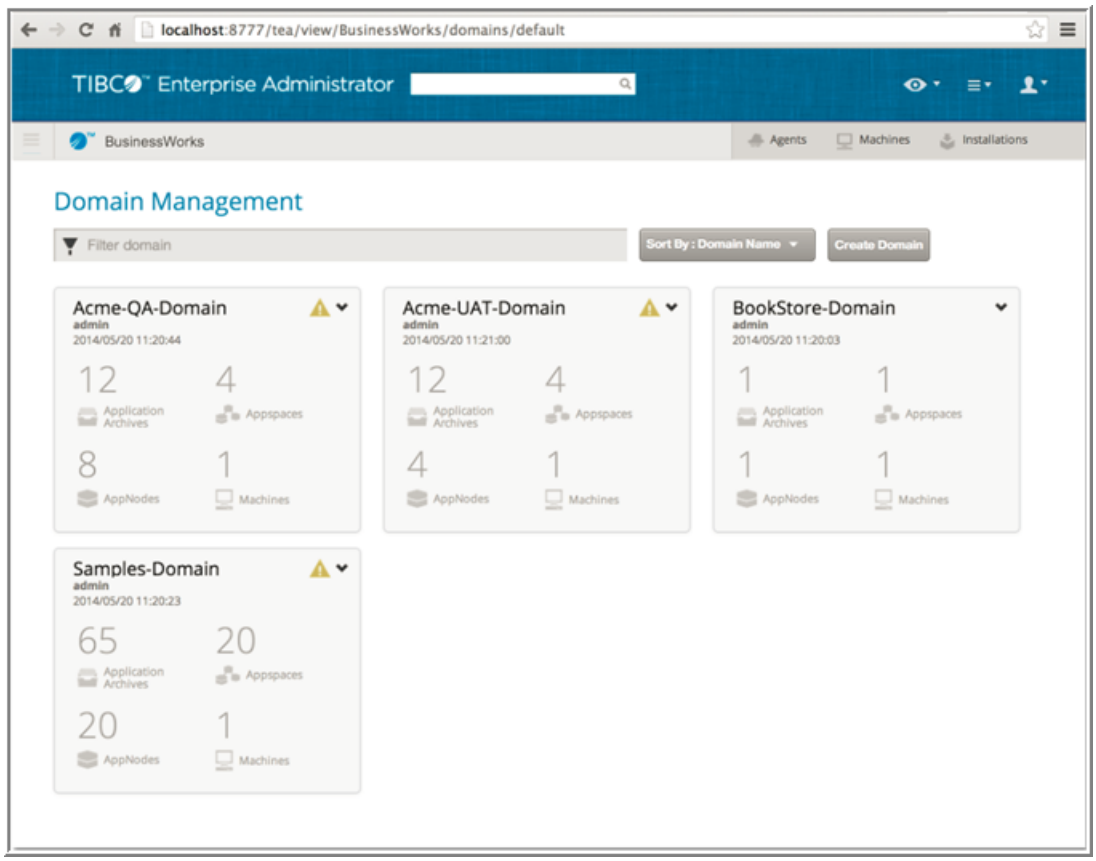
You can also issue the **`time ./runAll.sh`** command. The time command measures how long it takes for **`./runAll.sh`** to complete.

6. Launch Google Chrome. Open TIBCO Enterprise Server at the following URL: `http://localhost:8777/tea`
7. Log in to TIBCO Enterprise Administrator using the credentials of **Username**: `admin` and **Password:** `admin`

8. Click BusinessWorks in the **Products** list to see the following domains in the Domain Management screen.

   - Acme-QA-Domain
   - Acme-UAT-Domain
   - BookStore-Domain
   - Samples-Domain

   Drill down into any of the domains to see the created Admin assets, such as Archives, AppSpaces, AppNodes, and Applications.



Deploying the Application

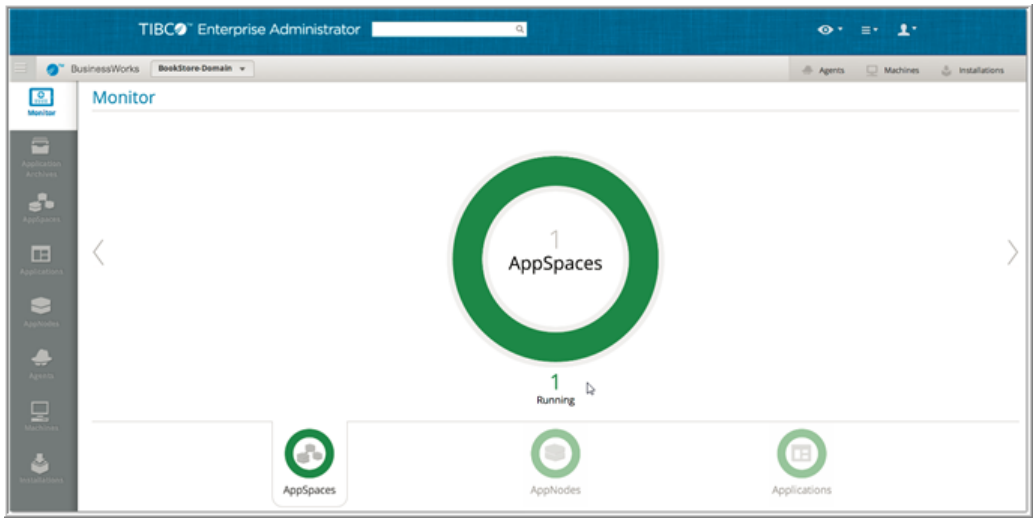
Deploying the application involves creating a Domain, an AppSpace, and an AppNode.

Procedure

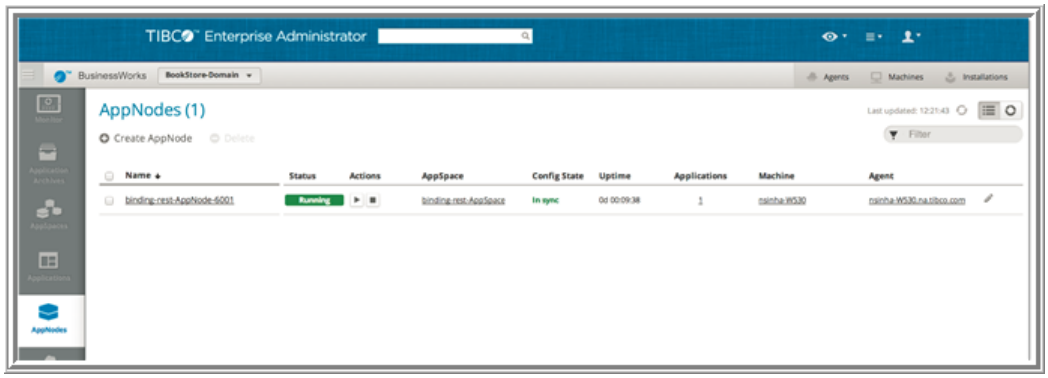
1. In the TIBCO Enterprise Administrator browser window, click **TIBCO Enterprise Administrator** at the top of the page to open the home page.

2. Click the **BusinessWorks** icon in the **Products** list to display the **Domains** page. The BookStore-Domain is displayed.

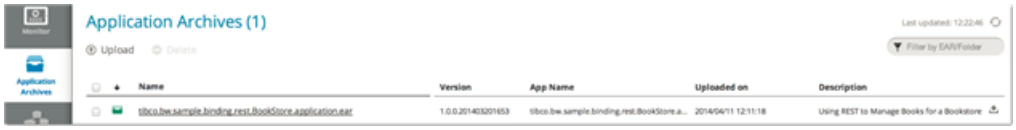
3. Click **BookStore-Domain** to drill down into domain details.
4. Click an entity such as **AppSpaces** to pivot views.



5. Click **AppNodes** to pivot to the AppNodes view and verify that the AppNode is running.

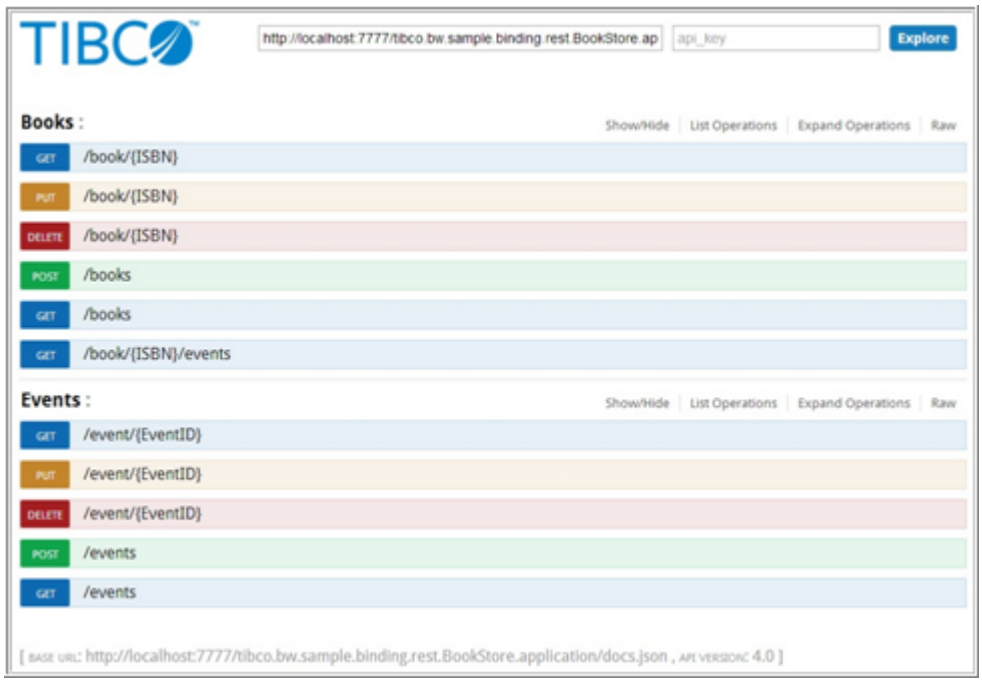


6. To view the BookStore application deployed earlier, select **Application Archives** on the left. A single application archive is displayed.



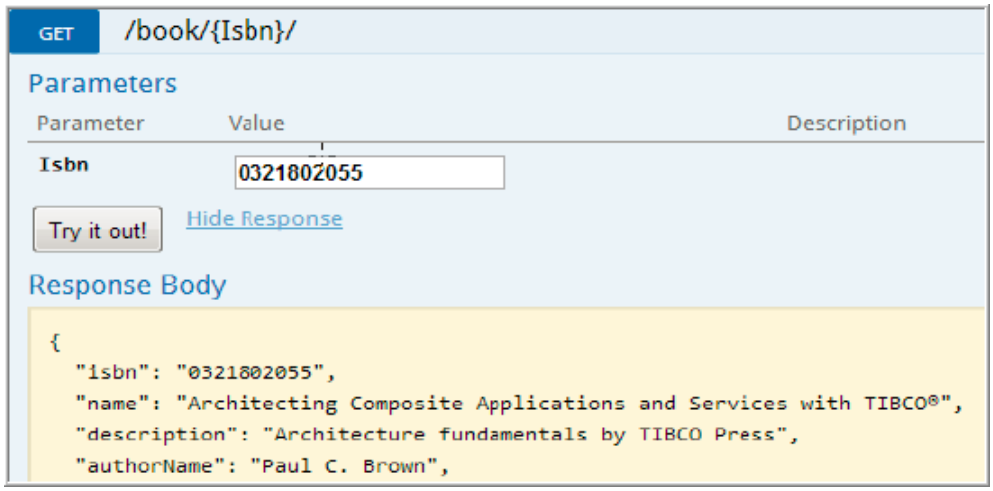
7. Click the archive **tibco.bw.sample.binding.rest.BookStore**.
The **Deployed Application** tab shows the application status as **Running**.

8. Test the deployed application in the Google Chrome browser using Swagger UI.



9. To execute the REST operations exposed by the BookStore application, click on the first GET icon and then click **Try it Out!**.
This displays a list of all books. Locate and copy an ISBN (for example: 0321802055).
10. Click on the second GET icon, which lets you get a book by its ISBN.

11. Add the ISBN (for example, 0321802055), and then click **Try it Out!**.
    You can see the book details.

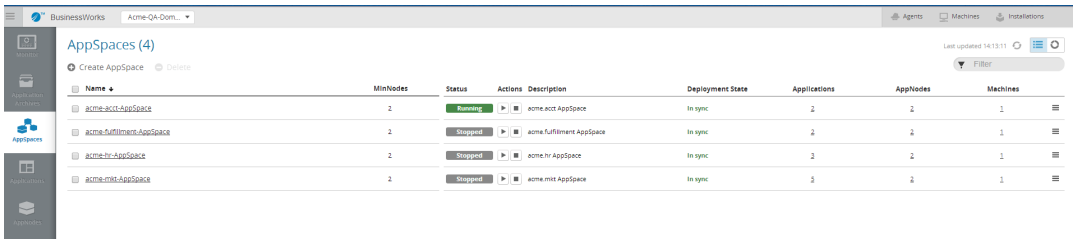


Defining and Deploying Multiple AppSpaces and AppNodes

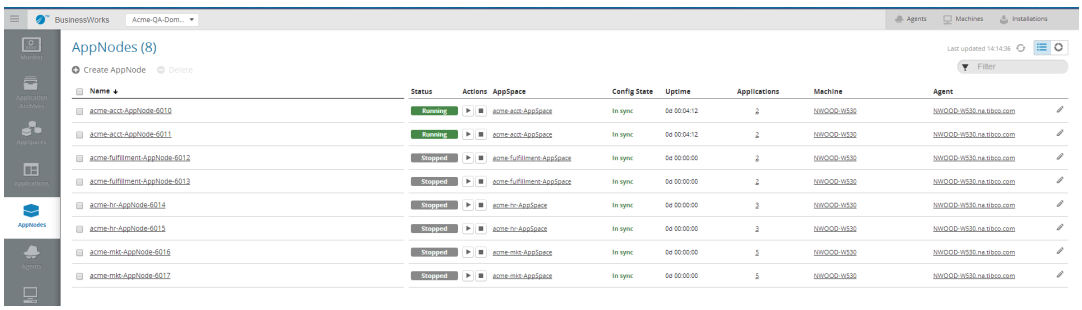
You can define multiple AppSpaces and AppNodes and deploy multiple applications to these AppNodes with the help of scripts.

Procedure

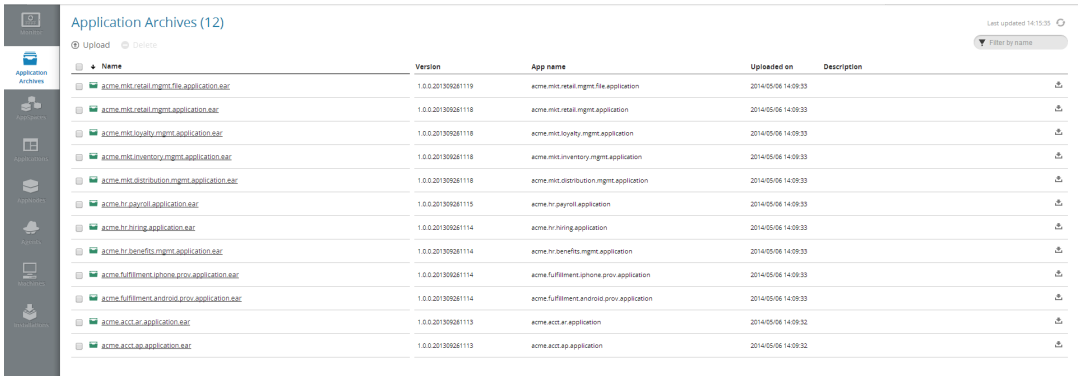
1. Run the **`./runAcme.sh`** command to deploy multiple AppSpaces, AppNodes and applications.
2. On the **Domain Management** page, click **Acme-QA-Domain**.
3. On the page, you will see that multiple AppSpaces have been created and running.



4. Now, view the AppNodes.



5. View the application archives.



6. In the Cygwin window, execute the **./killall.sh** command to shut down all running components.
7. Run the **./bwclean.sh** command to clean up.

Core Admin Sample Scripts

The sample scripts provide a simple and faster way to run the core Admin samples.

Script	Task Performed By the Script
bwagent.sh	Starts bwagent OS process.
killbwagent.sh	Kills bwagent OS process.
killbwappnodes.sh	Kills all bwappnode_* OS process. No script is required to start bwappnodes because bwadmin **start** command is used to start the bwappnodes.
bwadmin.sh	• A wrapper script to start bwadmin library. • Runs both in Interactive and Batch modes with **-f <cmd file>**
tea.sh	Starts TEA Server OS process.
killtea.sh	Kills TEA Server OS process.
tibemsd64.sh	Starts EMS Server (64-bit) OS process. TIBCO ActiveMatrix BusinessWorks 6.x does not require TIBCO Enterprise Message Service. However, some ActiveMatrix BusinessWorks 6.x samples use JMS.
killtibemsd64.sh	Kills EMS Server OS process.
killall.sh	Kills tea, bwagent, bwappnode, bwadmin, and asmm OS processes.
genbwagentini.sh	Generates `${BW_HOME}/config/bwagent.ini` file with config read from **config/bwadmin-default-config.sh**

Script	Task Performed By the Script
bwclean.sh	• Kills bwappnodes, bwagent, bwadmin and wipes clean ActiveMatrix BusinessWorks 6 Domain Datastore and calls **genbwagentini.sh** • Safeguards the above action by prompting user confirmation. • User confirmation can be enforced by using **-forceClean** command argument.
teaclean.sh	• Kills TEA Server and wipes clean TEA Server Datastore. • Safeguards the above action by prompting user confirmation. • User confirmation can be enforced by using **-forceClean** command argument.
registeragent.sh	• Scripted action to register bwagent to TEA Server. • Saves time from registering bwagent in TEA Web UI. • Assumes that both bwagent and tea are already running.
showprocs.sh	Detects whether the following OS processes are running: • tibemsd • tea • bwagent • bwappnode • bwadmin
bootstrap.sh	Bootstraps a clean ActiveMatrix BusinessWorks 6.x Admin environment for repetitive purpose. This is a wrapper script that does the following: • **teaclean.sh** • **bwclean.sh** • **tea.sh** • **bwagent.sh** • **registeragent.sh**
runBookStore.sh	Creates BookStore-Domain and deploys the ActiveMatrix BusinessWorks 6.x RESTful BookStore application.

Script	Task Performed By the Script
runSamples.sh	• Creates Samples-Domain and deploys all EARs found in ears/samples. • Deploys multiple ActiveMatrix BusinessWorks 6 Applications in the same AppSpace. • By default creates each AppSpace with a single AppNode.
runAcme.sh	• Creates and deploys EARs found in ears or acme sub-directories in two Domains. • By default, creates and deploys EARs in Acme-QA-Domain. • **./runAcme.sh -domain Acme-UAT-Domain** creates Acme-UAT-Domain and deploys the same EARs. • Creates each Appspace with 1 to 3 AppNodes. • Ideally shows multiple Applications per AppSpace and scaling of Applications on multiple AppNodes.
runAll.sh	Bootstraps ActiveMatrix BusinessWorks 6.x Admin environment and runs all of the following **run*.sh** scripts: • **bootstrap.sh** • **runBookStore.sh** • **runAcme.sh -domain Acme-QA-Domain** • **runAcme.sh -domain Acme-UAT-Domain** Creates multiple Domains, AppSpaces , AppNodes and deploys multiple Applications in batch mode.
deploy.sh	Deploys a single application. Creates an AppSpace and corresponding AppNode by convention, before deploying the EAR file.
AppManage.sh	• An emulation utility that provides exact AppManage command line arguments, and translates them into bwadmin commands. • Shows the ActiveMatrix BusinessWorks 5.x users, how the AppManage commands (familiar to ActiveMatrix BusinessWorks 5.x users) map to the corresponding bwadmin commands. The current emulation may not be 100% complete.

Each **runAcme.sh**, **runBookStore.sh**, **runSamples.sh**, **deploy.sh**, and **AppManage.sh** generates **bwadmin** commands before executing them.

The generated bwadmin command files are found under `cmd` sub-directory.