

ActuateOne™

One Design
One Server
One User Experience

Installing an Actuate Java Component

This documentation has been created for software version 11.0.5.

It is also valid for subsequent software versions as long as no new document version is shipped with the product or is published at <https://knowledge.opentext.com>.

Open Text Corporation

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111

Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440

Fax: +1-519-888-0677

Support: <https://support.opentext.com>

For more information, visit <https://www.opentext.com>

Copyright © 2017 Actuate. All Rights Reserved.

Trademarks owned by Actuate

“OpenText” is a trademark of Open Text.

Disclaimer

No Warranties and Limitation of Liability

Every effort has been made to ensure the accuracy of the features and techniques presented in this publication. However, Open Text Corporation and its affiliates accept no responsibility and offer no warranty whether expressed or implied, for the accuracy of this publication.

Document No. 170215-2-781510 February 15, 2017

Contents

Introduction	iii
Understanding ActuateOne	iii
About Actuate Java Component documentation	iii
Obtaining documentation	v
Using PDF documentation	vi
Obtaining late-breaking information and documentation updates	vi
Obtaining technical support	vi
About supported and obsolete products	vii
Typographical conventions	vii
Syntax conventions	vii
About <i>Installing an Actuate Java Component</i>	viii
 Chapter 1	
Before you begin	1
About Actuate Java Components	2
About deployment formats	2
Checking installation prerequisites	3
About the license key file	3
 Chapter 2	
Deploying a Java Component	5
Setting web application parameters	6
Configuring locale parameters	7
Configuring parameters for Deployment Kit	7
Configuring parameters for BIRT Viewer and BIRT Interactive Viewer	7
Configuring parameters for BIRT Studio	8
Compressing files and renaming the WAR file	8
Deploying the WAR file	9
Deploying Actuate Java Components EARs to JBoss 6	10
 Chapter 3	
Using a Java Component	11
Testing the installation	12
Using Java Components	12
Using BIRT Viewer and BIRT Interactive Viewer	12
Using BIRT Studio	13
Using the Deployment Kits	13
Licensing a Java Component in a WAR file	15
Setting JVM properties	16
 Index	19

Understanding ActuateOne

ActuateOne™ includes Release 11 of Actuate® Corporation's value-added products for the Eclipse BIRT open source project. ActuateOne institutes a paradigm shift in Business Intelligence technology from individualized tools to a suite of integrated capabilities within a single environment. ActuateOne is one designer, one server, one integrated environment providing a single unified user experience. A common architecture is precisely what today's information-rich global environment requires for development and deployment. This unified Actuate technology continues to enable information management and delivery while supporting advanced security, massive scalability, flexibility through programming, and reuse. ActuateOne realizes our corporate vision of a single user experience by providing extended new analytics capabilities reaching a broader spectrum of users. The new dashboard functionality supports building gadgets to enhance the visual presentation of information. Export to Excel® and other formats integrates Actuate output with other tools on the end-user desktop. Actuate's cloud-ready server supports elastic clustering for dynamic provision of uninterrupted efficient service.

Information, live demos, and endorsements about this release are available from birt-exchange.com and actuate.com. The Actuate site also makes "The Forrester Wave™: Open Source Business Intelligence (BI), Q3 2010" report freely available. The report recognizes Actuate and its value-added offerings for BIRT as a leader in Open Source Business Intelligence.

About Actuate Java Component documentation

The documentation includes the materials described in Table I-1. You can obtain HTML and PDF files from developer.actuate.com.

Documentation updates are created in response to customer requirements.

Table I-1 Product documentation

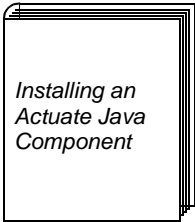
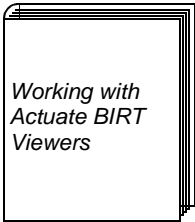

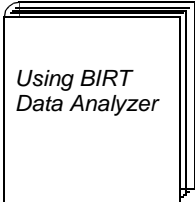
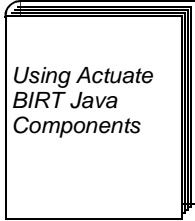

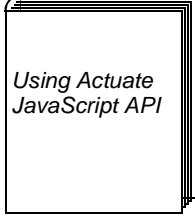
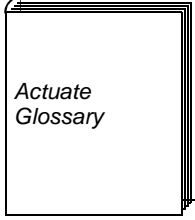
For information about this topic	See the following resource
Documentation updates	Updated HTML help and PDF files posted on developer.actuate.com .
Installing and configuring Actuate Java Components	 <p><i>Installing an Actuate Java Component</i></p>
Introduction to report viewers BIRT Viewer Navigating, printing and exporting to other formats BIRT Interactive Viewer Formatting reports, organizing data, working with charts, and hiding and filtering data	 <p><i>Working with Actuate BIRT Viewers</i></p>
Overview of BIRT Studio concepts Designing, editing, and formatting reports Organizing and filtering data, presenting data in a chart; inserting calculated data; functions and operators reference Creating and publishing report templates; configuring and customizing BIRT Studio pages	 <p><i>Using BIRT Studio - Java Component Edition</i></p>
Using BIRT Data Analyzer Organizing and formatting a cross tab Filtering cross tab data Working with data cubes Working with charts	 <p><i>Using BIRT Data Analyzer</i></p>

Table I-1 Product documentation

For information about this topic	See the following resource
Introduction to the Deployment Kit and Viewers, including concepts and online reporting Accessing files and folders; running reports	 <p><i>Using Actuate BIRT Java Components</i></p>
Overview of Java Component concepts and the web applications Using, customizing, and configuring Java Components Using the code components for JSPs, URL parameters, JavaScript files, Java servlets, Java Beans, and security facilities	 <p><i>Actuate Java Components Developer Guide</i></p>
Overview of concepts for programming with Actuate JavaScript Creating custom pages using Actuate JavaScript Reference of Actuate JavaScript classes and methods	 <p><i>Using Actuate JavaScript API</i></p>
Glossary	 <p><i>Actuate Glossary</i></p>

Obtaining documentation

Actuate provides technical documentation in PDF and HTML formats. Actuate products access HTML-format documentation from the Actuate web site. If you do not have web access or prefer to use documentation hosted on your local system, install the documentation from the Online Documentation and Localization Resource Files package. You can download PDF or view HTML versions of the documentation from developer.actuate.com. If you purchase the product, you can also download documentation from OpenText My Support. If

you select the typical setup when you install from the downloaded ActuateLocalizationandOnlineDocumentation.exe, the installation creates the ACTUATE_HOME\Manuals directory.

Using PDF documentation

In each book, the table of contents and the index page numbers contain links to the corresponding topics in the text. In the table of contents, you access the link by positioning the pointer over the topic. In the index, you access the link by positioning the pointer over the page number.

The ACTUATE_HOME\Manuals directory contains a file, master-index.pdx, which is an Adobe Acrobat Catalog utility that can search all the documents in the Actuate Manuals directory. This tool provides a convenient way to find a particular topic in Actuate documentation.

Obtaining late-breaking information and documentation updates

The release notes contain late-breaking news about Actuate products and features. The release notes are available on OpenText My Support at the following URL:

<https://support.opentext.com>

A new user must first register on the site and log in to view the release notes.

Updates to documentation in PDF form are available on OpenText My Support or at the following URL:

<http://www.developer.actuate.com>

Alternatively, after installing Actuate Release 11 products, use Start→Programs →Actuate 11→Update Documentation to access this URL.

Obtaining technical support

You can contact customer support by submitting a ticket. To submit a ticket, go to the following URL:

<https://support.opentext.com>

About supported and obsolete products

The Actuate Support Lifecycle Policy and Supported Products Matrix are available on OpenText My Support. You can access this site at the following URL:

<https://support.opentext.com>

Typographical conventions

Table I-2 describes the typographical conventions in this document.

Table I-2 Typographical conventions

Item	Convention	Example
Code examples	Monospace	<code>Dim Text1 As String</code>
File names	Initial capital letter, except where file names are case-sensitive	<code>Detail.roi</code>
Key combination	A + sign between keys means to press both keys at the same time	<code>Ctrl+Shift</code>
Menu items	Capitalized, no bold	<code>File</code>
Submenu items	Separated from the main menu item with a small arrow	<code>File→New</code>
User input or user response	Monospace	<code>M*16*</code>
User input in XML and Java code	Monospace italics	<code>chkjava.exe</code> <code>cab_name.cab</code>

Syntax conventions

Table I-3 describes the symbols used to present syntax.

Table I-3 Syntax conventions

Symbol	Description	Example
[]	Optional item	[Alias<alias name>]
	Array subscript	matrix[]

(continues)

Table I-3 Syntax conventions (continued)

Symbol	Description	Example
{ }	Groups two or more mutually exclusive options or arguments when used with a pipe	{While Until}
	Defines array contents	{0, 1, 2, 3}
	Delimiter of code block	public AJCDesigner() { }
	Separates mutually exclusive options or arguments in a group	Exit {Do For Function Sub}
	Java OR operator	int length 4
<>	Argument you must supply	<expression to format>
	Delimiter in XML	<xsd:sequence>

About *Installing an Actuate Java Component*

Installing an Actuate Java Component includes the following chapters:

- *Introduction*. Provides an overview of this guide, a summary of Actuate Java Components documentation, and a list of the typographical conventions in this book.
- *Chapter 1. Before you begin*. Explains Java Components capabilities, installation prerequisites, and licensing.
- *Chapter 2. Deploying a Java Component*. Covers editing parameters for using Java Components and deploying the Java Components war file.
- *Chapter 3. Using a Java Component*. Describes testing, using, and licensing a Java Component.

1

Before you begin

This chapter discusses the following topics:

- About Actuate Java Components
- About deployment formats
- Checking installation prerequisites
- About the license key file

About Actuate Java Components

Actuate Java Components provide a complete reporting technology consisting of a report designer, report distribution in files and folders, and viewing or interactive viewing of reports. The following list describes each component:

- BIRT Deployment Kit
A file and folder browser that optimizes performance and viewing of BIRT reports
- BIRT Interactive Viewer
A viewer that you can use to customize and personalize a BIRT report to answer ad hoc business questions
- BIRT Spreadsheet Deployment Kit
A file and folder browser that optimizes performance and viewing of BIRT Spreadsheet reports
- BIRT Studio
A web-based BIRT report designer for creating reports that can include charts and graphs
- BIRT Viewer
A viewer for BIRT reports that extends the capabilities of BIRT to deliver rich dynamic content and display aggregate data in cross tabs

About deployment formats

You can install Java Components on Windows, Linux, and UNIX. Actuate provides Java Components in WAR and EAR formats. These formats are suitable for deployment on application servers shown in Table 1-1.

Table 1-1 Java Component deployment formats

File	Application server
WL_TOMCAT_ ActuateBIRTJavaComponent.war	Apache Tomcat Jetty HTTP & Application Server Oracle WebLogic Application Server
JBOSS_Wildfly_ ActuateBIRTJavaComponent.ear	JBoss Enterprise Application Platform WildFly (JBoss Application Server)
WAS_ ActuateBIRTJavaComponent.ear	IBM WebSphere Application Server

For supported versions of each application server, see the Supported Products Matrix on OpenText My Support or at the following URL:

<http://developer.actuate.com/resources/supported-products/iserver/iserver-11sp5/>

Checking installation prerequisites

Before you deploy a Java Component, ensure that your system meets the following requirements:

- Java Development Kit (JDK)
You need to use the JAR utility included in the JDK to decompress and compress the Java Components archive. Java Components also require the Java Runtime Environment (JRE) included in the JDK. The Deployment Kits require the use of the local JVM and cannot differentiate multiple JVMs on the same machine. Therefore, you must install Deployment Kit with only one JVM. For supported versions of Java, see the Supported Products Matrix.
- An application server
Java Components run as a web application in the application server, such as Apache Tomcat 6.
- Administrators group privileges on Windows, or root privileges on Linux or UNIX
- Permission to start and stop the application server
- A Java Virtual Machine and Web Server configured for the same application platform, either 32-bit or 64-bit

About the license key file

You can download Java Components from OpenText My Support. Included in the download is an evaluation license for using each of the components for a limited time.

If you install products using the evaluation license, you do not need to reinstall before you activate the permanent license. To arrange for a permanent license key file, go to OpenText My Support at the following location:

<https://support.opentext.com> (Log in required)

Choose License Keys. Then, submit a ticket.

If you are a maintenance customer, you should have login information for OpenText My Support. If you do not have access, please contact your OpenText sales representative.

If you are not a direct OpenText customer, contact the partner or distributor who supplies the product for the license file. If you have a problem obtaining a license file from this source, please contact your OpenText sales representative.

The name for an Actuate Java Component license key file uses the following format:

```
Actuate_AJC_key_<ID>.xml
```

<ID> is a unique number generated by OpenText Licensing when it creates the license key file. You receive an e-mail that contains the permanent license key file information when OpenText processes your request.

If the license file does not exist, is invalid, or has expired, an error appears in the reportService log file, described later. When the license is invalid or expired, Java Components load, but any attempt to access an Actuate application, including the login page, results in an error. If you have a problem with a license key file, contact OpenText Support.

2

Deploying a Java Component

This chapter discusses the following topics:

- Setting web application parameters
- Compressing files and renaming the WAR file
- Deploying the WAR file

Setting web application parameters

The first step in deploying the Java Component WAR file is editing the web.xml file. Extract web.xml from the WAR file, then edit web.xml to set configuration parameters for one or more components that you purchased or are evaluating. The tables in this section describe these parameters and actions you take. Perform these actions as you edit web.xml.

How to edit web.xml

To access web.xml for editing, first decompress the WAR file that contains web.xml. Next, configure parameters for the Java Components that you want to use.

- 1 Create a directory named /temp. Java Components writes temporary files to this directory after deployment.
- 2 Decompress the ActuateJavaComponent.war file in a subdirectory of /temp. Type the following command:

```
jar -xf ActuateJavaComponent.war
```

Use this command on Windows, Linux, and UNIX. The Java Component files appear in the subdirectory of /temp.

Figure 2-1 shows these files on Linux.

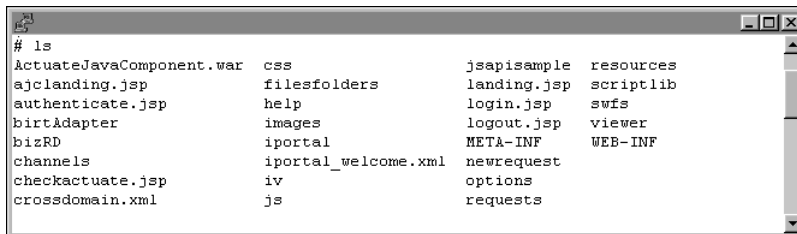


Figure 2-1 Files extracted from ActuateJavaComponent.war

Keep all downloaded files together in their original relative locations.

- 3 Using a text editor that accepts UTF-8 encoding, edit web.xml in /WEB-INF to configure the parameters for the Java Components you purchased or are evaluating.
- 4 Save and close web.xml.

Configuring locale parameters

The locale parameters, described in Table 2-1, are applicable to all Java Components.

Table 2-1 Locale parameters

Parameter name	Description	Action
DEFAULT_LOCALE	The value of this parameter determines the default locale. The default locale is en_US. Users can select a locale when they log in.	Use the default locale or a value from Localemap.xml in the WEB-INF directory.
DEFAULT_TIMEZONE	The value of this parameter determines the default time zone. The default time zone is Pacific Standard Time (PST).	Use the default time zone or a value from Timezonemap.xml in the WEB-INF directory.

Configuring parameters for Deployment Kit

The components for BIRT Deployment Kit and BIRT Spreadsheet Deployment Kit have additional core configuration parameters in web.xml. Table 2-2 describes these parameters. Update these parameters before deployment.

For BIRT Deployment Kit, also review the parameters described in Table 2-3.

Table 2-2 Deployment Kit parameters

Parameter name	Description	Action
LOG_FILE_LOCATION	The location of the files that log Deployment Kit activities.	You must create this physical location on the file system. The default value is /logs.
TEMP_FOLDER_LOCATION	The location where Deployment Kit creates temporary files.	You must create this physical location on the file system. There is no default value.
TRANSIENT_STORE_PATH	Another location where Deployment Kit creates temporary files.	You must create this physical location on the file system. The default value is /temp.

Configuring parameters for BIRT Viewer and BIRT Interactive Viewer

The BIRT Viewer and BIRT Interactive Viewer components also have core configuration parameters in web.xml. Table 2-3 describes these configuration parameters. Consider changing these parameters to optimize performance.

Table 2-3 BIRT Viewer parameters

Parameter name	Description	Action
BIRT_ARCHIVE_MEMORY_TOTALSIZE	Total memory available for BIRT report document files in kilobytes. The default value is 50 MB.	None required
REPOSITORY_CACHE_TIMEOUT_SEC	Specifies, in seconds, how long to retain temporary files that BIRT Viewer creates when a user modifies the appearance of a report. The default value is 900, which is 15 minutes.	None required

Configuring parameters for BIRT Studio

BIRT Studio component has one configuration parameter, shown in Table 2-4. Edit the parameter to change the path to BIRT resources. BIRT resources include libraries and templates for Actuate BIRT report designs and Actuate BIRT Studio.

Table 2-4 BIRT Studio parameter

Parameter name	Description	Action
BIRT_RESOURCE_PATH	Path to Actuate BIRT shared resources. The default value is the following directory: <pre><Tomcat install directory> /webapps/newJavaComponent /resources</pre>	None required

Compressing files and renaming the WAR file

After editing the web.xml file, compress and rename the archive. For example, rename the archive newJavaComponent.war. Compress the files into a WAR file of a different name to preserve your original WAR file.

How to compress files and rename the WAR file

- 1 After editing web.xml in the WEB-INF directory, move up one directory.
- 2 Check that files shown in Figure 2-1 are present in your current directory.
- 3 Compress the files and specify a new name for the WAR file. Type the following command:

```
jar -cf newJavaComponent.war *
```

Use this command on Windows, Linux, and UNIX. The WAR file containing the modified configuration values in web.xml appears.

Deploying the WAR file

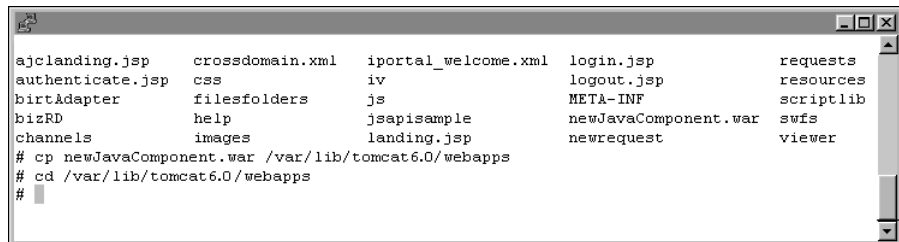
This section discusses deploying a Java Component WAR file. For example purposes, deployment on Tomcat 6 is shown.

To deploy the WAR file on an application server other than Tomcat 6, use your application server's deployment tools. If your application server does not have deployment tools, add the Java Components context root to your application server. Typically, you modify the application server's configuration file.

Perform the steps in the following procedure to deploy the WAR file by putting it in the /webapps directory.

Deploying the WAR file on Apache Tomcat

- 1 Stop the Apache Tomcat service.
- 2 Copy the renamed Java Component WAR file from the working directory to the /webapps directory of your Tomcat installation files, as shown in Figure 2-2.



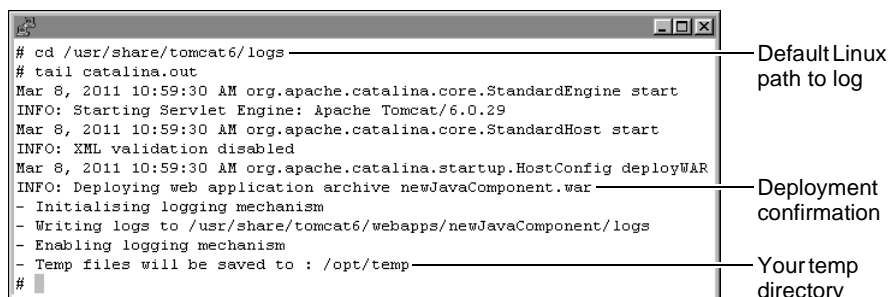
```
ajclanding.jsp    crossdomain.xml  iportal_welcome.xml  login.jsp          requests
authenticate.jsp  css              iv                   logout.jsp         resources
birtAdapter       filefolders     js                   META-INF          scriptlib
bizRD             help            jsapisample         newJavaComponent.war  swfs
channels          images          landing.jsp         newrequest         viewer
# cp newJavaComponent.war /var/lib/tomcat6.0/webapps
# cd /var/lib/tomcat6.0/webapps
#
```

Figure 2-2 Copying the WAR file to /webapps on Linux

- 3 Start Tomcat.

The Tomcat log confirms the deployment of the WAR file and the location of the /temp directory that you created earlier.

Figure 2-3 shows an example of a Tomcat log file on Linux.



```
# cd /usr/share/tomcat6/logs
# tail catalina.out
Mar 8, 2011 10:59:30 AM org.apache.catalina.core.StandardEngine start
INFO: Starting Servlet Engine: Apache Tomcat/6.0.29
Mar 8, 2011 10:59:30 AM org.apache.catalina.core.StandardHost start
INFO: XML validation disabled
Mar 8, 2011 10:59:30 AM org.apache.catalina.startup.HostConfig deployWAR
INFO: Deploying web application archive newJavaComponent.war
- Initialising logging mechanism
- Writing logs to /usr/share/tomcat6/webapps/newJavaComponent/logs
- Enabling logging mechanism
- Temp files will be saved to : /opt/temp
#
```

Annotations:

- Default Linux path to log
- Deployment confirmation
- Your temp directory

Figure 2-3 Tomcat log file confirms deployment

Deploying Actuate Java Components EARs to JBoss 6

In order to deploy Actuate BIRT Java Component EARs to JBoss 6, you must remove the commons-logging.jar library from the EAR file before deploying it. Leaving this file in the EAR causes JBoss 6 to throw an exception during deployment.

The commons-logging.jar file is still necessary for deployments to JBoss 5 and JBoss 7.

To deploy Java Components to JBoss 6, perform the following steps:

- 1 In a separate directory, extract the contents of JBOSS_Wildfly_ActuateBIRTJavaComponent.ear by typing the following command into a command prompt:

```
jar -xf JBOSS_Wildfly_ActuateBIRTJavaComponent.ear
```

Keep all extracted files together in their original relative locations.

- 2 Copy JBOSS_Wildfly_ActuateBIRTJavaComponent.war into a separate directory and extract the contents of JBOSS_Wildfly_ActuateBIRTJavaComponent.war by typing the following command into a command prompt:

```
jar -xf JBOSS_Wildfly_ActuateBIRTJavaComponent.war
```

- 3 Delete <warFile>/WEB-INF/lib/commons-logging.jar.
- 4 Compress the all files in the directory into a war file by typing the following command:

```
jar -cf JBOSS_Wildfly_ActuateBIRTJavaComponent.war *
```

- 5 Copy JBOSS_Wildfly_ActuateBIRTJavaComponent.war into the separate directory containing the extracted files from JBOSS_Wildfly_ActuateBIRTJavaComponent.ear. Compress these files into a ear file by typing the following command:

```
jar -cf JBOSS_Wildfly_ActuateBIRTJavaComponent.ear *
```

- 6 Deploy JBOSS_Wildfly_ActuateBIRTJavaComponent.ear to the JBoss 6 server.

Using a Java Component

This chapter discusses the following topics:

- Testing the installation
- Using Java Components
- Licensing a Java Component in a WAR file
- Setting JVM properties

Testing the installation

To test the Java Component install, open the Java Components landing page. Go to the following URL to access the landing page shown in Figure 3-1:

```
http://host:8080/newJavaComponent/ajclanding.jsp?
userid=anonymous
```

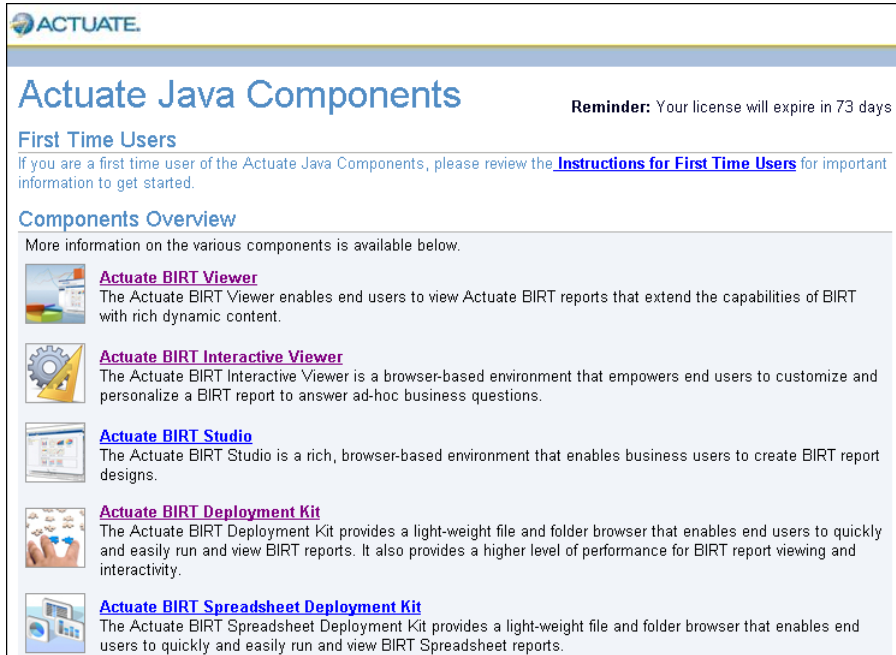


Figure 3-1 Actuate Java Components landing page

Using Java Components

If you are evaluating Java Components, use all products for the evaluation period. If you purchased particular Java Components, you can use only those products.

Using BIRT Viewer and BIRT Interactive Viewer

BIRT Viewer is the default viewer that BIRT uses when you open a report. To use BIRT Interactive Viewer, open a report, and choose Enable Interactivity.

Figure 3-2 shows Enable Interactivity.

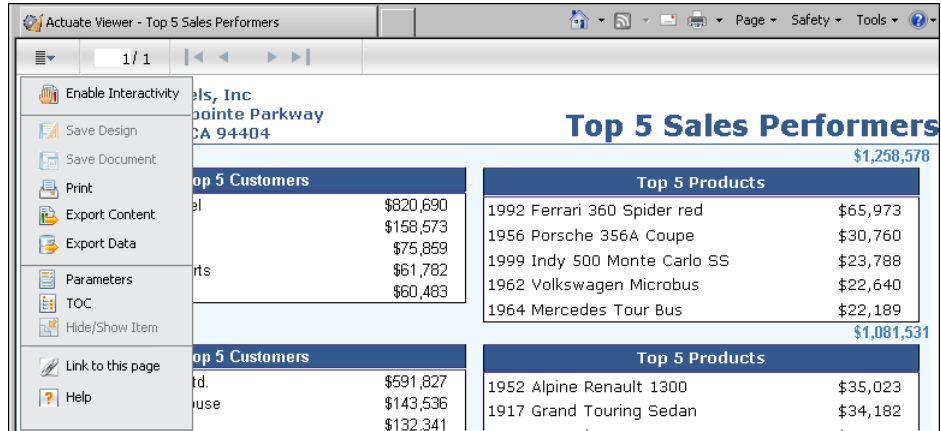


Figure 3-2 Choosing Enable Interactivity

Using BIRT Studio

Launch BIRT Studio by going to the following URL:

<http://host:8080/newJavaComponent/wr?&iportal>

Figure 3-3 shows the BIRT Studio opening page.

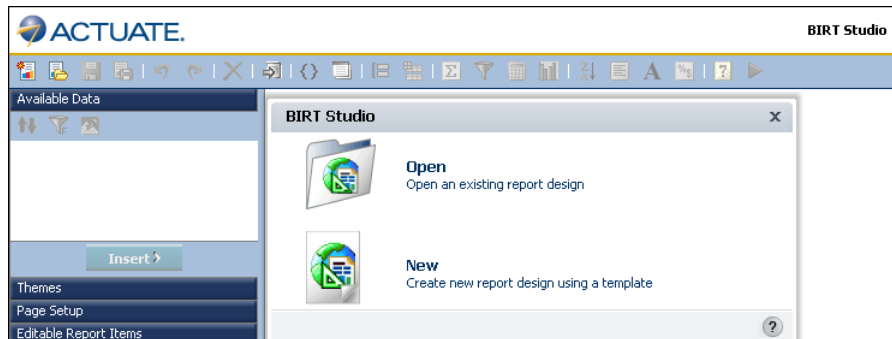


Figure 3-3 Launching the BIRT Studio Component

Using the Deployment Kits

Launch BIRT Deployment Kit and BIRT Spreadsheet Deployment Kit from the Actuate Java Components landing page.

How to launch BIRT Deployment Kit

- 1 On the landing page, choose Actuate BIRT Deployment Kit, as shown in Figure 3-4.



Figure 3-4 Actuate Java Components landing page

The Actuate BIRT Deployment Kit appears, as shown in Figure 3-5.

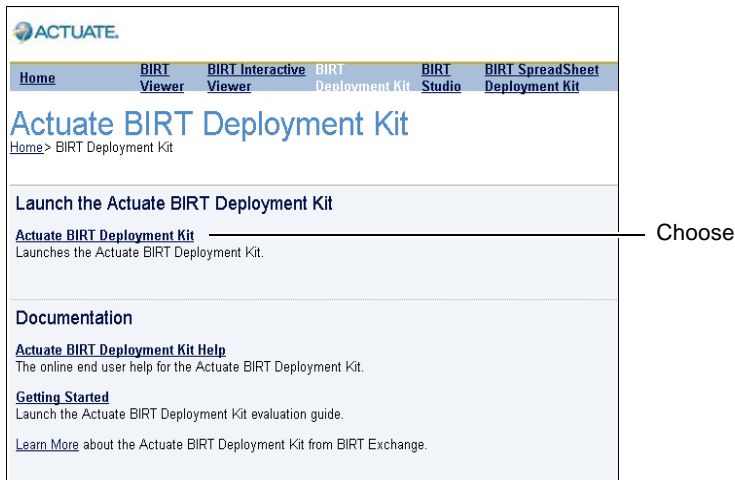


Figure 3-5 Launching the Actuate BIRT Deployment

- 2 Launch the Actuate BIRT Deployment Kit. Figure 3-6 shows the folders of report documents that you can browse.

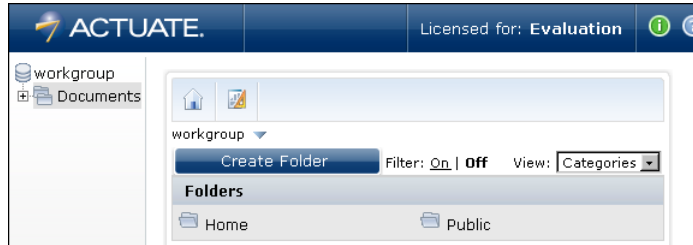


Figure 3-6 Accessing folders of report documents

Licensing a Java Component in a WAR file

Included in the download is an evaluation license for using each of the components for a limited time. After installation, check the evaluation license expiration date by opening the reportService log in the following directory:

`<Tomcat install directory>/webapps/newJavaComponent/logs`

Figure 3-7 shows the reportService log on Windows.

```

Supported options
License expiration
Jun 27, 2012 3:23:52 PM com.actuate.ipoortal.license.LicenseManager initialize
SEVERE: ***** Actuate Java Component License Information *****
License Type: CPUBased
Usage Type: Evaluation
License ID: 3
IsDevelopment: true
IsNodeLocked: false
CPU Limit: unlimited
Major Release: 11
ServicePack: 5
Mode: workgroup
Expiration Date: Sun Sep 09 00:00:00 PDT 2012

Supported Options: Spread BIRTReport BIRTInteractiveviewer BIRTReportStudio DeploymentKit
BIRTDataAnalyzer
Non-Supported Options:
Jun 27, 2012 3:23:52 PM com.actuate.reportcast.servlets.StartupServlet startJREM

```

Figure 3-7 Checking the license expiration date and supported options

When you complete the evaluation, replace the evaluation license with a permanent license. After obtaining a new license key file, as explained earlier, perform the steps in the following procedure:

How to license a Java Component

- 1 Rename the Java Component license file that Actuate sent you to `ajcllicense.xml`.
- 2 Decompress the Java Component WAR file.

```
jar -xf ActuateJavaComponent.war
```

The Java Component files appear.

- 3 Copy the `ajcllicense.xml` file into the extracted `<context root>\WEB-INF` directory. For example, assuming you named the file `newJavaComponent.war`, the path to access the `WEB-INF` directory is:

```
<Tomcat install directory>/webapps/newJavaComponent/WEB-INF
```

- 4 Assuming `WEB-INF` is your current directory, move up one directory and type the following command on Windows, Linux, or UNIX:

```
jar -cf MyJavaComponent.war *
```

This command creates `MyJavaComponent.war`. This WAR file contains the license. The WAR file now has the context root `MyJavaComponent`.

- 5 Deploy `MyJavaComponent.war` to the application server or servlet engine as an application, as described earlier.
- 6 Restart the application server or servlet engine.

Setting JVM properties

Deploying Actuate Java Components within some application server environments requires Java configuration to avoid Java Virtual Machine (JVM) errors. Set the following JVM properties:

- Initial heap size
Include `-Xms64m` as a Java option.
- Maximum heap size
Include `-Xmx512m` as a Java option.
- Maximum size for the permanent generation heap
Include `-XX:MaxPermSize=128m` as an option.

How to configure JVM properties for Tomcat 5.x on Windows

- 1 Navigate to the following directory:

```
<Tomcat install directory>\bin
```
- 2 Open `catalina.bat`.
- 3 Add `-Xms64m -Xmx512m` and `-XX:MaxPermSize=128m` to the `JAVA_OPTS` variable specification.
- 4 Save and close `catalina.bat`.
- 5 Restart Tomcat.

How to configure JVM properties for Tomcat 6.x on Windows

- 1 From the Windows Start menu, choose All Programs→Apache Tomcat 6.x →Configure Tomcat.
- 2 On Apache Tomcat 6.x Properties, choose Java.
- 3 Change the value of the Initial memory pool to 64.
- 4 Change the value of the Maximum memory pool to 512.
- 5 Add -XX:MaxPermSize=128m to Java Options, as shown in Figure 3-8.

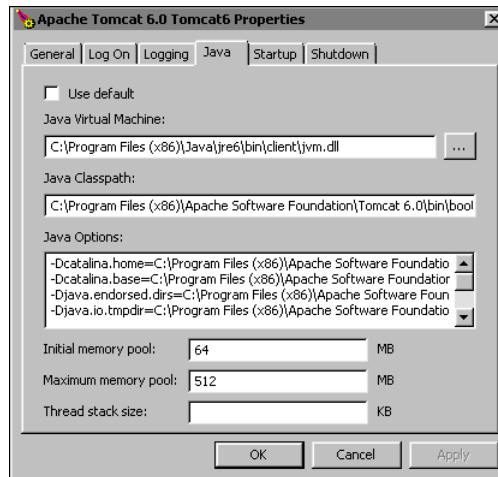


Figure 3-8 Apache Tomcat 6 Java configuration

- 6 Choose OK.
- 7 Restart Tomcat.

How to configure JVM properties for JBoss on Windows

- 1 Navigate to the following directory:
`<JBoss install directory>\bin`
- 2 Open run.bat.
- 3 Add -Xms64m -Xmx512m and -XX:MaxPermSize=128m to the JAVA_OPTS variable specification.
- 4 Save and close run.bat.
- 5 Restart Tomcat.

How to configure JVM properties for Tomcat on Linux

- 1 Navigate to the following directory:

```
/etc/tomcat6
```

- 2 Open tomcat6.conf.
- 3 Add `-XMs64m -Xmx512m` and `-XX:MaxPermSize=128m` to the `JAVA_OPTS` variable specification.
- 4 Save and close tomcat6.conf.
- 5 Restart Tomcat.

Index

A

- accessing
 - customer support vi
- accessing online documentation vi
- activity logs 7
- ActuateJavaComponent.ear 2
- ActuateLocalizationandOnlineDocumentation.exe vi
- Adobe Acrobat Catalog vi
- ajclicense.xml file 16
- Apache Tomcat Configuration 16, 17
- application servers. *See* servers

B

- BIRT Interactive Viewer 2, 7
- BIRT reports 8
- BIRT Studio 2, 8
- BIRT Viewer 2, 7
- BIRT_ARCHIVE_MEMORY_TOTALSIZE parameter 8
- BIRT_RESOURCE_PATH parameter 8

C

- cache 8
- changing
 - default locales 7
 - time zones 7
- configuration parameters 7
- configuring
 - application servers 16
 - Java Components 7–8
- context root 16
- context roots 9
- current release iv
- Customer Care 4
- customer licenses 4
- customer support vi
- customizing WAR files 9

D

- Data Analyzer iv

- decompressing WAR files 15
- DEFAULT_LOCALE parameter 7
- DEFAULT_TIMEZONE parameter 7
- deploying Java Components 2, 3, 9
- Deployment Kit for BIRT Reports 2, 7
- Deployment Kit for Spreadsheet Reports 2, 7
- deployment tools 9
- designs 8
- document files 8
- documentation iii, vii
 - obtaining v
- downloading
 - documentation v

E

- editors 6
- expired license keys 4

F

- file names 16
- files 15

H

- HTML documentation. *See* documentation

I

- index (online documentation) vi
- installation guide viii
- installing Java Components 12
- invalid license keys 4

J

- jar utility 16
- Java Components
 - configuring 7–8, 16–18
 - deploying 2, 3
 - installing 12
 - obtaining licenses for 3
- Java Virtual Machine Configuration 16
- JBoss Configuration 17

L

- libraries 8
- license key files 4
- license keys 3
- licensing support (Actuate) 4
- links vi
- Localemap.xml 7
- locales 7
- log files 7
- LOG_FILE_LOCATION parameter 7

M

- maintenance customers 4
- manuals. *See* documentation
- Manuals directory vi
- master index
 - finding vi
- memory 8

N

- naming WAR files 16

O

- online documentation, downloading v
- online documentation. *See* documentation

P

- parameters. *See* configuration parameters
- PDF documentation v
 - See also* documentation

R

- release notes iv, vi
- report designs 8
- report document files 8
- reporting applications. *See* applications

- reports 8
- REPOSITORY_CACHE_TIMEOUT_SEC
 - parameter 8
- rich internet applications
 - See also* applications

S

- servers
 - configuring 16
 - deploying Java Components to 9
 - setting context root for 16
- servlet engines 16
- Support Lifecycle Policy vii
- Supported Products Matrix vii

T

- technical support vi
- TEMP_FOLDER_LOCATION parameter 7
- templates 8
- temporary files 7, 8
- testing Java Components installations 12
- text editors 6
- time zones 7
- Timezonemap.xml 7
- TRANSIENT_STORE_PATH parameter 7

U

- URLs 13, 16

W

- WAR files 9
- web applications. *See* applications
- web archive files. *See* WAR files
- web servers. *See* servers
- web.xml 6
- websphere61_ActuateJavaComponent.war 2