



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# eServices Deployment Guide

Connecting Tomcat and its Web Server

12/25/2025

# Connecting Tomcat and its Web Server

This section describes connecting Tomcat and its web server on the Windows platform.

## Configure the Connectors

### Configuring the Connectors

## Installing the JK1 Connector with Apache

### Prerequisites

- Apache web server is installed.
  - Tomcat is installed.
1. On the Interaction Management CD, navigate to ThirdPartyApplications\Tomcat\jk\_apache\_for\_windows\binary and locate the file that matches your version of Apache:
    - mod\_jk-1.2.28-httpd-2.2.3.so (for Apache 2.2 or later)
    - mod\_jk-1.2.28-httpd-2.0.52.so (for Apache 2.0.52 or later)
  2. Copy this file to your <APACHE\_HOME>/modules directory, and rename it to mod\_jk.so.
  3. In the <CATALINA\_HOME>/conf/jk directory, create a file called workers.properties, with the following contents:

```
worker.list=ajp13
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
```

### Important

If there is no jk directory, create one.

4. In the <APACHE\_HOME>/conf directory, create a file called mod\_jk.conf, with the following contents:

```
<IfModule !mod_jk.c>
  LoadModule jk_module "<APACHE_HOME>/modules/mod_jk.so"
</IfModule>
JkWorkersFile "<CATALINA_HOME>/conf/jk/workers.properties"
JkLogFile "<CATALINA_HOME>/logs/mod_jk.log"
JkLogLevel emerg
JkMount /examples/* ajp13
```

5. Add the following statement to the end of the <APACHE\_HOME>/conf/httpd.conf file:

```
Include "<APACHE_HOME>/conf/mod_jk.conf"
```

## Testing the JK1 Connector with Apache

### Prerequisites

- JK1 connector is installed.
1. Start Tomcat and Apache. Open a web browser and enter the following address: <http://localhost/examples/jsp/jsp2/el/basic-arithmetic.jsp>  
You should see the following web page:

### JSP 2.0 Expression Language - Basic Arithmetic

This example illustrates basic Expression Language arithmetic. Addition (+), subtraction (-), multiplication (\*), division (/ or div), and modulus (%) or mod) are all supported. Error conditions, like division by zero, are handled gracefully.

EL Expression	Result
\$(1)	1
\$(1 + 2)	3
\$(1.2 + 2.3)	3.5
\$(1.2E4 + 1.4)	12001.4
\$(-4 - 2)	-6
\$(21 * 2)	42
\$(3/4)	0.75
\$(3 div 4)	0.75
\$(3/0)	Infinity
\$(10%4)	2
\$(10 mod 4)	2
\$\$\$((1==2) ? 3 : 4)	4

## Installing the JK1 Connector with IIS 6

### Important

Make sure that IIS 6 Management Compatibility is installed on the host with IIS 7 (go to **Server Manager > Roles > Web Server (IIS) > Management Tools** and install "IIS 6 Management Compatibility" if it is not already installed).

### Prerequisites

- Tomcat is installed.
1. On the Interaction Management CD, go to ThirdPartyApplications\Tomcat\jk\_apache\_for\_windows\binary, locate isapi\_redirect<release number>.dll, and place it in c:\jakarta-tomcat\bin.
  2. Create the properties files workers.properties and uriworkermap.properties in c:\jakarta-tomcat\conf.  
Here is a minimum workers.properties example, using just ajp13 to connect your IIS web server to the Tomcat engine.

```
# workers.properties
#
# This file provides minimal jk configuration properties needed to
# connect to Tomcat.
# Define 1 real worker using ajp13
#
worker.list=ajp13w
#
# Set properties for worker ajp13w
# Note that the name and the type do not have to match.
#
worker.ajp13w.type=ajp13
worker.ajp13w.host=localhost
worker.ajp13w.port=8009
```

Here is a minimum example of uriworkermap.properties:

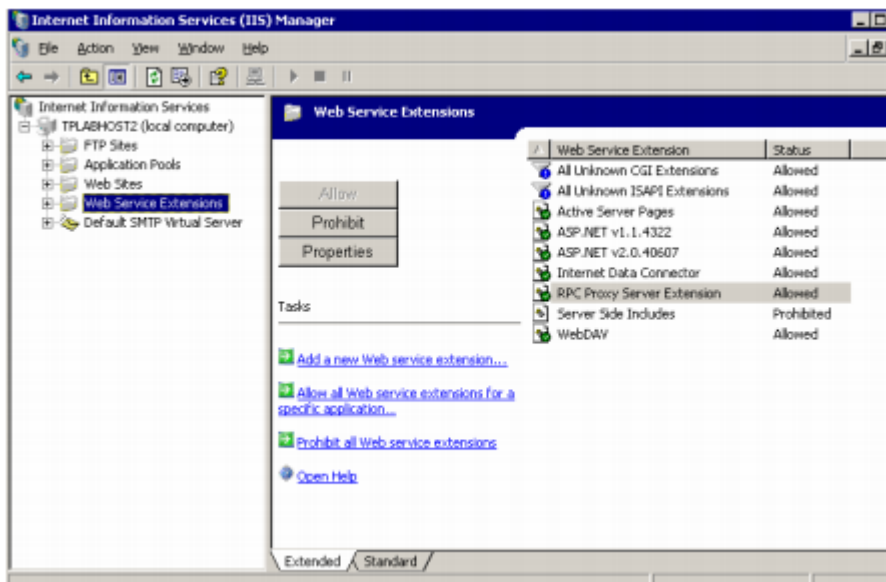
```
# This file provides sample mappings for example
# ajp13w worker defined in workermap.properties
/examples/*=ajp13w
```

### Important

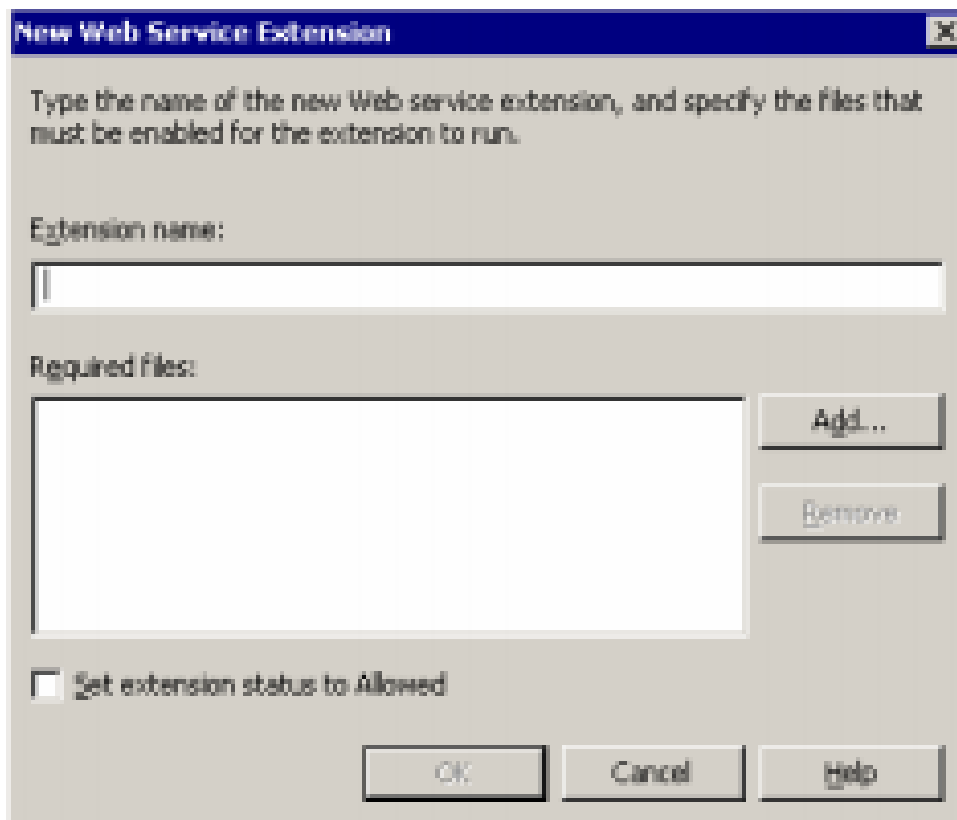
The workers.properties and uriworkermap.properties files must be saved in Microsoft Windows/MS-DOS format. Otherwise, an error will occur when installing the Web API Server and setup will not proceed.

3. In the registry, create a new registry key named: HKEY\_LOCAL\_MACHINE\SOFTWARE\Apache Software Foundation\Jakarta Isapi Redirector\1.0

4. Add a string value with the name `extension_uri` and a value of `/jakarta/isapi_redirect.dll`.
5. Add a string value with the name `log_file` and a value pointing to where you want your log file to be (for example `c:\jakarta-tomcat\logs\isapi.log`).
6. Add a string value with the name `log_level` and a value for your log level (can be debug, info, error or emerg).
7. Add a string value with the name `worker_file` and a value which is the full path to your `workers.properties` file (for example, `c:\jakarta-tomcat\conf\workers.properties`).
8. Add a string value with the name `worker_mount_file` and a value which is the full path to your `uriworkermap.properties` file (for example `c:\jakarta-tomcat\conf\uriworkermap.properties`).
9. Using the IIS management console, add a new virtual directory to your IIS web site. The name of the virtual directory must be `jakarta`. Its physical path should be the directory where you placed `isapi_redirect.dll` (in our example it is `c:\jakarta-tomcat\bin`). While creating this new virtual directory assign it with execute access.
10. Using the IIS management console, add `isapi_redirect.dll` as a filter in your IIS web site. The name of the filter should reflect its task (for example, `tomcat`), its executable must be our `c:\jakarta-tomcat\bin\isapi_redirect.dll`.
11. Open IIS Manager:



12. Click **Add New Web Service Extension**. The resulting dialog box is shown:



13. In the **New Web Service Extension** dialog box:
  - In the **Extension** name box, type jakarta.
  - In the **Required files** box, click **Add** and browse to the location of the isapi\_redirect.dll file.
  - Select the **Set extension status to Allowed** check box, then click **OK**.
14. Restart the IIS service. Ensure that the filter you added in Step 10 is marked with a green up-pointing arrow.

## Installing the JK1 Connector with IIS7

### Prerequisites

- Tomcat is installed.
1. On the Interaction Management CD, go to ThirdPartyApplications\Tomcat\jk\_apache\_for\_windows\binary, locate isapi\_redirect<release number>.dll, and place it in c:\jakarta-tomcat\bin.
  2. Create the properties files workers.properties and uriworkermap.properties in c:\jakarta-tomcat\conf.  
Here is a minimum workers.properties example, using just ajp13 to connect your IIS web server to

the Tomcat engine.

```
# workers.properties
#
# This file provides minimal jk configuration properties needed to
# connect to Tomcat.
# Define 1 real worker using ajp13
#
worker.list=ajp13w
#
# Set properties for worker ajp13w
# Note that the name and the type do not have to match.
#
worker.ajp13w.type=ajp13
worker.ajp13w.host=localhost
worker.ajp13w.port=8009
```

Here is a minimum example of uriworkermap.properties:

```
# This file provides sample mappings for example
# ajp13w worker defined in workermap.properties
/examples/*=ajp13w
```

### Important

The `workers.properties` and `uriworkermap.properties` files must be saved in Microsoft Windows/MS-DOS format. Otherwise, an error will occur when installing the Web API Server and setup will not proceed.

3. In the registry, create a new registry key named: `HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software Foundation\Jakarta Isapi Redirector\1.0`
4. Add a string value with the name `extension_uri` and a value of `/jakarta/isapi_redirect.dll`.
5. Add a string value with the name `log_file` and a value pointing to where you want your log file to be (for example `c:\jakarta-tomcat\logs\isapi.log`).
6. Add a string value with the name `log_level` and a value for your log level (can be `debug`, `info`, `error` or `emerg`).
7. Add a string value with the name `worker_file` and a value which is the full path to your `workers.properties` file (for example, `c:\jakarta-tomcat\conf\workers.properties`).
8. Add a string value with the name `worker_mount_file` and a value which is the full path to your `uriworkermap.properties` file (for example `c:\jakarta-tomcat\conf\uriworkermap.properties`).
9. Open IIS 7 Manager.
10. Navigate to your host.
11. Double click on the **ISAPI and CGI Restrictions** icon.
12. On the right hand panel, click **Add...**
13. Point the path to your `isapi_redirect.dll` file and give it a description, for example, `jakarta`.
14. Check **Allow extension path to execute**.

15. Click on the **Default Website** and navigate to **ISAPI Filters**.
16. On the right hand panel, click on **Add...** give the filter a name (for example, jakarta), and point to your `isapi_redirect.dll` file.
17. Navigate to your **Default Website** again.
18. Add a virtual directory and name it jakarta. Point this to the directory containing `isapi_redirect.dll`.
19. Click on the newly created virtual directory jakarta.
20. Navigate into **Handler Mappings**, on the right hand side there is a panel called **Actions**. Click on **Edit Feature Permissions** and click the execute permission.
21. Navigate to your host context again and do a restart.

### Important

If you are running a 32 bit version of Tomcat with a 32 bit JK connector on a x64 OS, you must set the **Application Pool -> Advanced Settings -> Enable 32 bit Applications** option to TRUE.

## Testing the JK1 Connector with IIS

### Prerequisites

- The JK1 connector is installed.
1. Start Tomcat. Start or restart IIS.
  2. Open a web browser and go to: `http://localhost/examples/jsp/jsp2/el/basic-arithmetic.jsp`. You should see the following web page:



## JSP 2.0 Expression Language - Basic Arithmetic

This example illustrates basic Expression Language arithmetic. Addition (+), subtraction (-), multiplication (\*), division (/ or div), and modulus (%) or mod) are all supported. Error conditions, like division by zero, are handled gracefully.

EL Expression	Result
<code>\$(1)</code>	1
<code>\$(1 + 2)</code>	3
<code>\$(1.2 + 2.3)</code>	3.5
<code>\$(1.2E4 + 1.4)</code>	12001.4
<code>\$(-4 - 2)</code>	-6
<code>\$(21 * 2)</code>	42
<code>\$(3/4)</code>	0.75
<code>\$(3 div 4)</code>	0.75
<code>\$(3/0)</code>	Infinity
<code>\$(10%4)</code>	2
<code>\$(10 mod 4)</code>	2
<code>\$( ((1==2) ? 3 : 4) )</code>	4

## Configure the Tomcat Context

## Configuring the Tomcat Context

To perform additional advanced configuration of your IIS-Tomcat implementation, see the following steps:

## Configuring IIS to Server Static Content

Genesys eServices web applications (Web API Server) use a combination of static web pages and active java content. You can reduce processing time by letting IIS serve static pages, even if these files are part of a context already served by Tomcat. For example, the .html and .gif files in the examples context do not need to be served from the Tomcat process—it is enough for IIS to serve them.

1. Configure IIS to include the Tomcat contexts. Adding a Tomcat context to IIS requires the addition of a new IIS virtual directory that covers the Tomcat context. For example, you could add a /example IIS virtual directory that covers the D:\tomcat\webapps\examples directory.
2. Configure the redirector to leave the static files for IIS. To configure the redirector, you must edit the uriworkermap.properties file to specify the URL-path pattern(s) that you want Tomcat to handle (usually only JSPs and servlets). For the examples context, you must replace the following line:

```
/examples/*=ajp13
```

with the following two lines:  
`/examples/*.jsp=ajp13`  
`/examples/servlet/*=ajp13`

The second line is more explicit; it actually instructs the redirector to redirect only the following:

- Requests to resources under `/examples/servlet/`.
- Resources under `/examples/` whose name ends with `.jsp`.

You can be more explicit and provide lines such as the following:

`/examples/servlet/<servletname>=ajp13`  
This instructs the redirector to redirect requests whose URL path equals

`/examples/servlet/<servletname>` to the worker named `ajp13`.

## Protecting the WEB-INF Directory

Each servlet application (context) has a special directory named `WEB-INF`. This directory contains sensitive configuration data and Java classes, and you must prevent web users from accessing or even seeing these directories. Although you can use the IIS management console to do this, the administrator must remember that it has been done. To avoid this complication, the redirector plugin automatically protects your `WEB-INF` directories by rejecting any request that contains `WEB-INF` in its URL-path.

- You do not need to perform any actions to obtain this `WEB-INF` protection. The redirector plugin does this automatically.

## Configuring Tomcat to Server Multiple Contexts

For Tomcat to serve multiple contexts, you must:

- Define the workers (the Tomcat processes or instances) that you want to use.
- Assign each context its own worker.

1. Open the `workers.properties` file, located in the `<CATALINA_HOME>/conf` directory.
2. In the `worker.list` entry, add workers using the following format:  
`worker.list=<a comma separated list of worker names>`  
Example  
`worker.list=ajp13, ajp13second`
3. Add entries for each individual worker, one for each associated host and port, using the following format:  
`worker.<worker name>.<property>=<property value>`  
Example  
  
`worker.ajp13.host=localhost`  
`worker.ajp13.port=8007`  
`worker.ajp13second.host=otherhost`  
`worker.ajp13second.port=8007`
4. You can now assign the workers to serve the various contexts (one worker for each context).

For example, look at the following uriworkermap.properties fragment:

```
/examples/*=ajp13  
/webpages/*=ajp13second
```

The examples context is served by ajp13 while the webpages context is served by ajp13second.