JAX-RS - 1.3

On this page:

Target audience:

Users of the extension providing **JAX-RS** support for Web Services.

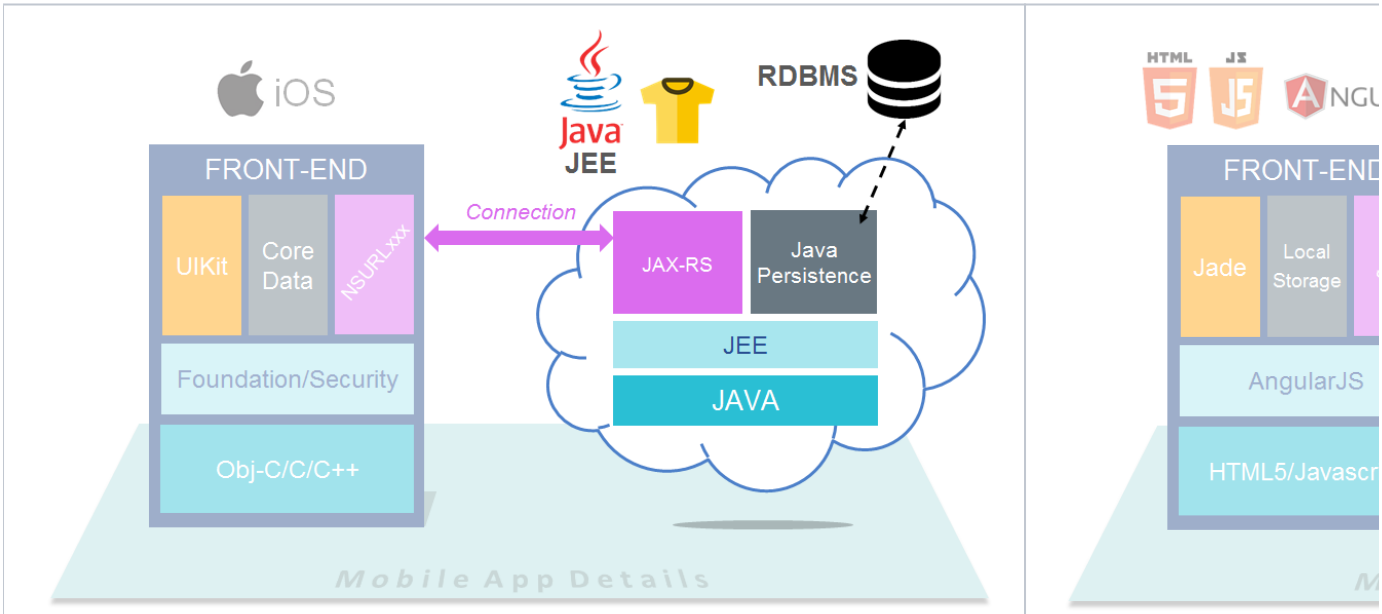
Summary: This document provides basic information about the extension providing **JAX-RS** support for Web Services.

Description

This extension provides support for **JAX-RS**.

In what situation should you install this extension?

The main purpose of this extension is to create **HTTP API entry points**, to enable linking from a Web App front end. Therefore if your Web application contains source code which uses **JAX-RS** (**1.0** (JSR 311) and **2.0** (JSR 339)) and you want to view these object types and their links with other objects, then you should install this extension.



iOS Front-End Example

Features

This extension handles JAX-RS Web Services used in J2EE applications, for example:

```
@Singleton
@Path("/printers")
public class PrintersResource {
     @GET
     @Produces({"application/json", "application/xml"})
     public WebResourceList getMyResources() { ... }

     @GET @Path("/list")
     @Produces({"application/json", "application/xml"})
     public WebResourceList getListOfPrinters() { ... }

     @PUT @Path("/ids/{printerid}")
     @Consumes({"application/json", "application/xml"})
     public void putPrinter(@PathParam("printerid") String printerId,  Printer printer) { ... }

     @DELETE @Path("/ids/{printerid}")
     public void deletePrinter(@PathParam("printerid") String printerId) { ... }
 }
```

For each class annotated with **javax.ws.rs.Path (@Path)**:

- A Web Service object will be created whose name is deduced from the value of @Path on the class
- A Web Service Port object will be created that is a child of the web service with the same name and has with a prototype link to the class
- For each method annotated with @GET, @PUT, @DELETE or @POST, the following will be created:
    - a Web Service operation child of the Web Service Port whose name is the concatenation of the @Path specified at the class level and the @Path specified at the method level. If no @Path on the method, then method path = /
    - a fire link from the Web Service operation to the method

Annotations considered

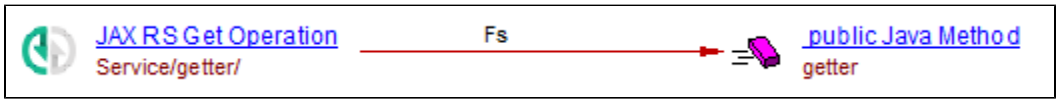
- for the URLs:
    - **javax.ws.rs.Path**
- for the access type:
    - **javax.ws.rs.GET**
    - **javax.ws.rs.PUT**
    - **javax.ws.rs.DELETE**
    - **javax.ws.rs.POST**

Basic case

The following Java code:

```
import javax.ws.rs.Path;
import javax.ws.rs.GET;


@Path("Service")
public class MyClass
{
    @Path("/getter")
    @GET
    public void getter() {}
}
```

will generate :



Inheritance case

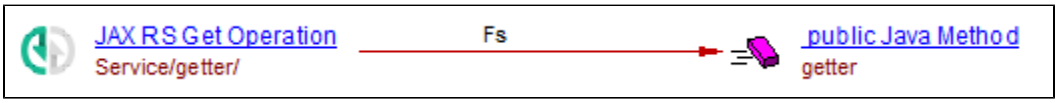
The annotations may also be in the parent class or interface:

```
import javax.ws.rs.Path;


@Path("Service")
public class MyClass implements MyInterface
{
    public void getter() {}
}
```

Here the interface defines the exposed methods:

```
import javax.ws.rs.Path;
import javax.ws.rs.GET;


public interface MyInterface
{
    @Path("/getter")
    @GET
    public void getter();
}
```

and will generate :

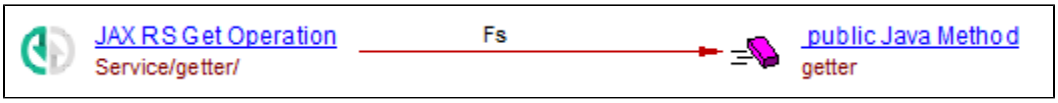


Here the interface exposes the main path and the implementation defines the exposed methods:

```
import javax.ws.rs.Path;


public class MyClass implements MyInterface
{
    @Path("/getter")
    @GET
        public void getter() {}
}
```

```
import javax.ws.rs.Path;
import javax.ws.rs.GET;

@Path("Service")
public interface MyInterface
{

    public void getter();
}
```

and will generate :



What's New

- The extension is now shipped with a set of CAST Transaction Configuration Center **Entry Points**, specifically related to **JAX-RS** Please see CAST Transaction Configuration Center (TCC) Entry Points below for more information about this.

Function Point, Quality and Sizing support

This extension provides the following support:

- **Function Points (transactions)**: a green tick indicates that OMG Function Point counting and Transaction Risk Index are supported
- **Quality and Sizing**: a green tick indicates that CAST can measure size and that a minimum set of Quality Rules exist

Function Points (transactions)	Quality and Sizing
✅	❌

CAST AIP compatibility

This extension is compatible with:

CAST AIP release	Supported
8.3.x	✓
8.2.x	✓
8.1.x	✓
8.0.x	✓
7.3.4 and all higher 7.3.x releases	✓

Supported DBMS servers

This extension is compatible with the following DBMS servers:

CAST AIP release	CSS2	Oracle	Microsoft
All supported releases	✓	✓	✗

Prerequisites

✓	An installation of any compatible release of CAST AIP (see table above)

Dependencies with other extensions

Some CAST extensions require the presence of other CAST extensions in order to function correctly. The **JAX-RS** extension requires that the following other CAST extensions are also installed:

- **Web services linker service** (internal technical extension)

> Note that when using the **CAST Extension Downloader** to download the extension and the **Manage Extensions** interface in **CAST Server Manager** to install the extension, any dependent extensions are **automatically** downloaded and installed for you. You do not need to do anything.

Download and installation instructions

Please see:

- http://doc.castsoftware.com/display/EXTEND/Download+an+extension
- http://doc.castsoftware.com/display/EXTEND/Install+an+extension

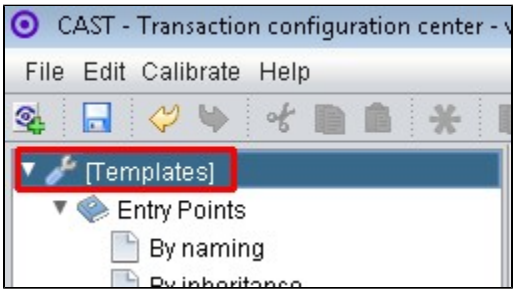
> The latest release status of this extension can be seen when downloading it from the CAST Extend server.

CAST Transaction Configuration Center (TCC) Entry Points

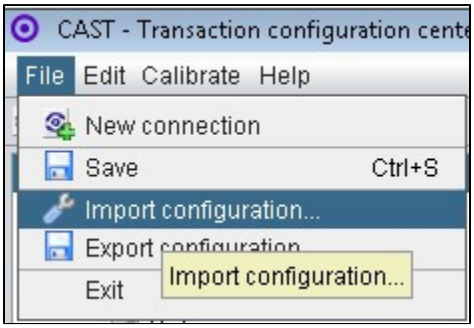
In **JAX-RS 1.3.x**, if you are using the extension with **CAST AIP 8.3.x**, a set of JAX-RS specific **Transaction Entry Points** are now **automatically imported** when the extension is installed. These **Transaction Entry Points** will be available in the CAST Transaction Configuration Center:

Manual import action for CAST AIP 8.2.x

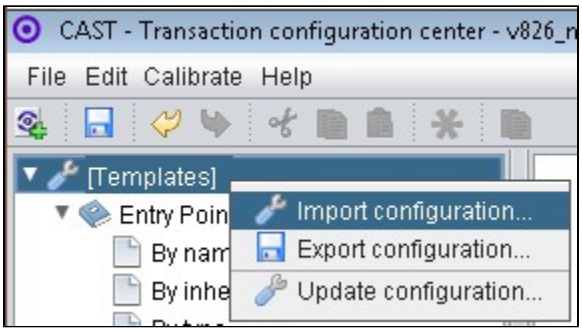
- Locate the .TCCSetup file in the extension folder: **Configuration\TCC\Base_Java_SpringMVC.TCCSetup**
- In the CAST Transaction Configuration Center, ensure you have selected the **Templates** node:



- This .TCCSetup file is to be imported into the CAST Transaction Calibration Center using either the:
  - **File** > **Import Configuration** menu option:

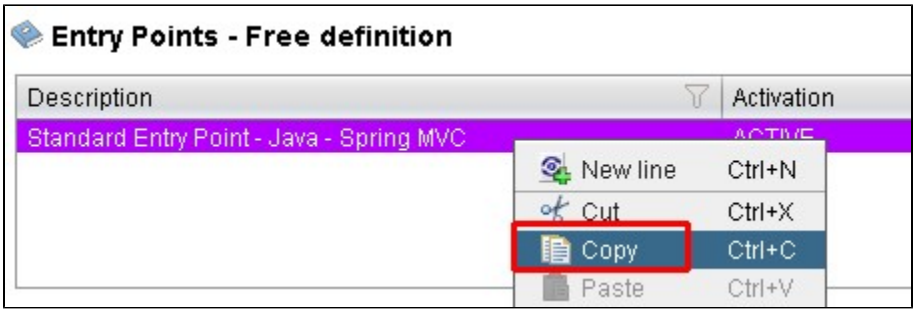


  - Or right clicking on the **Template node** and selecting **Import Configuration**:

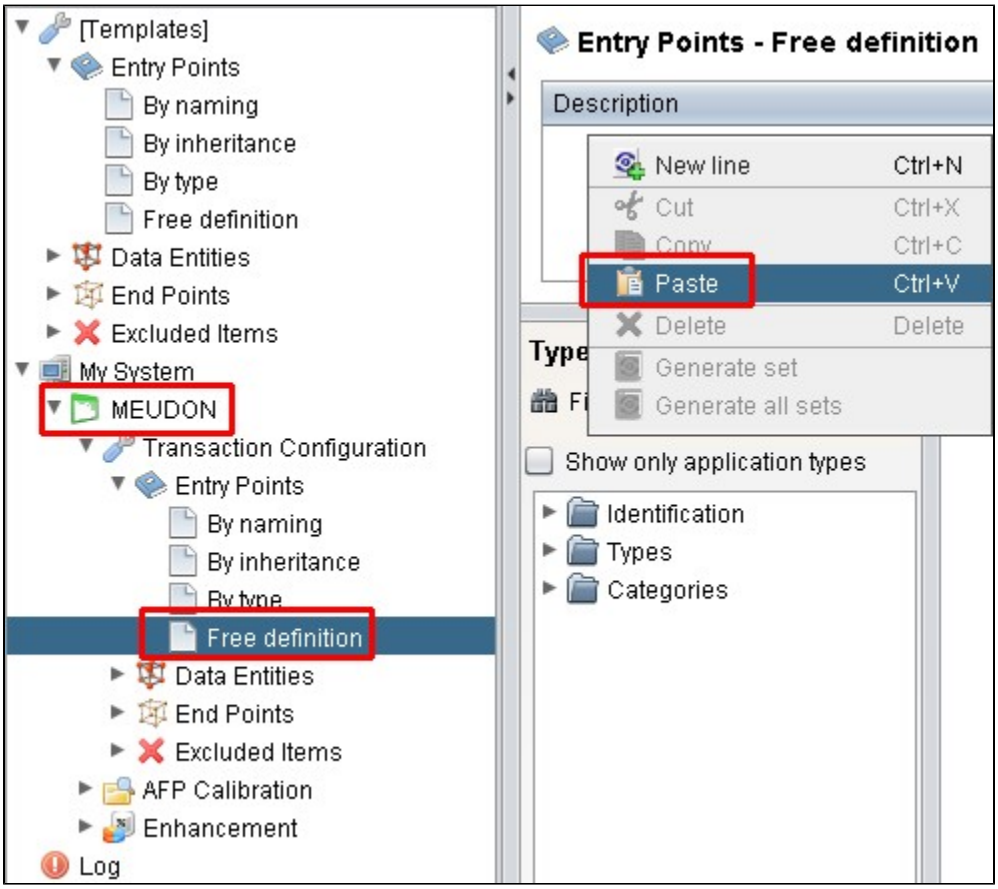


- The import of the "**Base_Java_SpringMVC.TCCSetup**" file will provide you with a sample Transaction Entry point in the **Free Definition** node under **Templates**:

- Now right click the "**Standard Entry Point**" item and select copy:



- Paste the item into the **equivalent node** under the **Application**, for example, below we have copied it into the **Application MEUDON**:



- Repeat for any additional items or generic sets that have been imported from the .TCCSetup file.

Packaging, delivering and analyzing your source code

Once the extension is installed, no further configuration changes are required before you can package your source code and run an analysis. The process of packaging, delivering and analyzing your source code does not change in any way:

- **Package and deliver** your application (that includes source code which uses **JAX-RS**) in the exact same way as you always have. You can refer to the existing official CAST documentation for more information about this - see: http://doc.castsoftware.com/display/DOC82/Source+Code+Delivery+Guide+for+Application+Teams.
- **Analyze** your delivered application source code in the CAST Management Studio in the exact same way as you always have - the source code which uses **JAX-RS** will be detected and handled correctly. You can refer to the existing official CAST documentation for more information about configuring an analysis - see: http://doc.castsoftware.com/display/DOC82/2.+Application+Analysis+Process+with+CAST+AIP

What results can you expect?

Once the analysis/snapshot generation has completed, **HTTP API** transaction entry points will be available for use when configuring the CAST Transaction Configuration Center. In addition, you can view the results in the normal manner (for example via CAST Enlighten).

Objects

The following objects are displayed in CAST Enlighten:

Icon	Description
	JAX-RS Delete Operation Service
	JAX-RS Get Operation Service
	JAX-RS Post Operation Service
	JAX-RS Put Operation Service
	JAX-RS Port
	JAX-RS Service