

CMS - Properties Files Handling (J2EE)

Properties Files Handling (J2EE)

Note that the CAST Script feature described in this page is deprecated. You should instead consider using the [Extension SDK](#) to handle properties files.

This page provides information about how the J2EE Analyzer handles **Properties** files (*.properties).

The problem...

Many JSP files use the properties mechanism provided by the Java platform. Hence, instead of hard coding string values, they are stored in separate files in the Key = Value format. Each value is then referenced in the code (JSP Page, Java file...) using its corresponding key.

Since the J2EE Analyzer does not have any knowledge of these properties it will try to resolve Keys as target objects wherever resolution is to be carried out. This, of course, will lead to a failure or to bad results (a key name matches an object name but its value doesn't).

In the code fragment below, the J2EE Analyzer will attempt to resolve the string "LoginPage" as a URL and this is obviously not what we want.

```
<a href="LoginPage">Log In</a>
```

The line:

```
LoginPage = /authentication/login.jsp
```

exists in a properties file within the application but will not be seen by the analyzer unless we tell it about it.

Solution

By allowing the user to select .properties files as part of a job, the J2EE Analyzer provides a way to identify properties in these files and resolve their values as target objects.

Once the job is run, the properties files and their content will be stored in the Analysis Service. A **JSP Properties Files** object will be created for the file and a **JSP Property Mapping** for each key. For compound keys (with qualified names) such as MyKey.InThisCase, an object is created for each qualifier and stored in a hierarchical manner. The key value is associated to the rightmost (leaf) object.

In this way, each piece of code that references a key will have an outgoing link toward the JSP Property Mapping that creates this key.

In turn, the Property Mapping will be linked to the object (if any) that corresponds to its value.

To find this object, the J2EE Analyzer resolves the value of the key as:

1. Java Type (value need to be a fully qualified java type name)
2. URL (including files, servlet mappings, property mappings and action mappings)
3. Server Object (tables). Resolution succeeds if a substring of the value or the whole value matches a server object name.

Example:

Given a properties file **maps.properties** containing the following keys:

```
Functionality.Controller = com.functionality.controllerFunctionality.JSPPage = /functionality/i.jsp
```

and a JSP file **maps.jsp** containing:

```
<a href="Functionality">Functionality </a><a href="Functionality.Controller">Functionality.Controller</a><a href="Functionality.JSPPage">Functionality.JSPPage</a>
```

...the following objects will be created and organized as shows the image bellow:

- JSP Properties Files "maps.properties"
- JSP Property Mapping "Functionality"
- JSP Property Mapping "Functionality.Controller"
- JSP Property Mapping "Functionality.JSPPage"



And the links between the objects will be displayed as follows in CAST Enlighten:

