CMS - External Link engine parametrization

External Link engine parametrization

CAST provides the Dynamic Link Manager to process unverified links (links that have been identified between objects but their authenticity cannot be absolutely determined by CAST) that have been identified during an analysis and then saved to the Analysis Service.

However, in some circumstances, a large amount of unverified links can be generated during an analysis and often these unverified links systematically occur or have no meaning for the specific technology you are working with. In this case, it can be very time consuming to process the links manually in the Dynamic Link Manager.

As such, CAST has introduced a method to allow the automatic verification (and subsequent link validation or ignore) of any links that are detected as unverified during the analysis and which would have been displayed in the Dynamic Link Manager. The method relies on an **XML file** that contains specific instructions about how a potential link should be handled when it is encountered in an analysis. The file is named as follows:

CAST_INSTALL_FOLDER\configuration\parametrization\CustomExternalLinksRules.xml

Note that the file may be delivered empty, therefore the basic XML elements must be added as follows:

```
<?xml version="1.0" encoding="utf-8"?><ExternalLinksRules xmlns="http://tempuri.org/ExternalLinksRules.xsd">
[rules go here]</ExternalLinksRules>
```

To use the file, you need to fill it with your own instructions based on what you have done in the past with links that have consistently been displayed in the Dynamic Link Manager. The file is taken into account automatically when an analysis occurs. As such, if you make changes to the file, ensure you re-run your analyses.

Rule creation

There are three main types of rules that can be created. These are represented by the following XML elements:

- **<When-NotMatch-FullName>** - this is a matching rule. This forces the search mechanism to search on an object's full name and not on its simple name. See example below.
- **<remove>** - this is an action rule - any link that is matched in this element will be removed from the CAST Analysis Service and will NOT appear in the Dynamic Link Manager
- **<valid>** - this is an action rule - any link that is matched in this element will be verified and created in the CAST Analysis Serviceand will NOT appear in the Dynamic Link Manager

All these rules must appear within the **<ExternalLinksRules>** element. Here are some example rules to get you going. You can mix and match rule elements as required:

Example 1

Matching rule: forces the search mechanism to search on an object's full name, rather than a simple name. For the below example, only objects with the full name "a.b.c" will be found, object "c" with full name "a.b" will not be found.

```
<When-NotMatch-FullName>    <callee>a.b.c</callee></When-NotMatch-FullName >
```

Example 2

Matches all links to callee of type JSP_PROPERTY_MAPPING and name begins with P_:

```
<When-Match-Regexp>   <callee type="JSP_PROPERTY_MAPPING">      <regexp>P_[a-z0-9]+</regexp>    </callee></When-Match-Regexp>
```

Example 3

Matches all links when the caller is of type JV_FIELD (Java Member):

```
<When-Match-Regexp> =>     <caller type="JV_FIELD">   </caller></When-Match-Regexp>
```

Example 4

Matches all links when the callee is of type CAST_COBOL_Program, AND with name beginning with P_, AND when the caller is of type JV_CLASS (Java Class):

```
<When-Match-Regexp>   <caller type="JV_CLASS">    </caller>   <callee type="CAST_COBOL_Program">       <regexp>P_
[a-z0-9]+</regexp>    </callee> </When-Match-Regexp>
```

Example 5

Matches all links created when string contains "MsBox". E.g.: "MsgBox : select * from Authors"

```
<When-Match-Regexp> =>    <string-from-code>      <regexp>MsgBox</regexp>   </string-from-code></When-Match-
Regexp>
```

Example 6

Matches all links when the callee is of type JSP_SERVLET_MAPPING, AND that have a "?" after the matched string. E.g.: "Myservlet?"

```
<callee type="JSP_SERVLET_MAPPING">    <right-part-extracted-string-from-code>      <regexp>"[ \t\r\n]*\?"<
/regexp>    </right-part-extracted-string-from-code> </callee>
```

Example 7

Matches all links when the callee is of type JSP_SERVLET_MAPPING, AND that have a "#" before the matched string. E.g.: "#Myservlet"

```
<callee type="JSP_SERVLET_MAPPING">    <left-part-extracted-string-from-code>      <regexp>"#"</regexp>    </left-
part-extracted-string-from-code> </callee>
```

Default rules

CAST provides default rules in the following file:

CAST_INSTALL_FOLDER\configuration\parametrization\DefaultExternalLinksRules.xml

CAST recommends leaving these default rules in place and creating new rules for your own environment.

Rule hierarchy

Note that the rule **<valid>** has precedence over the rule **<remove>**. As such, if a link matches a rule in the **<valid>** AND the **<remove>** elements, then the rule in the **<valid>** element will take precedence.

See Also

[Reviewing Dynamic Links](#)