

CMS - Dynamic Link Manager filter rules

Dynamic Link Manager filter rules

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](#).

In previous releases, CAST introduced [External Link engine parametrization](#) which enabled the user to create rules to either validate or ignore certain links every time an analysis was run. However, this method is not flexible and cannot be tested accurately before being put into practice. To counter this, CAST has introduced filter rules that can be written, tested and applied in the CAST Management Studio GUI at Application and Analysis Unit/Technology level: each time an analysis is run, the filter rules will be applied, either validating or ignoring links as required. In addition, rules can be processed via a command line utility for one-shot processing - this negates the need to re-run the analysis just to get the rules processed.

Rule files and filters

Filter rules use XML formatting and must, therefore be created in a file with an **.DLM.XML extension** (for example **rules.DLM.XML**):

- You can create as many XML "rule" files as you need, or you can place all your filter rules in one XML "rule" file.
- The XML rule file will contain a **flat list of filtering rules**
- A filtering rule is an **ordered list of filters**, with an action to perform on the filtered links (validate or ignore)
- A rule must be **unique for the Application**, i.e. it cannot be present in the same rule file more than once and cannot be present in multiple rule files
- A rule must use a **unique, case sensitive name**, in order to be identified easily and define an action: **validate** or **ignore**
- A rule has an optional **scope**, which means it is possible to specify on what **application/analysis unit/technology** you want to apply the rule. Multiple applications in the can be added to the scope (multiple Applications are handled via an OR operation, i.e. the link only has to be part of one of the sub-scopes - i.e. analysis unit/technology)
- A rule must contain a **link definition** (the link that will be validated or ignored) based on the Callee or Caller object, a link type, or a line/part of a line of source code.
- The rules are executed in the order that they are defined in the XML file, and inside each rule, the filters are also executed in the order that they are defined. The filters are applied with a logical conjunction (an AND operation, i.e. the link has to be accepted by every filter in the rule)
- CAST recommends that the [DlmRulesModel.xsd](#) is applied to the XML rule file

Example rule file and explanation

Take the following example rule file and the filters defined in it:

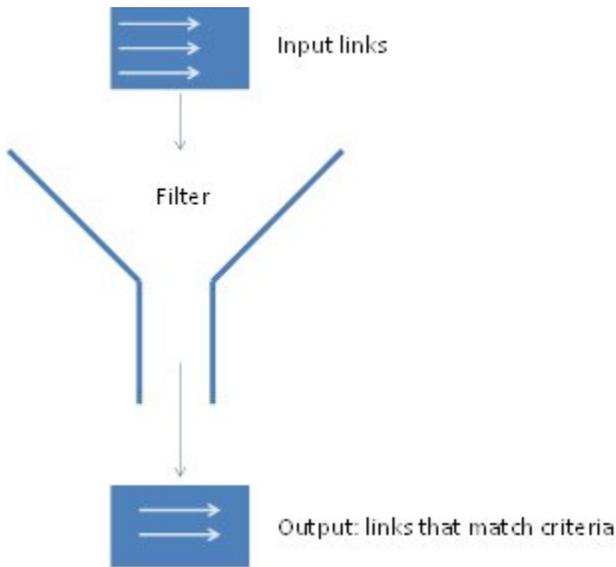
```
<?xml version="1.0" encoding="utf-8"?><dynamicLinksRules xmlns="http://www.castsoftware.com/DlmRulesModel.xsd"
><rule name="Rule1" action="validate" > <scope> <application name="app1"/> <application name="app2">
<analysisUnit name="au21"/> </application> <application name="app2"> <technology name="J2EE"/> <
/application> </scope> <calleeName regexp="a2"/> <calleeType names="A_Metamodel_Type"/> <callerFullName regexp="
a4"/></rule><rule name="Rule2" action="ignore"> <calleeFullName regexp="a4"/></rule></dynamicLinksRules>
```

When this rule file is used:

- **Rule1** will be processed first (this has a **validate** action)
- The entire set of unreviewed Dynamic Links that are available in the Analysis Service will be taken into account
- The following actions are then executed in the following order:
 - Selecting as a **scope** all links that belong to:
 - Application **app1**
 - The Analysis Unit **au21** located in Application **app2**
 - All J2EE links in the application **app2**
 - Filtering by callee name = **a2**
 - Filtering by callee type = **A_Metamodel_Type**
 - Filtering by caller full name = **a4**
 - Tag the result as **Validate**
- Then **Rule2** will be processed
- The entire set of unreviewed Dynamic Links that are available in the Analysis Service will be taken into account (there is no scope specified at all), except the links that have been processed by **Rule1**.
- The following actions are then executed in the following order:
 - Filtering by callee full name= **a4**
 - Tag the result as **Ignore**

Filters in detail

As described previously, a filter is an operator that selects links that match criteria:



Filters are piped together in order to compose the body of a rule. The final result will be tagged as **validated** or **ignored**.

Scope

You can define the following scope entries in your filter:

Application	<code><scope> <application name="appl"/></scope></code>
Technology	<code><scope> <application name="application"> <technology name="technology"/> </application></scope></code>
Analysis Unit	<code><scope> <application name="application"> <analysisUnitname="analysisUnit"/> </application></scope></code>

Note that scopes are completely ignored when you assign the rule file in the CAST Management Studio. In this case, the location where you assign the rule file (see below) will act as the scope (Application/Technology/Analysis Unit).

Filter list

You can define the following link definition filters:

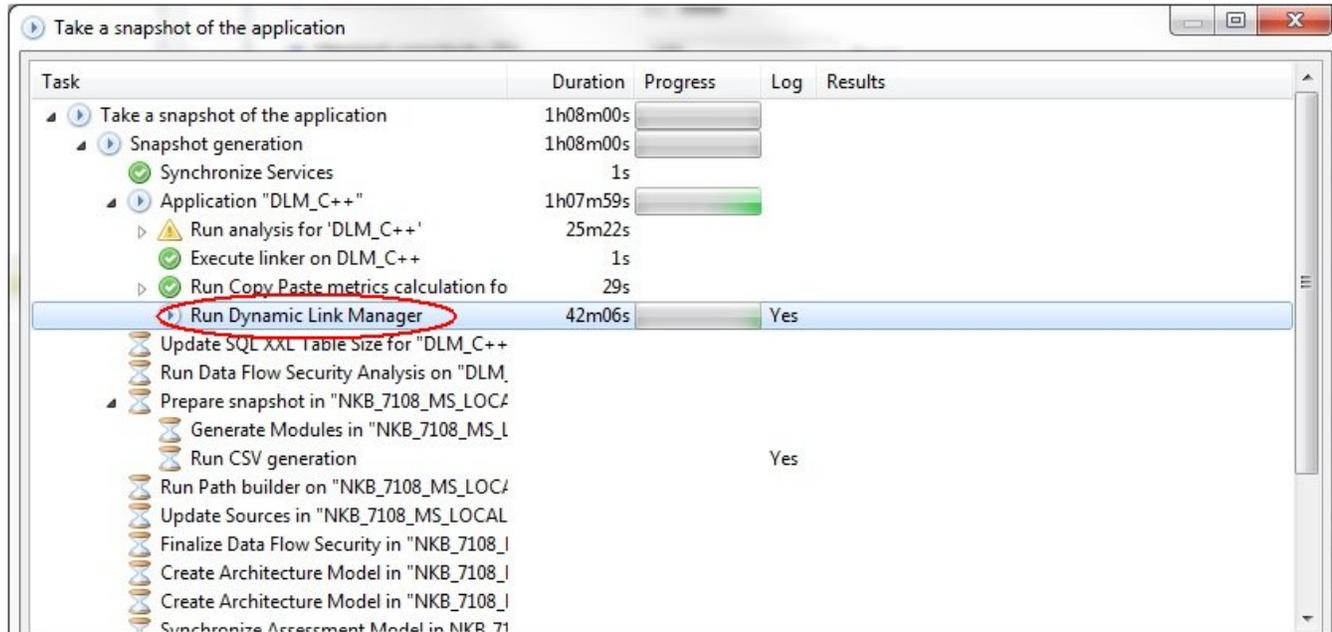
<code><callerName regexp="string"/></code>	Filters the stream according to the caller name	See Filter rules - Regular Expression based
<code><callerFullName regexp="string"/></code>	Filters the stream according to the caller full name	See Filter rules - Regular Expression based
<code><callerType names="string"/></code>	Filters the stream according to the caller type	See Filter rules - Metamodel based
<code><calleeName regexp="string"/></code>	Filters the stream according to the callee name	See Filter rules - Regular Expression based
<code><calleeFullName regexp="string"/></code>	Filters the stream according to the callee full name	See Filter rules - Regular Expression based
<code><calleeType names="string"/></code>	Filters the stream according to the callee type	See Filter rules - Metamodel based
<code><linkType names="string"/></code>	Filters the stream according to the link type	See Filter rules - Metamodel based
<code><rawStringFromCode regexp="string"/></code>	Filters the stream according to a regular expression on the whole line of code	See Filter rules - Bookmark based
<code><leftStringFromCode regexp="string"/></code>	Filters the stream according to a regular expression on the left part of the code before the matched callee name	See Filter rules - Bookmark based
<code><rightStringFromCode regexp="string"/></code>	Filters the stream according to a regular expression on the right part of the code after the matched callee name	See Filter rules - Bookmark based

Putting filters into practice

CAST recommends that filters are introduced in the following way:

1. **Identify recurring dynamic links** using the GUI based Dynamic Link Manager (see [Reviewing Dynamic Links](#)) based on Caller and Callee objects and the line of code.
2. Create filters to match these recurring dynamic links

3. Run the filters using the command line utility (see below for more information)
4. Verify that the initial command line run **has functioned as expected** using the GUI based Dynamic Link Manager (see [Reviewing Dynamic Links](#))
 - i.e. check that the links you wanted to validate/ignore have been validated/ignored.
5. Associate your rule file to the Application/Technology/Analysis Unit in the CAST Management Studio GUI (see below for more information)
6. Any rule files associated through the GUI will be processed each time you run an analysis:



Command line

To test your filters, please use the following command line. You can find the **DynamicLinks.CLI.exe** file in the **DLM folder** at the root of your CAST installation:

```
DynamicLinks.CLI.exe executeRules --ruleFile="yourRuleFile.DLM.XML" --profile="My Management on CastStorageService"
```

- Replace "**yourRuleFile.DLM.XML**" with the name of the XML files containing your filters.
- Replace "**My Management on CastStorageService**" with the name of the Connection Profile used to connect to the current Management Service. You can find this in the [Edit Connection Profiles dialog box](#).

Note that if you want to **undo** any actions that have been applied by the Command Line explained above, you can use the following "undo" syntax. In this situation, all actions previously applied in your filters will be **undone** (whether Validate or Ignore) and the status of the links will be set to **To be reviewed**.

```
DynamicLinks.CLI.exe undoRules --ruleFile="yourRuleFile.DLM.XML" --profile="My Management on CastStorageService"
```

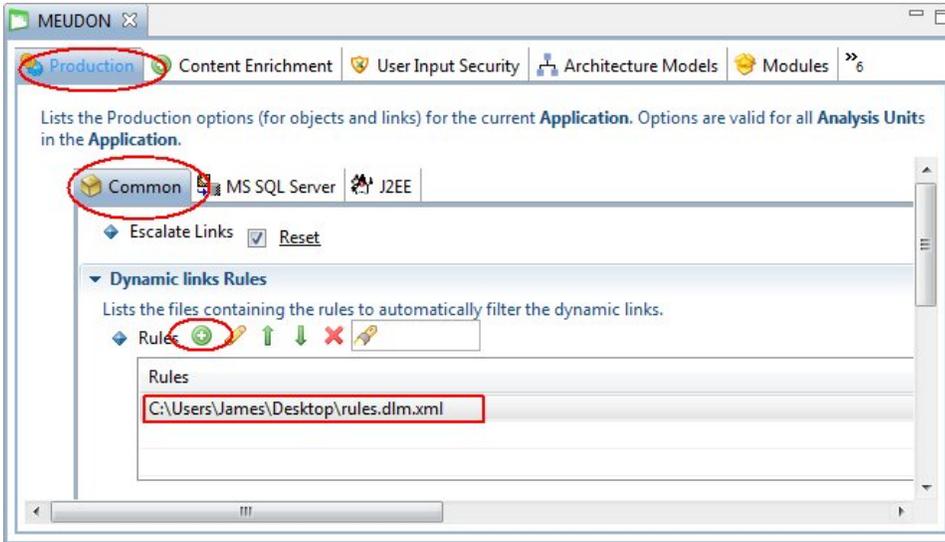
Associating rule files to Applications/Technologies/Analysis Units

To associate your rule file in the CAST Management Studio GUI:

Application

To apply the filter rule at Application level (i.e. to all Analysis Units and Technologies present in the Application), use the:

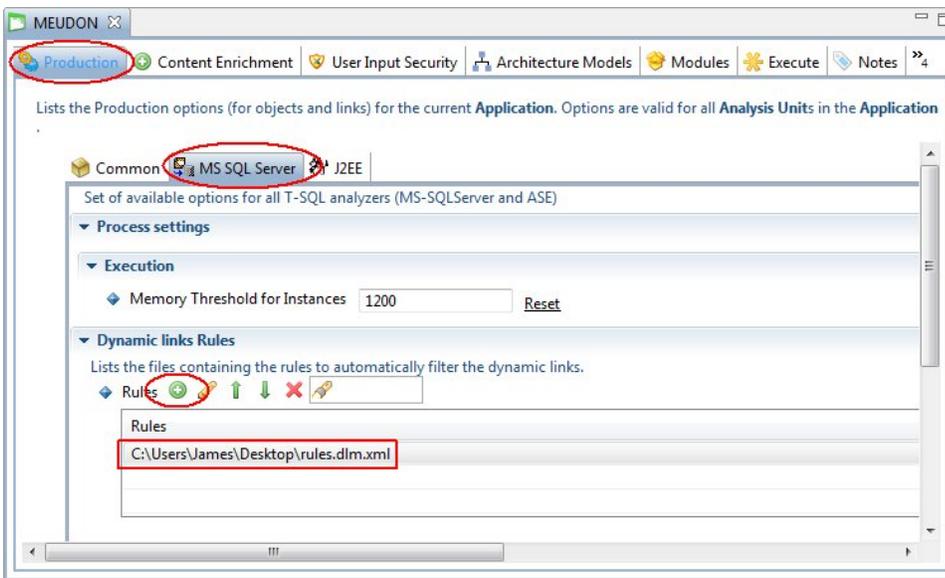
- **Application editor > Production tab > Common tab > Dynamic Link Rules:**



Technology

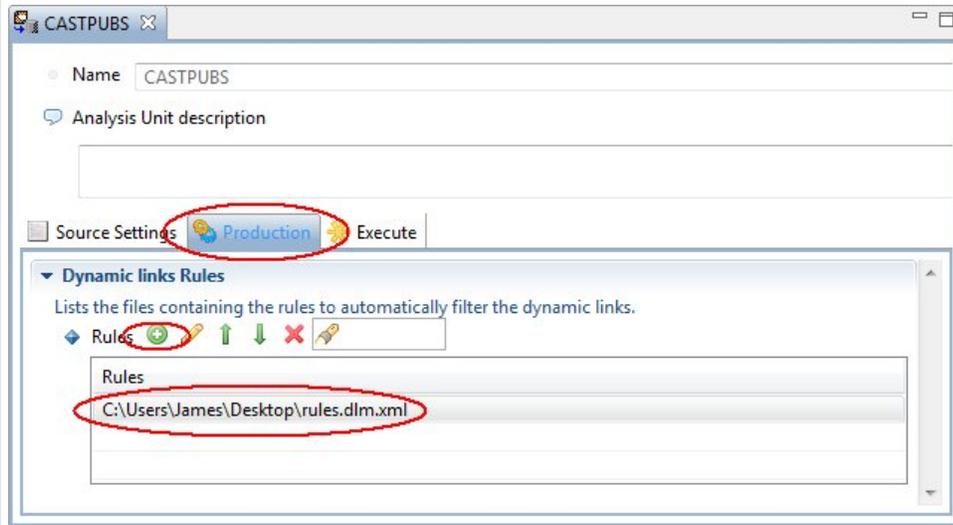
To apply the filter rule at Technology level (i.e. to all of the selected Technologies present in the Application), use the:

- **Application editor > Production tab > [Technology_type] tab > Dynamic Link Rules:**



To apply the filter rule at Analysis Unit level (i.e. to the Analysis Unit alone), use the:

- Analysis Unit editor > Production tab > Dynamic Link Rules:



See Also

[Filter rules - Bookmark based](#) | [Filter rules - Metamodel based](#) | [Filter rules - Regular Expression based](#)