Application - Config - Reference Finder

Summary: this section explains how to work with the Reference Finder feature.

Introduction

The **Reference Finder** feature enables you to define one or multiple rules at Application level to search for links (i.e. a **word**, **series of words** or **string** (ba sed on **Regular Expressions**)) between Source and Target code when an analysis is run. The Reference Finder is an extension of the **Dependencies** feat ure managed in the **Application - Config - Analysis** panel: in the Dependencies feature, references are traced using **search strings** which is less selective than parser based technology used for other links traced by the analyzer. This technology detects a reference to an object wherever its name is mentioned, regardless of the context in which this reference occurs. As a result, **incorrect links** may be traced if a string happens to match a given name even if it is not logically related to the corresponding object. As a result you may have to **intervene** to filter incorrect references. Refer to the **Application - Config - Summary of Dynamic Links** for more details on how to ignore these ambiguous links.

The Reference Finder is therefore particularly useful when **standard (i.e. built in to Console) Dependencies** based solely on simple word based matches between Technologies are either too broad (creating too many "false" links between objects) or do not detect the links you require. Since the Reference Finder is based on Regular Expression matching, you can define very specific search strings to identify the links between source code that you require.

If you have **imported existing Applications into AIP Console that were previously managed in CAST Management Studio**, any **Reference Pattern configurations** that exist in CAST Management Studio will not be made available in this screen in Console, however they will still be taken into account when an analysis is run. You should continue to use CAST Management Studio to manage these existing Reference Patterns.

Available options

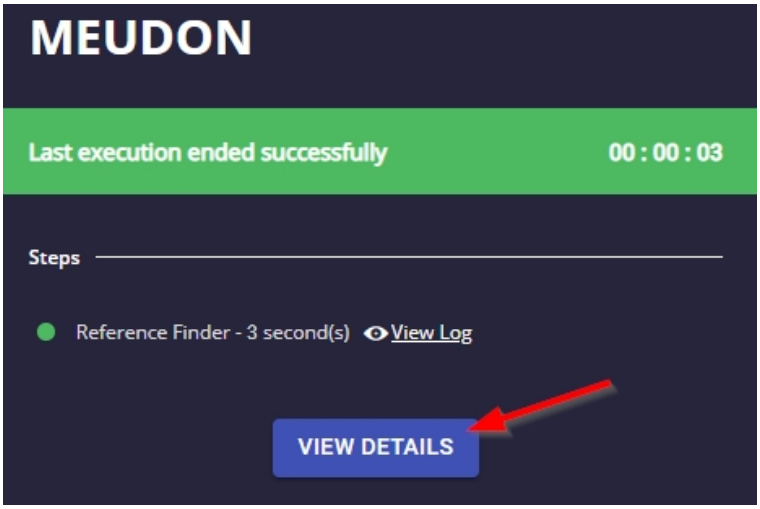
This section lists all Reference Finder rules that already exist with a brief summary of the configuration:

+ ADD	Add a new Reference Finder rule. See below.
Name	Name of the Reference Finder rule.
Description	An optional description for the rule.
Source Technology	The source technology for the rule.
Target Technology	The target technology for the rule

Options

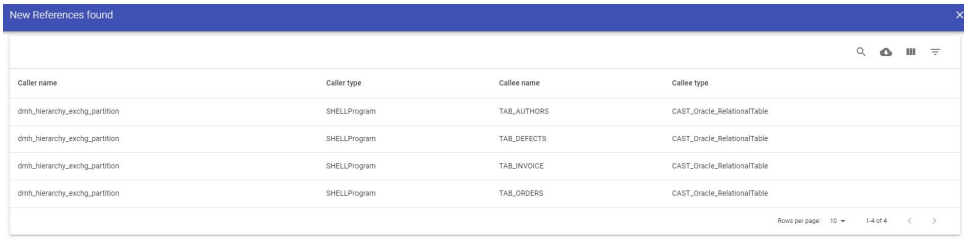
Check

Use this option to perform a manual check of the rule. The process slide in will be displayed:



Results are available using the **VIEW DETAILS** button:

Click to enlarge

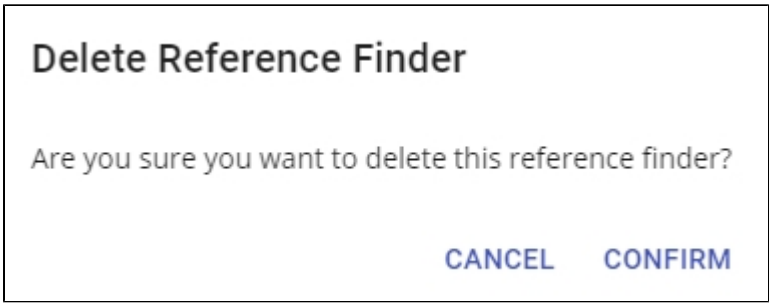


Edit

Will open the selected rule for editing.

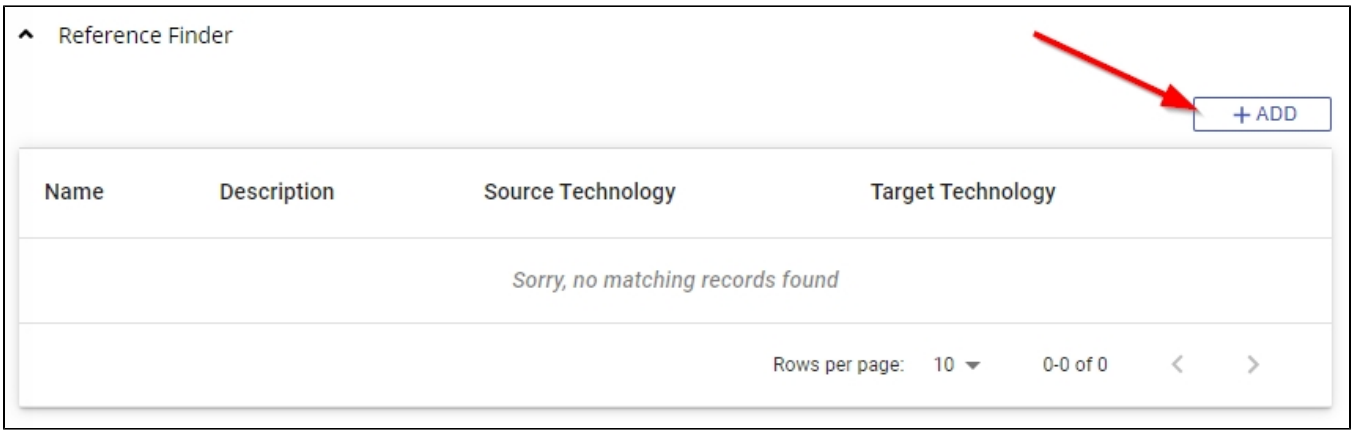
Delete

Removes the rule - the rule will no longer be available for use and will not be taken into account during the next analysis. Yo
delete action:

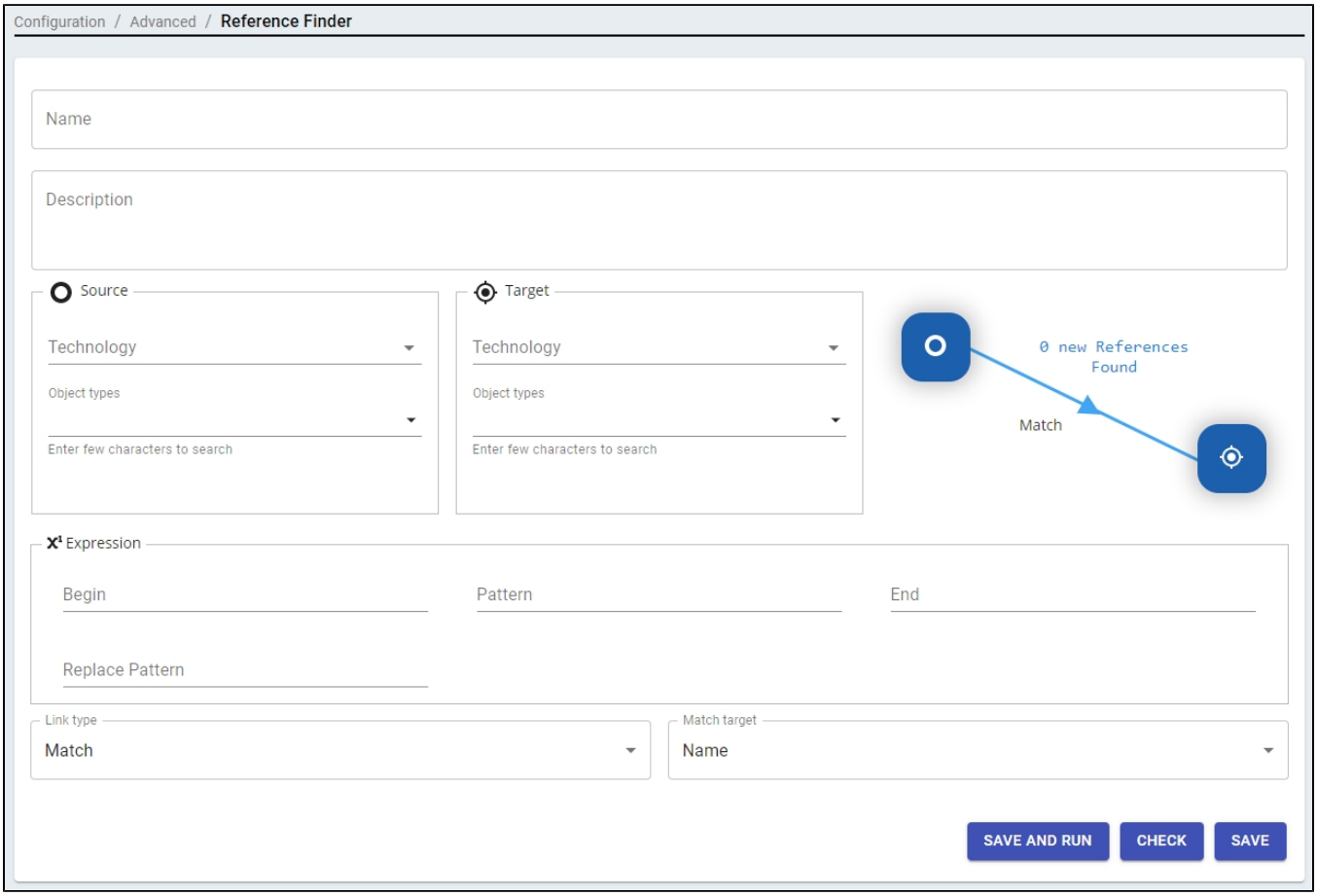


Add a new Reference Finder rule

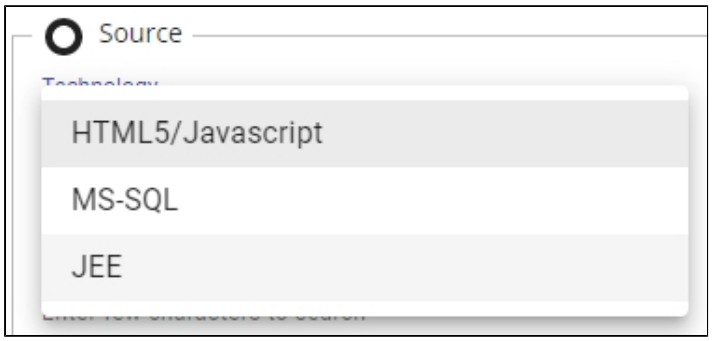
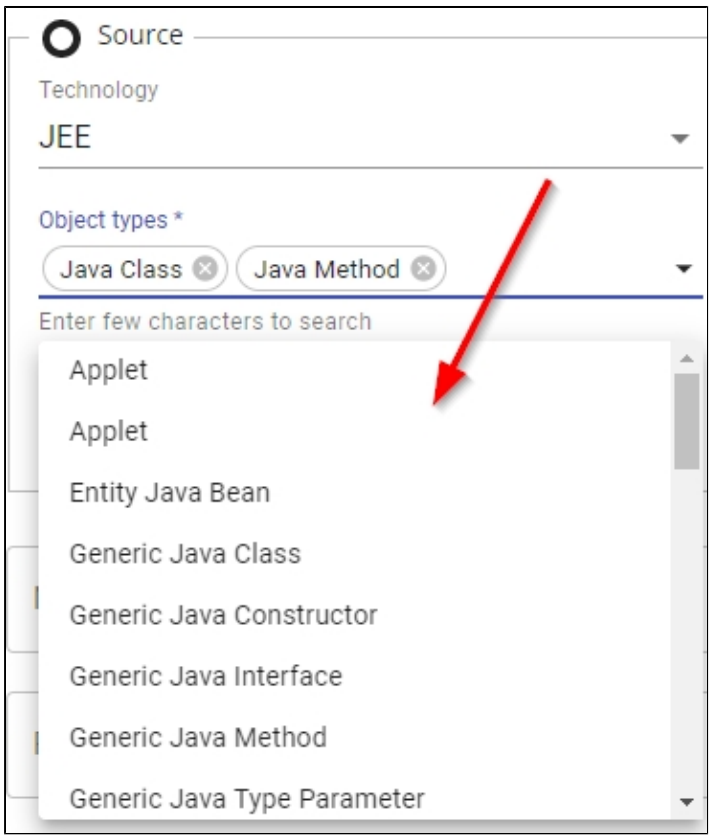
Use the **Add** button to create a new empty Reference Finder rule:



The creation window will then be displayed enabling you define the rule:



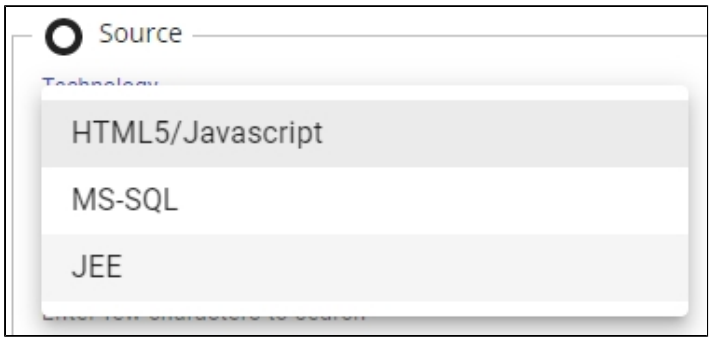
Name	Choose a name for the Reference Finder rule to identify it. The characters in the Reference Finder name must match the following regular `[a-zA-Z][a-zA-Z0-9]+` For example, a name such as "**Test Rule**" (with a white space) is not permitted.
Description	This field is a simple free text field that allows you to enter an optional description of the rule - i.e. what is it intended to do.

Source	**Technology** Select the **Source** technology in the drop down. The list of technologies are those that **exist in the current version** and therefore the list when a **Version** has been **set as the current version**:  **Object types** Now select the specific object types within your chosen Technology that you want to focus on. These object types are taken from the tech not from the current application, therefore, the specific object type may not be present in your application. You can select multiple object t below. Use the X icon in an object type if you change your mind and do not want to include that specific object. 

Target

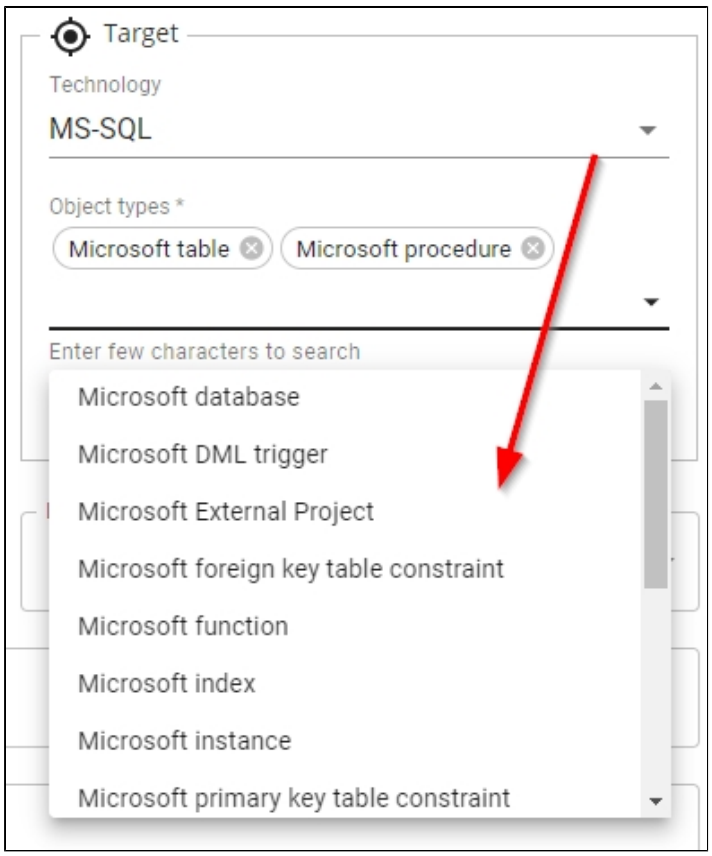
Technology

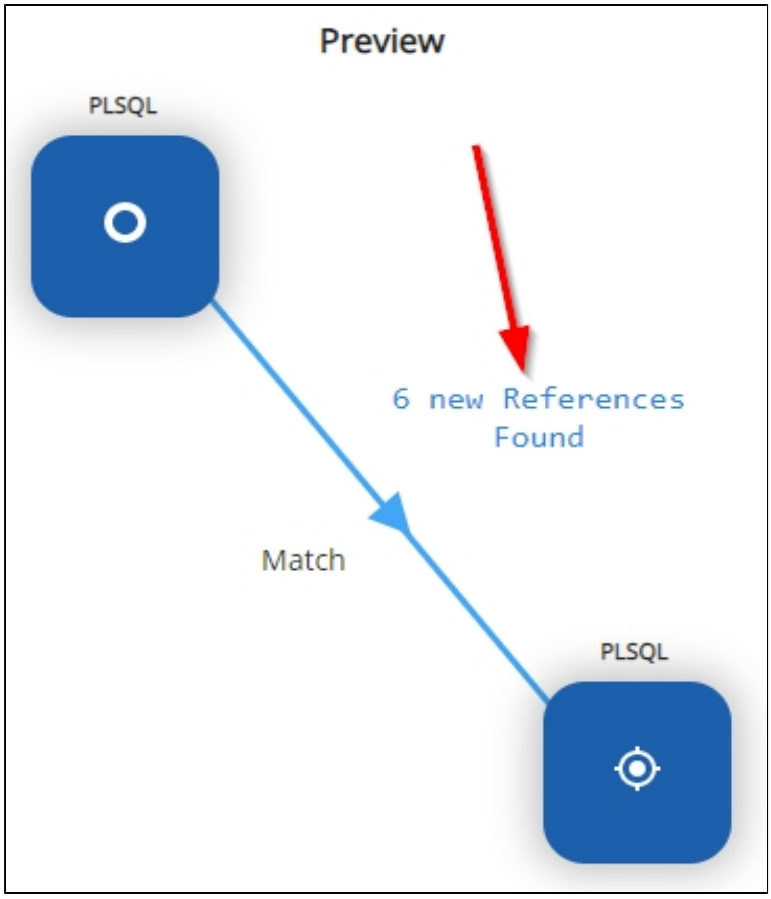
Select the **Target** technology in the drop down. The list of technologies are those that **exist in the current version** and therefore the list v
when a **Version** has been **set as the current version**:



Object types

Now select the specific object types within your chosen Technology that you want to focus on. These object types are taken from the tech
not from the current application, therefore, the specific object type may not be present in your application. You can select multiple object t
below. Use the X icon in an object type if you change your mind and do not want to include that specific object.



Preview window	The preview window is used to display the source and target technologies for the current rule and any results that the rule identifies when pressed (see **below**). 
Expression	**Pattern - mandatory** Enter the **Regular Expression**, **word** or **phrase** that you want the Reference Finder rule to target. For example to match any word in upp beginning with TAB use: `TAB_[A-Z]+` Console uses **Python** Regular Expression syntax and you can can find some hints and tips here: • **https://docs.python.org/3/howto/regex.html#regex-howto** • **https://docs.python.org/3/library/re.html#regular-expression-syntax**

Begin / End - optional

You can optionally choose to restrict the pattern search to a specific zone. You can specify the **begin** and/or **end** of the zone through Reg
(you can enter only Begin, only End or both Begin and End) - the pattern will then only be searched when a matching begin or end is loca

For example, in a Cobol program, you want to search all code for all instances of PERFORM that can be found between IF and END-IF. T
the following:

- The limits of the zones you are searching (i.e.: the Begin and End Expressions) MUST NOT be located in comments.
- Zones can overlap one another and a zone can be included within another zone.
- Each zone will be searched independently for its own regular expression.

Replace Pattern - optional

Activating this option enables you to apply a **replacement process** to the results of the Regular Expression search prior the results being
Analysis Service schema:

How does it work?

Each time the Regular Expression is matched in the source, the chosen replacement string is produced and is used to match the name (w
/over/whole match options). Replacement is based on Regular Expression grouping. For example:

- Using the Regular Expression R: **([a-zA-Z_])([a-zA-Z_0-9]+)** each parentheses pair generates a grouping and that grouping can be r
notation \1, \2, \etc., \n. If R matches the text **"The_Cat"** then **\1** is the character **"T"** and **\2** is the string **"he_Cat"**.
- Using the Regular Expression S: **(HisFunc|MyFunc)\(([^)]*)\).** If S matches the text **"HisFunc(his_parameter)"**, then **\1** is **"HisFun**
ameter".

These examples illustrate how this feature could be used:

1)

- Regular Expression entered: **LoadLibrary\("(([^"\r\n]|""|\\")*)\.dll"\)**
- Replacement entered: **\1**
- This combination will match the name of the DLLs (without the extension) called in C/C++ source code

2)

- Regular Expression entered: **com\.my_package(\.([a-zA-Z_][a-zA-Z_0-9]*))+**
- Replacement entered: **\2**
- This combination matches the last part of qualified names beginning with **com.my_package**

3)

- Regular Expression entered: **Id(d|x)_Object_([a-zA-Z_][a-zA-Z_0-9]*)**
- Replacement entered: **Id\1_\2**
- This combination matches the the names that have the form "Idd_Object_Something" and then eliminates the "Object" in the middle
- E.g.:
  - Idd_Object_Frame -> Idd_Frame
  - Idd_Object_Window -> Idd_Window
  - Idx_Object_Button -> Idx_Button

4)

- Regular Expression entered: **System.loadLibrary\([\t]*"([^"]+)"[\t]*\)**
- Replacement entered: **\1.dll**
- This combination matches the names of the C libraries used to call functions via Java native methods

Order of events during execution

A source code analysis has already been carried out and the Analysis Service schema contains the objects resulting from this analysis. A is then created and the **Replace Pattern** option is activated and a replacement text entered in the field. When the Reference Pattern is th occurs:

- The string chosen as the replacement string is checked for validity, then one of the following occurs:
  - If the chosen replacement string is not valid, the Reference Pattern cannot be executed. The replacement string is considered i e. nothing entered in the field) or if it contains a reference to a non existent grouping (i.e. \4 whereas the Regular Expression co groups).
  - If the chosen replacement string is valid, the Reference Pattern will then be run. During the process, each time a match with the is located in the selected objects, it is transformed using the chosen replacement string. The result of this transformation is then names/full names/paths of the Target objects. For each object "A" whose name matches the result of the transformation, a link between the object containing the Regular Expression match and object "A".

Limitations

- The maximum number of groupings that you can reference is 9, which means that if the you specify **\10** as a replacement it will be in ed by the character **"0"** (zero).

Notes

- It is worth remembering that if the grouping is the object of a repetition then only the last match in the grouping will be retained. Take
  - Regular Expression entered: **[a-zA-Z_]([a-zA-Z_0-9-])***
  - Match string: **James_BrowN -** Replacement entered: **\1**

In this case, **\1** corresponds to "**N**" and not to "**ames_BrowN**" Thus there is a difference between the following two Regular Expressions u match string and replacement:

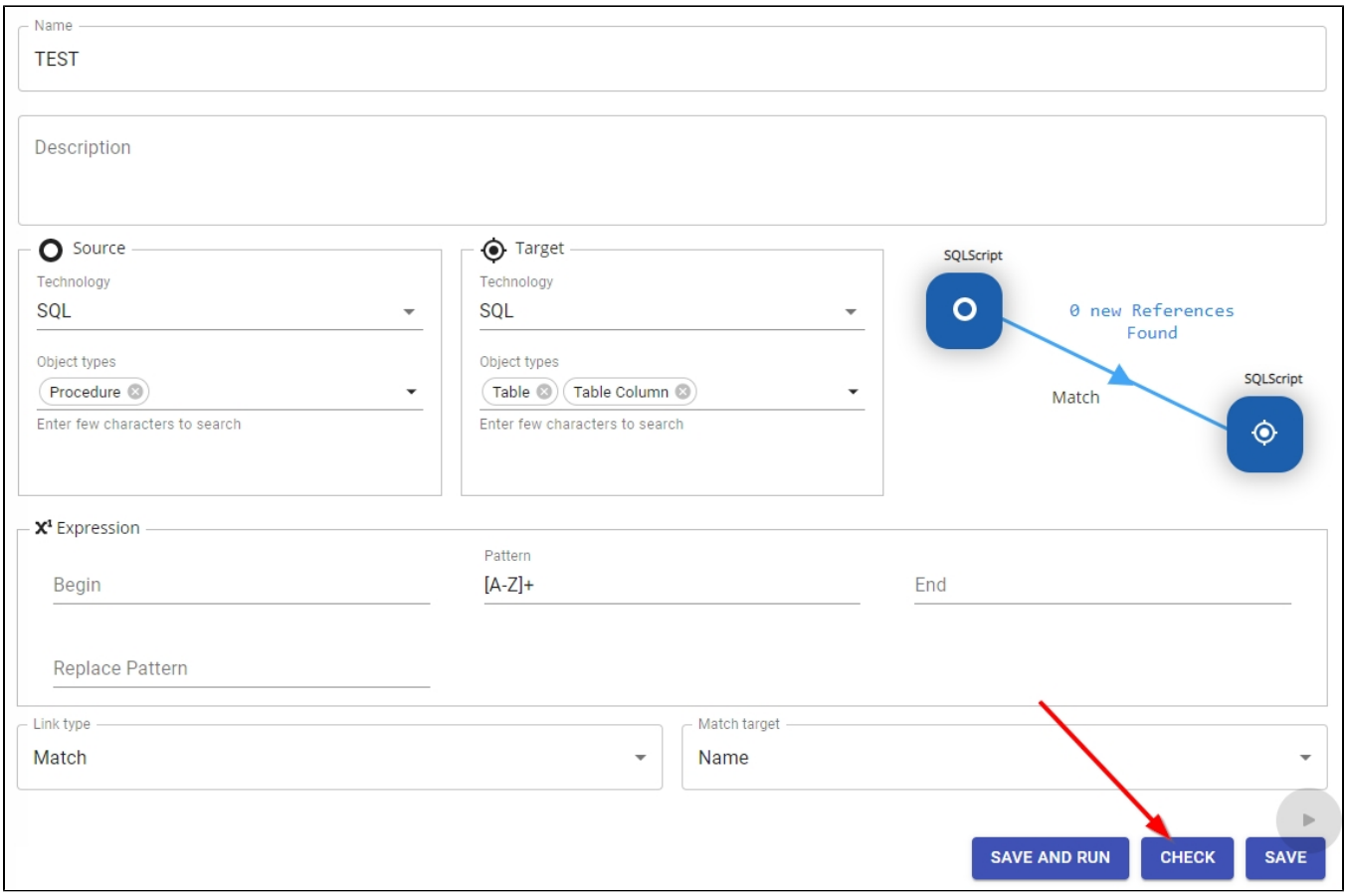
- **[a-zA-Z_]([a-zA-Z_0-9-])* = "N"**
- **[a-zA-Z_]([a-zA-Z_0-9-]*) = "ames_BrowN"**

Link Type	Use this option to select the link that will be created between the Source and the Target objects: 
Match target	This section enables you to define what the results of the **Source** search will be matched to in the **Target**. Choose from: • **Name** > The object's short name • **Full Name** > The object's full name as stored in the Analysis schema
Save and Run	See **below**.
Check	See **below**.
Save	See **below**.

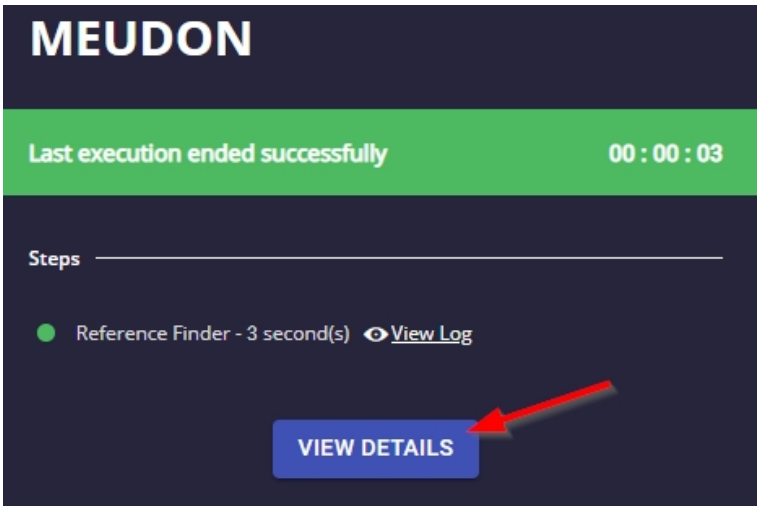
Run the Reference Finder rule manually

The **Check** button will run the Reference Finder check on the configuration you have entered (Source and Target technologies / object types and the Regular Expression pattern) - the **configuration will not be saved** and **no links will generated using this option.** This is purely a "**preview**":

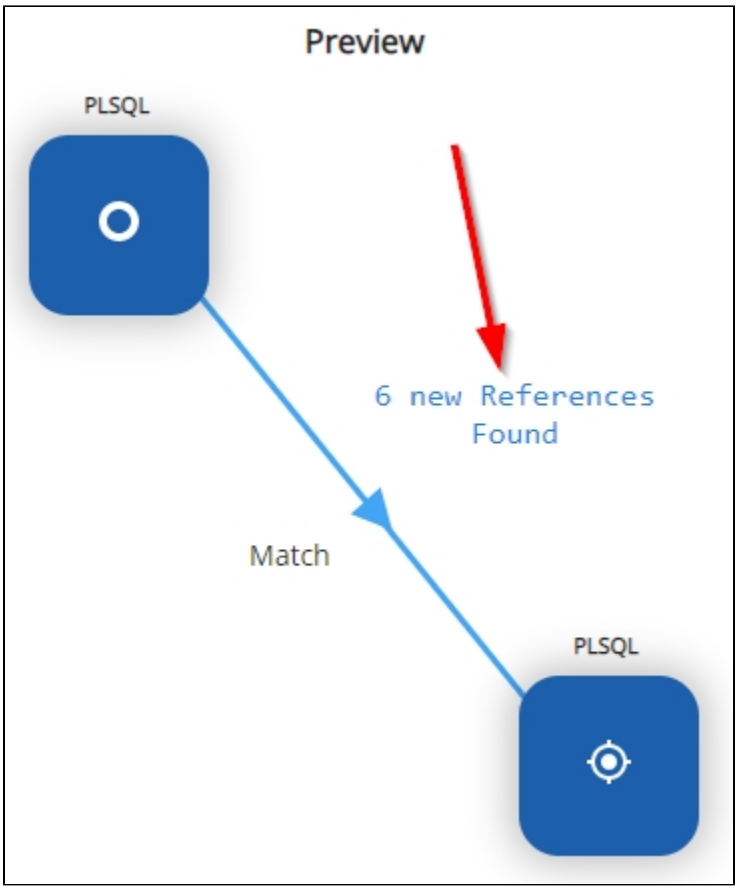
Click to enlarge



Results are available via the VIEW DETAILS button:



And in addition, any items that match this configuration will be displayed in the left hand **preview box**:



Click the **X new References Found** to view the links that have been identified:

Caller name	Caller type	Callee name	Callee type
GETAUTHORS	CAST_Oracle_Procedure	AUTHORS	CAST_Oracle_RelationalTable
INSERT_SALESDETAIL_PROC	CAST_Oracle_Procedure	SALESDETAIL	CAST_Oracle_RelationalTable
INSERT_SALES_PROC	CAST_Oracle_Procedure	SALES	CAST_Oracle_RelationalTable
LISTEAUTHORS_PROC	CAST_Oracle_Procedure	AUTHORS	CAST_Oracle_RelationalTable
ADD_DEFECT	CAST_Oracle_Procedure	DEFECTS	CAST_Oracle_RelationalTable
CALC_NUMBER_DEFECTS	CAST_Oracle_Procedure	DEFECTS	CAST_Oracle_RelationalTable

Rows per page: 10 1-6 of 6

Search

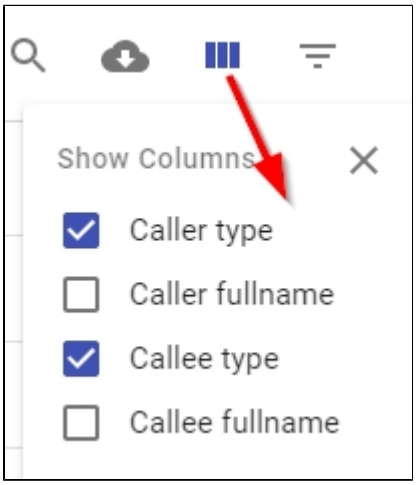
Use this option to search for a specific link in the list. Search is performed on the **Caller Name / Callee Name** fields.

Download

Use this option to download a CSV file containing the results.

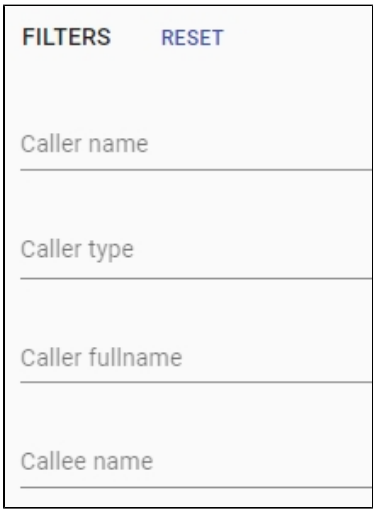
View Columns

Choose the columns you want to display:



Filter

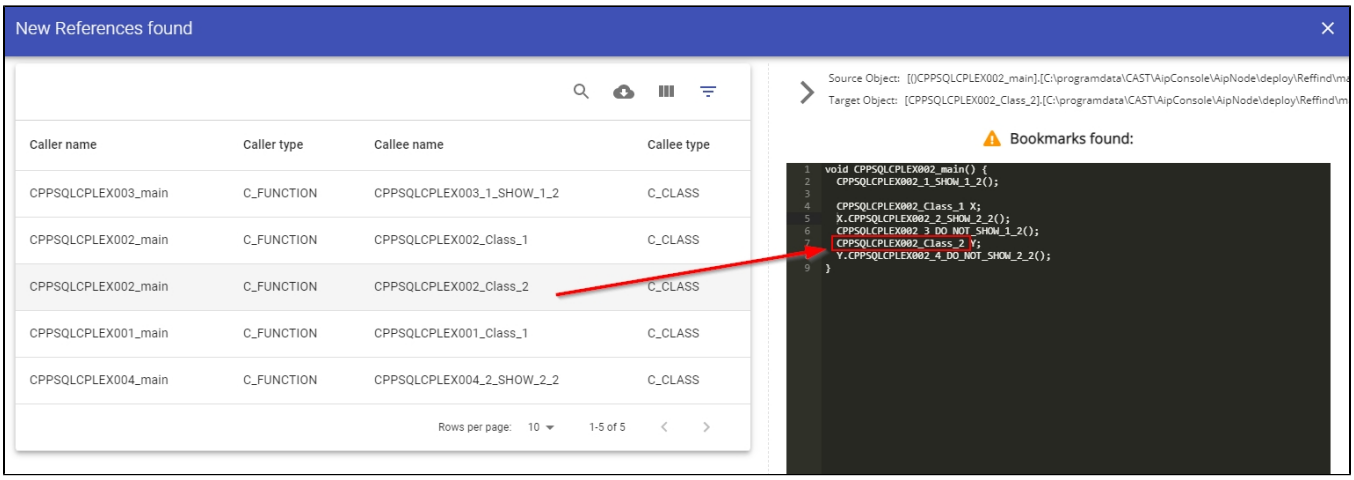
Use this option to filter the display:



Caller name	The name of the **Caller object** - i.e. the object containing the code that has been matched by the Regular Expression.
Caller type	Caller object type.
Callee name	The name of the **Callee object** - i.e. the target object.
Callee type	Callee object type.

Clicking a link in the list will also display the code of the Caller object in which the match has been located:

Click to enlarge



Save the Reference Finder rule

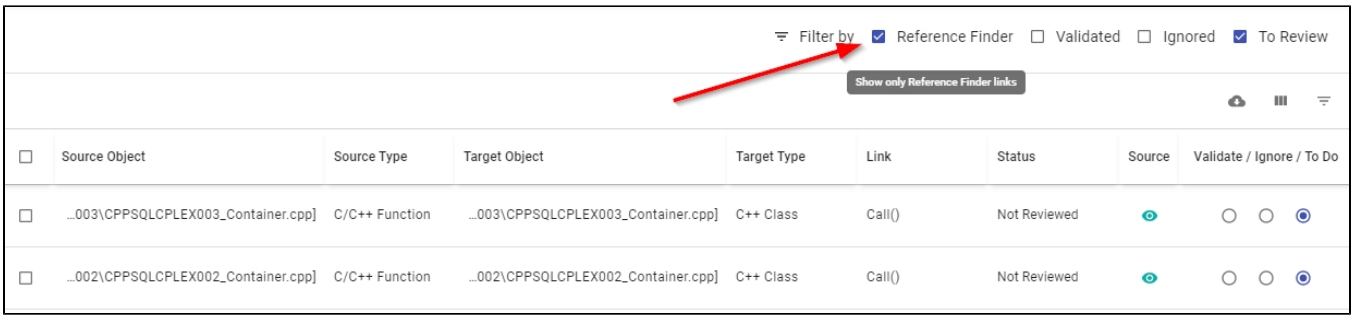
There are two methods to save the Reference Finder rule:

- **Save and Run** > The Reference Finder rule configuration will be saved (creating a new rule or updating an existing rule) AND the rule will be executed and any links resulting from the rule will be generated.
- **Save** > The Reference Finder rule configuration will be saved (creating a new rule or updating an existing rule) only.

When using either option, the rule will be displayed in the list of Reference Finder rules:

When is a Reference Finder rule run?

Any rule listed in the Reference Finder section (i.e. that has been saved) will be run **the next time an analysis is run**. This means that any links that are identified by any of the rules that have been saved will be created and stored as part of the analysis results. Links will appear in the **Application - Config - Summary of Dynamic Links** as they are classed as "dynamic":


Filter by
Reference Finder
Validated
Ignored
To Review
Show only Reference Finder links
Source Object
Source Type
Target Object
Target Type
Link
Status
Source
Validate / Ignore / To Do
...003\CPPSQLCPLEX003_Container.cpp]
C/C++ Function
...003\CPPSQLCPLEX003_Container.cpp]
C++ Class
Call()
Not Reviewed
...002\CPPSQLCPLEX002_Container.cpp]
C/C++ Function
...002\CPPSQLCPLEX002_Container.cpp]
C++ Class
Call()
Not Reviewed