

Tools - Smart Reference Finder

What's new?

Please see [Smart Reference Finder - Release Notes](#) for more information.

Purpose

This is a standalone alternate Reference Finder that allows users to save, load, edit and test their configurations, execute the search and then view the results in a table linked to a code viewer showing the bookmarks. Past execution result sets are stored and can be reviewed, and corresponding configurations can be restored for editing or launching.

It supports both search string mode and regular link mode, and provides a few more filters than the original embedded Reference Finder, as well as a slicing feature that adapts the search to the data volume and available memory, thus reducing global processing time.

A detailed (if verbose mode is on) execution log is available in `%APPDATA%\CAST\CAST\Smart Reference Finder\Logs`

The tool is written in Java, and delivered as a set of two Jar files :

- **CAST-ReferenceFinderEngine.jar** : this is the command-line application, it takes a configuration **properties** file as input
- **CAST-SmartReferenceFinder.jar** : this is the GUI application. It enables users to view results and refine the configuration through the GUI, then save it as a **properties** file for use as input for the command-line application

Since the results are stored in dedicated SQL tables, this tool should always be used along with a **KB Update Tool**, in order to have the results reflected in the analysis service.

Applicable in CAST Version

Release	Yes/No
8.3.x	✓

Applicable RDBMS

RDBMS	Yes/No
CSS3	✓
CSS2	✓

Details

Prerequisites

Follow the below prerequisites before running the tool:

Java

Java 8 (or higher) must be installed on the machine where you launch the tool

Objects in analysis service

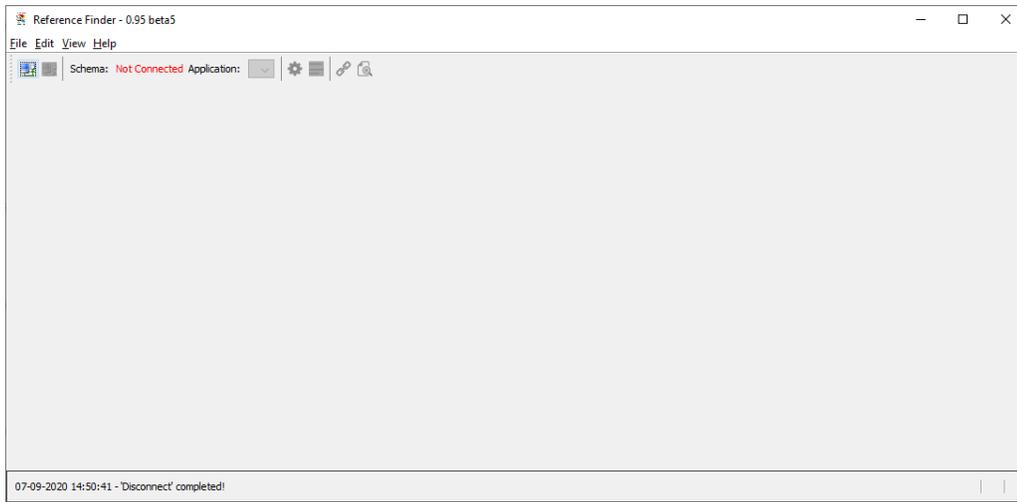
The application must have been analyzed first, so that analysis service is populated with objects

CAST-SmartReferenceFinder (GUI)

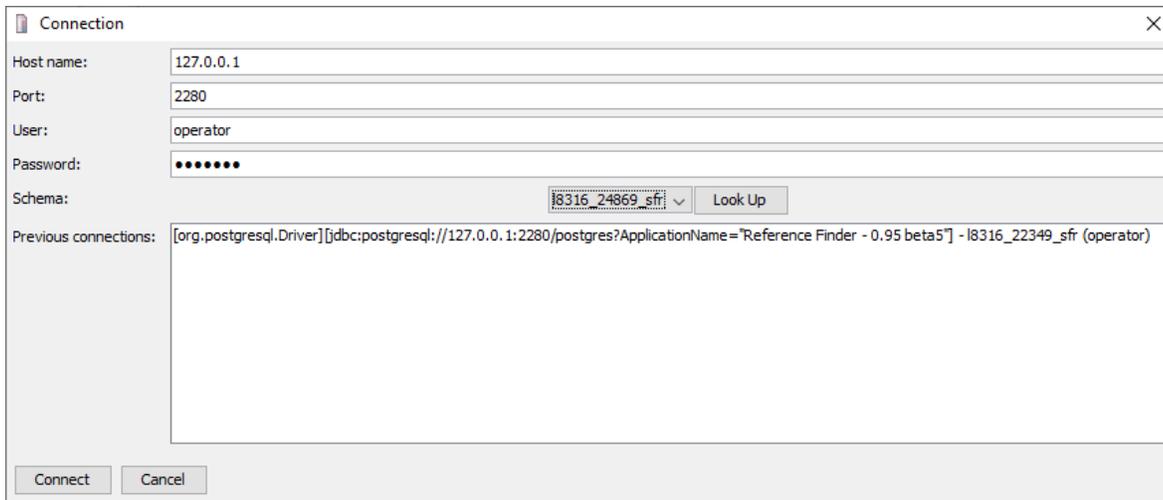
Thanks to the GUI Application, users can build, edit, test and fix their configurations. Once the configuration gives the expected results, then users can save it to a properties file that can be provided as input for the command-line application.

The command-line application and the properties file can then be integrated into an extension along with a KB Update Tool, in order to insert the results into the analysis service.

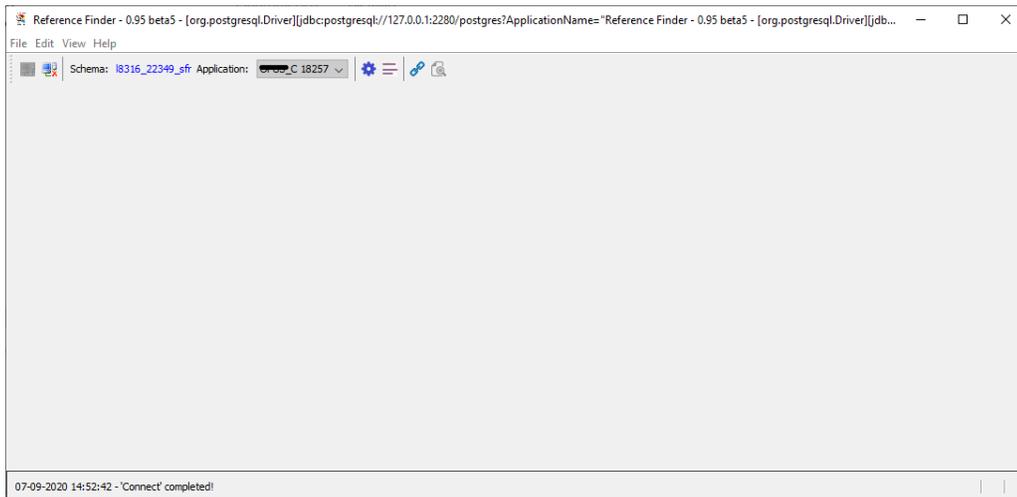
Double-click `CAST-SmartReferenceFinder.jar`



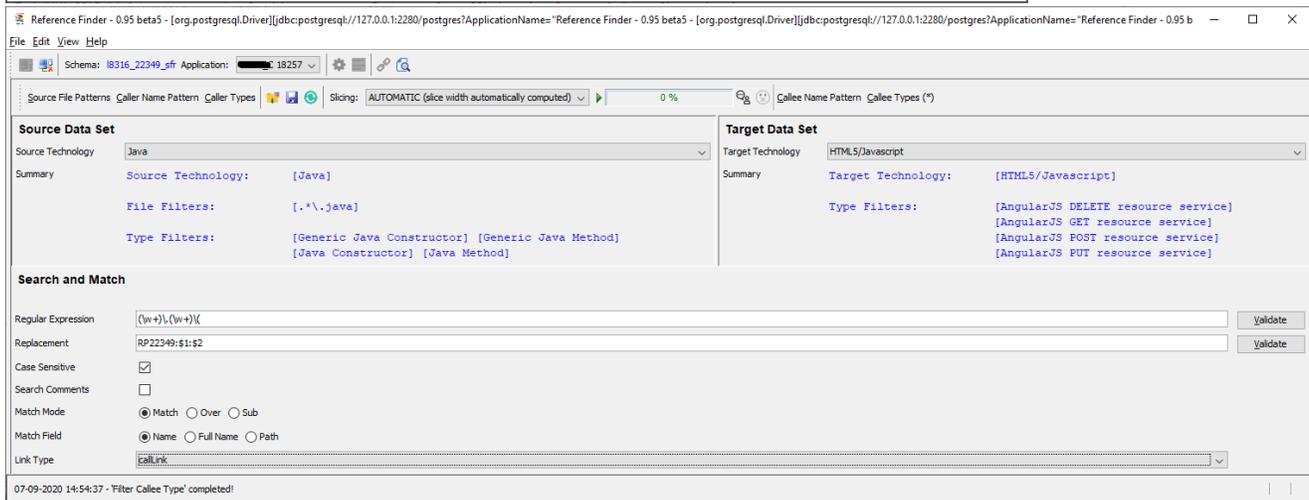
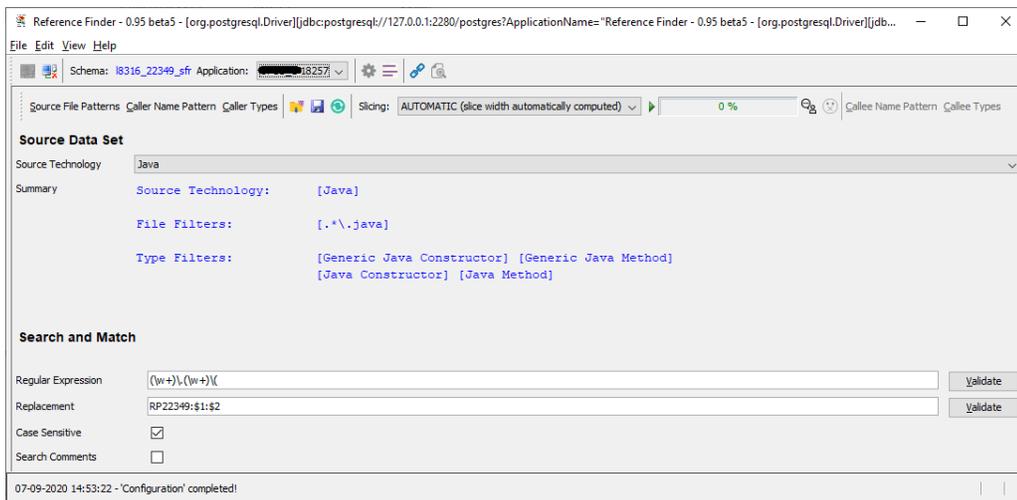
The first step is to connect to the appropriate analysis service. Click the "Connect" icon () , a pop-up appears



Once connected, user has the choice between opening the Configuration panel () and the Results panel () , and between the Search String mode () and the Regular Link mode ()



Configuration Panel



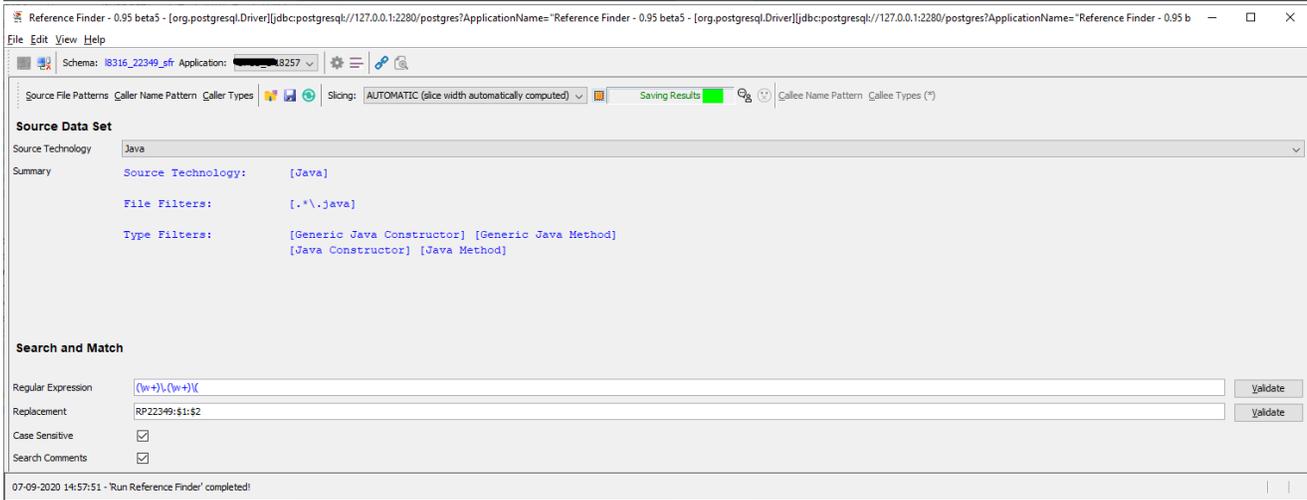
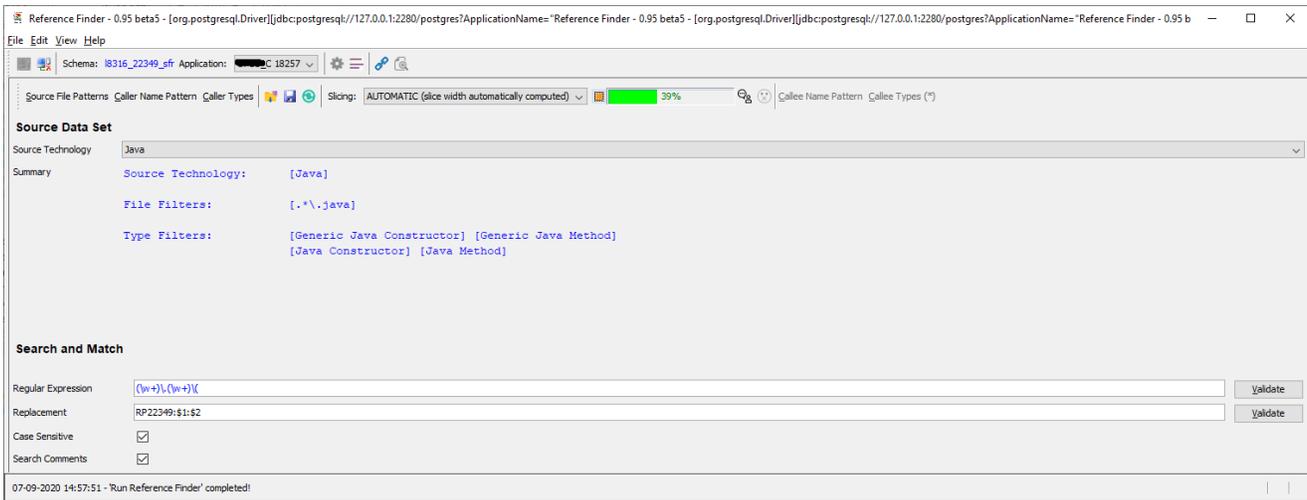
User can edit their configuration :

- Choose the technologies
- Possibly add filters on source file patterns, object names or object types
- Write the regular expression, and validate it
- Possibly write a replacement regular expression and validate it: **unlike in Cast-MS, the matching groups are referred to as : \$1, \$2 etc... and not \1, \2....**
- Choose whether the regular expression is case sensitive or not
- Choose whether Reference Finder should also search in comments
- In Regular Link mode only, choose the link type
- In Regular Link mode only, choose the target field to match with the regular expression (name, full name or path)
- In Regular Link mode only, choose the target matching mode (match, over or sub)
- In Regular Link mode only, choose if the target matching is case sensitive or not

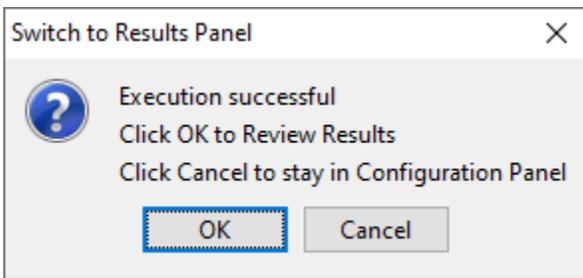
Configuration can be saved () , loaded from disk () or reset ()

Verbose mode () or non-verbose mode () can be selected.

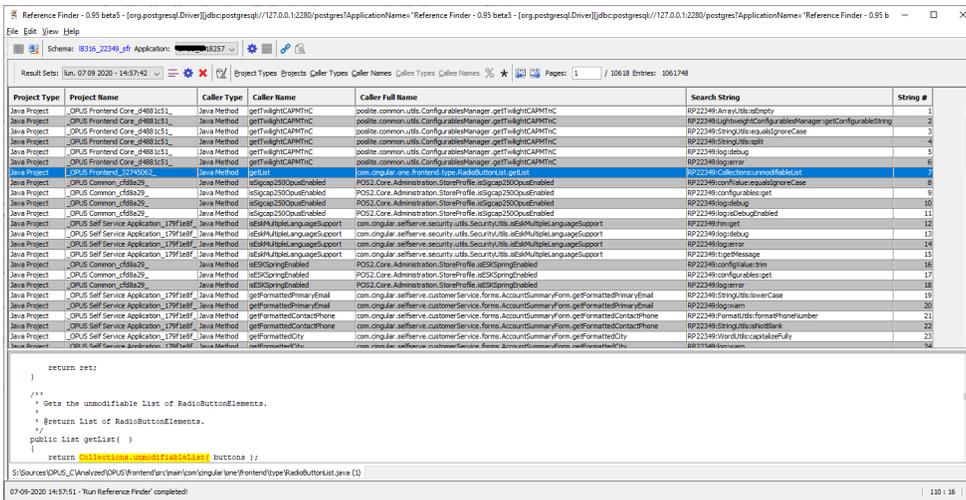
Then user can launch the Reference Finder by clicking  :



Once finished, a pop-up suggests to switch to the Results Panel in order to view the results of the execution:



Results Panel



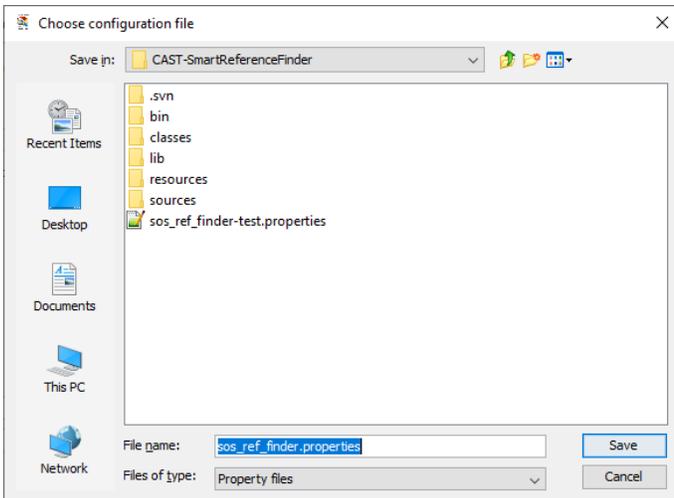
The Results Panel has a drop-down list of all execution result data sets by date and time.

Results can be viewed (🔍) and purged by data set (✖), or purged globally (🗑️). It is also possible to load the configuration that generated the selected data set (⚙️).

The result table is paginated (📄), and there also are filters by names and types. The filters can follow SQL Patterns (🔎) or Posix regular expressions (🌟).

Save the configuration into a specific place and reuse the file with the command-line application

Once user has edited their configuration, launched Reference Finder and reviewed the results, if they are satisfied they should save the configuration to a specific file :



The file can now be reused as an input parameter for the command-line application

Possibly note the slicing value

If user configured the slicing manually, then they should just note the value they chose. The same value will have to be provided as input for the command-line application.

If user selected "Automatic slicing" then they may want to open the execution log file and see the slice width that was computed.

%APPDATA%\CAST\CAST\Smart Reference Finder\Logs\sos-ref-finder-1599483462874.log

```
2020-09-07 14:57:52.045 - nbItems = 331395
2020-09-07 14:57:52.233 - Available memory: 744797011
2020-09-07 14:57:52.233 - Volume: 5733493 lines
2020-09-07 14:57:52.233 - Autoselected slicing: 153746
2020-09-07 14:57:52.233 - nbSlices = 3
2020-09-07 14:57:52.233 - lastSliceWidth = 23903
```

Here user may note that the computed slice width was 153746. This is not mandatory : if **slice** argument is forgotten, then an automatic slicing will be recomputed by the command-line application based on the available memory

Cast-ReferenceFinderEngine.jar (command-line application)

Usage

Cast-ReferenceFinderEngine.jar is a command-line application, that needs to be launched in a CMD using java.exe

Mandatory arguments :

- **host** : the hostname of the CSS server where the analysis service is installed
- **kb**: the name of the analysis service
- **log** : the reference finder log file
- **props** : the properties file containing the Reference Finder configuration

Optional arguments :

- **app** : the application present in analysis service on which the Reference Finder should be run (if analysis service contains several applications, then user should provide this argument)
- **port** : the port of the CSS server where the management base is installed (default : 2280)
- **user** : the user name used to connect to the CSS server (default : operator)
- **password** : the password used to connect to the CSS server (default : CastAIP)
- **verbose** : more details are given in the log file (default : False)
- **slice**: the maximum number of source objects processed in a slice (default : automatically computed)
- **id**: the session id that will be related to this execution in the results table (default : current timestamp)

Open a CMD window, and navigate to the folder where you installed CAST-ReferenceFinderEngine.jar. Type the following command-line, using the configuration file you saved with the GUI and possibly the slicing you read in the GUI log file. As a SESSION_ID, user should put a long number that can be retrieved, so that the ulterior KB Update Tool can refer to this execution results set only :

Command-line

```
java -jar CAST-ReferenceFinderEngine.jar -host <CSS_SERVER> -kb <ANALYSIS_SERVICE> -log <LOG_FILE> -props
<CONFIGURATION_PROP_FILE> [-id <SESSION_ID>] [-slice <SLICING>]
```

Integration with a KB Update Tool

User now has the proper command-line to fill-in the Smart Reference Finder dedicated tables with expected results (MAINT_TMP_REF_PATTERN_SEARCH_STRING and MAINT_TMP_REF_FINDER)

They now need to copy (or transform) the data into real analysis service tables. That is why a KB Update Tool is needed.

This integration can be done in an extension such like **com.castsoftware.labs.tools.toolsafteranalysis**

Example of KB Update Tool for Regular Links :

Regular Links KB Update Tool

```
INSERT INTO ci_links
(
    caller_id,
    called_id,
    link_type,
    error_id
)
SELECT mtrf.caller_id,
mtrf.callee_id,
mtrf.link_type,
0
FROM MAINT_TMP_REF_FINDER mtrf
WHERE mtrf.session_id = <SESSION_ID>;
```

Example of KB Update Tool for Search Strings :

Search Strings KB Update Tool

```
INSERT INTO ci_links
(
    caller_id,
    called_id,
    link_type,
    error_id
)
SELECT ss.caller_id,
methods.idkey,
'accessExecLink',
0
FROM MAINT_TMP_REF_PATTERN_SEARCH_STRING ss
JOIN Keys methods
ON methods.keynam = Split_part(ss.string, ':', 3) as method_name
AND methods.ObjTyp = 102
JOIN KeyPar kp
ON kp.idkey = methods.idkey
JOIN Keys classes
ON classes.idkey = kp.idparent
AND classes.ObjTyp = 100
AND classes.keynam = split_part(ss.string, ':', 2) AS class_name
WHERE ss.session_id = <SESSION_ID>
AND ss.string like 'RP1:%';
```

Notes/comments

The tool is available here :

[CAST-SmartReferenceFinder.0.95-beta6.7z](#)

Related Pages