

Enlighten - Links

Purpose (problem description)

This page will guide you to troubleshoot issues with Links in Enlighten. For more information regarding links in Enlighten, refer to:

- For CAST 8.3.x: [ENL - Links](#)

Observed in CAST AIP

Release	Yes/No
8.3.x	

Observed on RDBMS

RDBMS	Yes/No
CSS	

Action Plan

In order to start treating issues with links between objects in Enlighten you will first have to correctly identify the linked objects

1. Open Enlighten and search for the objects in the knowledge base as described in [ENL - Finding/locating objects in the Knowledge Base](#). The search results return objects based on your criteria in regards to the Objects Names, Full Names and Types.
2. Place all objects retrieved from previous step on the Enlighten Graphical View, open the objects properties view as described in [ENL - Viewing properties](#) and retrieve the objects source file path/name.
3. Open pgAdmin, [connect](#) to the knowledge base and run the queries found in [SQL Queries - CAST Knowledge Base - Queries on objects - How to get the ID of an object from its object name or full name](#) in order to retrieve the ObjectID for the objects. You can also retrieve the ObjectID by pressing CTRL+ALT+SHIFT+D and Zoom button(Zoom down or Up) in Enlighten.
4. Using the information from the above steps(**full name, source code path and object type**) in most cases and technologies you should be able to identify the objects. In some object oriented languages (**Java, .C#/VisualBasic,C++ etc**) and due to method overloading this info may not be enough. In these cases 2 objects can have the same full name, source code path and object type.
 - a. If you have identified the objects, remove all other objects from the Enlighten Graphical View.
 - b. If you still have not identified the objects, using the retrieved ObjectID obtain the [GUID information](#) for each object.
 - i. The object signature in the object GUID contains information concerning the source code file used and the signature of the method with method parameters in the cases of object . Take the below example where we have 2 Java methods referring to the same source code file:

```
96638; "setName"; "99?com?99?mts?99?mts?99?domain?99?common?99?search?99?xmlobjects?620?s:
/sources/mts/analyzed/mts/src/com/domain/common/search/xmlobjects/input.java?100?Input?
102?setName(99?java?99?lang?100?String,99?java?99?lang?100?Integer) "
19664; "setName"; "99?com?99?mts?99?mts?99?domain?99?common?99?search?99?xmlobjects?620?s:
/sources/mts/analyzed/mts/src/com/domain/common/search/xmlobjects/input.java?100?Input?
102?setName(99?java?99?lang?100?String) "
```

The GUID of the objects is different since it takes into account the different method parameters. In the first case three parameters are String+integer types and in the second just String types. with this info and the source code file you can identify the correct object

- ii. Remove all other objects from the Enlighten Graphical View

5. After completing the above steps, you have an Enlighten Graphical view with 2 objects and you should be able to [to view a link](#) if it exists. In case there is no link between the objects or an unexpected link between the objects this could be a case of missing/wrong links. You should check if the link exists in the corresponding knowledge base. Run the following query on your knowledge base by replacing CALLER_OBJECT_ID by the object id of the calling object and CALLEE_OBJECT_ID by the object id of the called object:

Possible link candidates

```
SELECT kClr.idkey as CALLER_ID, kClr.keynam as CALLER_NAME, ofnClr.fullname as CALLER_FULL_NAME, tClr.
typnam as CALLER_TYPE_NAME,
tClr.typdsc as CALLER_TYPE_DESCRIPTION, kCle.idkey as CALLEE_ID, kCle.keynam as CALLEE_NAME, ofnCle.
fullname as
CALLEE_FULL_NAME, tCle.typnam as CALLEE_TYPE_NAME, tCle.typdsc as CALLEE_TYPE_DESCRIPTION
from ACC a
join KEYS kClr ON kClr.idkey=a.idclr
JOIN KEYS kCle ON kCle.idkey=a.idcle
JOIN OBJFULNAM ofnClr ON ofnClr.idobj=kClr.idkey
JOIN OBJFULNAM ofnCle ON ofnCle.idobj=kCle.idkey
JOIN TYP tClr ON tClr.idtyp=kClr.objtyp
JOIN TYP tCle ON tCle.idtyp=kCle.objtyp
WHERE kClr.idkey='<CALLER_OBJECT_ID>'
AND kCle.idkey='<CALLEE_OBJECT_ID>';
```

The query will return 0 rows or 1 row.

- a. If it returns 0 rows
 - i. If the the link is visible in Enlighten Graphical View then there is a problem with Enlighten itself. In that case, contact [CAST Technical Support](#), with the below [Relevant Input](#)
 - ii. If the link is not visible in Enlighten Graphical View but the link is expected, than this is a problem of missing link. In this case, see [CMS Snapshot Analysis - Run Analyzer - Missing or Wrong links between two objects](#)
 - b. If it returns 1 row
 - i. If the the link is not visible in Enlighten Graphical View then there is a problem with Enlighten itself. In that case, contact [CAS T Technical Support](#), with the below [Relevant Input](#)
 - ii. If the link is visible in Enlighten Graphical View but the link is not expected, than this is a problem of wrong link. In this case, see [CMS Snapshot Analysis - Run Analyzer - Missing or Wrong links between two objects](#)
6. If you do not find the information you are looking for or solution for your problem, in this page, contact [CAST Technical Support](#). and provide below [relevant input](#) to help CAST Support team investigate your problem or procure the required information.

Relevant input

- [CAST Support Tool \(CST\) - alias Sherlock](#) with the options **Logs** and **CAST Local** and **Management Databases** Checked
- A screenshot in Enlighten showing the objects names, the objects types, the link type
- Explain what is expected and the step-by-step scenario

Notes/comments

Related Pages