

Conversion Tools - Bug #2357

callgraph - set schema triggers as entry points

08/04/2014 07:31 AM - Constantin Asofiei

Status: New	Start date:
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 8.00 hours
Target version:	case_num:
billable: No	
vendor_id: GCD	
Description	
Related issues:	
Related to Conversion Tools - Feature #2251: improve the call graph generation	Closed 03/03/2014
Related to Database - Support #2371: trigger behavior	Closed

History

#1 - 08/04/2014 07:33 AM - Constantin Asofiei

Currently schema trigger procedures are not automatically set as entry points - they need to be manually added to the root procedure list. Create a way of automatically adding these schema trigger procedures (collected from the schema) as entry points; determine if this should be done optionally (or maybe just list them in a root-list.xml format, and allow the user to review/include them as needed?)

#2 - 08/04/2014 07:33 AM - Constantin Asofiei

Original issue is in #2260 note 14.

#3 - 08/04/2014 09:10 AM - Greg Shah

Since these triggers are explicitly coded into the schema file, we must assume these are valid entry points and automatically add nodes for them IF the 4GL code that is reachable from the explicitly configured root nodes actually would cause the triggers to be fired. We should assume that these are in use, since that is the most common case.

It is possible that some or all of these are not in actual use in production and the configuration has been accidentally left behind. But in this case the external procedures would have to have been removed from the project (because by definition, if they are physically there and configured, then they will be executed in production). In such a case, the person running the call graph processing will see the missing schema triggers listed and they can decide to cleanup the schema as needed.

In regard to how we can determine if the triggers will ever be invoked, Eric will be documenting the exact rules for these cases. The only tricky part will be that we will need to know the transaction properties and buffer scopes in order to properly detect the linkage. That just means we have to move a portion of annotations processing to be executed earlier.