

TCC - Merge/Unsplit objects

Merge/Unsplit objects

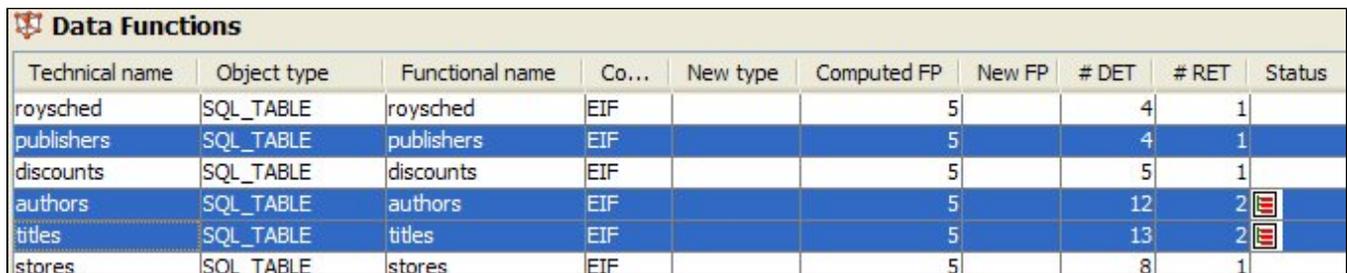
CAST provides two manual mechanisms that can be used to group objects together. They are similar, but different:

- **Merge** > A merge can be used with either Data Functions or Transactional Functions. Any number of objects can be merged together.
- **Unsplit** > An unsplit can ONLY be used with Transactional Functions that have already been [split](#). It is equivalent of reversing the split action. All objects added during the split action are removed from the results and the split Transactional Function returned to its normal state - i.e. it encompasses all its sub-objects and called objects.

Merging objects

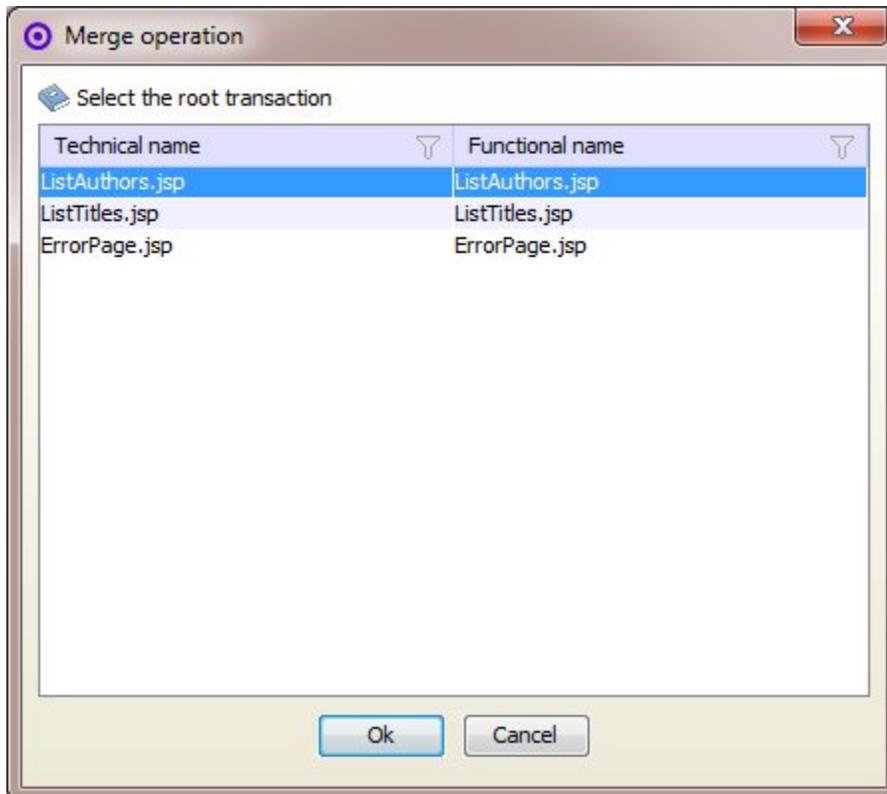
The process of merging Data Functions and merging Transactional Functions is identical.

- Move to the [AFP Calibration node - Right hand panel](#) to begin calibration
 - Choose either the **Data Functions** or the **Transactional Functions** sub node
- Hold down the CTRL key and select (with the mouse) the objects you want to merge so that they are highlighted as follows:



Technical name	Object type	Functional name	Co...	New type	Computed FP	New FP	# DET	# RET	Status
roysched	SQL_TABLE	roysched	EIF		5		4	1	
publishers	SQL_TABLE	publishers	EIF		5		4	1	
discounts	SQL_TABLE	discounts	EIF		5		5	1	
authors	SQL_TABLE	authors	EIF		5		12	2	
titles	SQL_TABLE	titles	EIF		5		13	2	
stores	SQL_TABLE	stores	EIF		5		8	1	

- Right click any of the selected objects and choose **Merge**
- You will now be prompted to choose which object is the "root" item:



- Choose the root and click **OK**.
- The root object will be the object that is displayed in the CAST Transaction Configuration Center, other objects will not.
- The **Status** column for the root object will display as follows, indicating that the object was merged (see [Data Functions/Transactional Functions nodes](#) for more information about the icons in the **Status** column):

RET	Status
1	
1	
1	
1	

Note that the split is not permanent until you **Save** the changes to the server (see the [Toolbar](#) for more information about this).

How are Function Point, DET, FTR and RET values calculated when multiple transactions or data functions are merged?

When you merge multiple transactions or data functions you will be requested to choose a "root" item (as explained above). The values of the root item will be used to represent the block of transactions or data functions in the GUI immediately after you have completed the merge. However, to ensure that the "correct" values (i.e. the "sum value" of all merged items) are taken into account going forward, you will need to use the [Compute](#) action (note that the "sum value" is not exactly equal to the sum of all items in all cases since some columns/tables could be identical and will therefore count as 1).

Unsplit Transactional Functions

The **Unsplit** mechanism is only available for **Transactional Functions**, not Data Functions. It enables you to effectively reverse a [split](#). All objects added during the split action are removed from the results and the split Transactional Function returns to its normal state - i.e. it encompasses all its sub-objects and called objects.

- Move to the [AFP Calibration node - Right hand panel](#) to begin calibration
- Choose either the **Transactional Functions** sub node
- Find a Transaction that has already been split:

Transactions									
Technic...	Object type	Function...	Compute...	New t...	Comput...	Status
NewSales...	CAST_Web_File	NewSales.jsp	EO_EQ		0	0			
getParam...	JV_METHOD	getParam...	EO_EQ		0	0	0		*
HttpServlet	JV_CLASS	HttpServlet	EO_EQ		0	0	0		*
transpare...	CAST_Web_File	transparen...	EO_EQ		0	0	0		*
title.gif	CAST_Web_File	title.gif	EO_EQ		0	0	0		*
logo_ani.gif	CAST_Web_File	logo_ani.gif	EO_EQ		0	0	0		*
bg_orang...	CAST_Web_File	bg_orange...	EO_EQ		0	0	0		*
web.xml	JSP_APPDESCRIPTOR	web.xml	EO_EQ		0	0	0		*
test_sale...	CAST_Javascript_ClientSi...	test_sales...	EO_EQ		0	0	0		*
display_d...	CAST_Javascript_ClientSi...	display_date	EO_EQ		0	0	0		*
Connect.jp...	CAST_Web_File	Connect.jp...	EO_EQ		0	0	0		*

- First you need to **Undelete** otherwise you will be prevented from unsplitting it - right click the Transactional Function and select **Undelete**.
- Next, right click the Transactional Function and select **Un-split**.
- All objects added during the split action will be removed from the results and the split Transactional Function will be returned to its normal state - i.e. it encompasses all its sub-objects and called objects.

Transactional Function and Data Function recovery when their root object disappears

When merging Transactional Functions or Data Functions, one of the Transactional Functions / Data Functions is set as the **root** item of the group. The **root** represents the group of merged items. If the entry point / main table of the root is **no longer available**, the associated root Transactional Function / Data Function will also no longer be available and will be dropped.

In this case, when the root group includes more than one item (excluding the root), then the group of merged items will remain but the root item will be replaced by the first available item in the list of items.

If the root group includes only one item, then the merged item becomes a single Transactional Function / Data Function.

As a consequence of this, the previous root item will be marked as **deleted** and the new root item representing the merged items will be marked as **added** after a snapshot is generated.

If the entry point / main table of the root item that was previously missing is then recovered, then the Transactional Function / Data Function which was associated to this will be recovered as the same object as before. However, it won't automatically become the root item of the merged items - this is because the merging information is lost. If the merge operation was achieved through a custom procedure, this merge may return to its initial state, otherwise if the merge was done manually, it has to be redone manually to get back to its initial state.

