

---

Subject: Access non key data from parent table  
Posted by [gpatti](#) on Sun, 31 Aug 2008 10:11:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I am trying to use the ADD4 transaction from a LIST2 transaction to add multiple rows of data to a table based on a parent record selected. However, to build the select clause for the data I want to add I need to know the value of a field from the selected parent row that is not a key. I understand that only the key fields are passed in context to the ADD4 (child) transaction.

How can I either pass the non-key data to the child ADD4 transaction or access the parent table to retrieve it from the child transaction?

Thanks

Graham

---

---

Subject: Re: Access non key data from parent table  
Posted by [AJM](#) on Sun, 31 Aug 2008 13:20:59 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

When you say that "only the key fields are passed in context to the child transaction" you are describing the default behaviour, but like most aspects of the Radicore framework this default can be overridden.

The string which is inserted into \$selection by the parent task and then passed to the child task is constructed by taking a list of field names (by default the primary key) then combining it with the values from the selected record(s) to form a string in the format:

(field1='value1a' AND field2='value2a') OR (field1='value1b' AND field2='value2b')

This list of field names is initially populated by using the getPkeyNames() method, but it can be modified by inserting code into the \_cm\_getPkeyNames() method of the relevant table class.

---

---

Subject: Re: Access non key data from parent table  
Posted by [gpatti](#) on Thu, 11 Sep 2008 14:55:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I've come bacjk to this topic because I coded in a quick & dirty way and want to come back and fix this now.

I've tried using the \_cm\_getPkeyNames() method as you suggested but without success. I actually think the method is not being called for my transaction.

I'm running a list2 transaction and am listing rows from an inner table based on a row selected from the parent table. I then have a task button for an ADD4 transaction based on the inner table

(I want to add a multiple number of rows to this table).

However, in order to select correct data to use in the addmultiple I need an item of data from the previously selected parent row. I can access the primary key in \$fieldarray, but the additional field I added in \_cm\_getPkeyNames() is not there. I've tried inserting the \_cm\_getPkeyNames() in the class of the child table and of the parent table. When debugging I see the code executed correctly during the building of the list2 data, but it doesn't seem to get passed to the ADD4 transaction.

I can't figure out where I am going wrong and your insight usually gets me straight to the correct solution. Here's hoping again!

---

Subject: Re: Access non key data from parent table  
Posted by [AJM](#) on Thu, 11 Sep 2008 15:54:09 GMT  
[View Forum Message](#) <> [Reply to Message](#)

This is where you really ought to be using a debugger to see exactly what is going on instead of making wild guesses.

When you are in a LIST2 task, make one or more selections in the INNER (child) area and then press a navigation button, this is what happens:

- (1) The LIST2 controller calls the childForm() function.
- (2) This function detects that \$\_POST['select'] is not empty, so it calls the getPkeyArray() method on the INNER object.
- (3) This function calls \$object->getPkeyNames() to get the list of primary key fields as defined in the data dictionary.
- (4) It then calls \$object->\_cm\_getPkeyNames() to allow this list of field names to be customised.
- (5) It then loops through \$object->fieldarray, and for every row which was selected it constructs an associative array using the list of field names obtained previously.
- (6) This associative array is then converted into a string which is passed to the task associated with the navigation button.

If your \_cm\_getPkeyNames() method is not being called then have you checked that you put it in the right class?

---

Subject: Re: Access non key data from parent table  
Posted by [gpatti](#) on Thu, 11 Sep 2008 16:31:21 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Yes I am using a debugger Tony.

There's a slight difference in what I am doing though, from what you described. Even though I am in a LIST2 task, I am NOT making any selections in the inner area. This is because I want to add rows to the inner table, using the primary key of the parent. That part works ok. The difficulty is in building the array or rows that I want to insert - that's where I need the additional non primary

value from the parent row.

The debugger is showing that just before the GetInitialDataMultiple transaction is called the \$where is being initialised. It does contain the additional value up until this point.

From what you are saying it would appear that this initialisation is because I haven't selected any child data? (or did I misunderstand that). In any case is it possible to prevent the \$where being initialised before calling the GetInitialDataMultiple?

---

---

Subject: Re: Access non key data from parent table

Posted by [AJM](#) on Thu, 11 Sep 2008 17:36:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

The selection from the INNER entity is passed to the child task via the \$selection string, which can be customised using the `_cm_getPkeyNames()` method. The \$where string is also extracted from the same object and passed in the \$where string, which can be customised using the `_cm_getWhere()` method. If no selection is made then \$selection will be set to \$where.

You therefore need to put some code in the `_cm_getWhere()` method if you are not making any selections from the INNER entity.

I have tried navigating from a LIST2 to a ADD4 and I see the \$where string correctly passed to the ADD4 task where it is passed to the `initialise()` and `_cm_initialise()` methods. It is still there when it is passed to the `getInitialDataMultiple()` method, but is converted into an associative array before being passed to the `_cm_getInitialDataMultiple()` method. There is nothing in the standard code which loses the contents of \$where. Where does it happen in your code?

---

---

Subject: Re: Access non key data from parent table

Posted by [gpatti](#) on Thu, 11 Sep 2008 18:39:12 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

On line 42 of std.add4.inc the \$where is initialised using  
`$where = $dbobject->initialise($where);`

Up to this point the \$where contains the primary key plus the additional field I have added, but after this the \$where only contains the primary key. It is this that then gets passed to GetInitialDataMultiple.

---

---

Subject: Re: Access non key data from parent table

Posted by [gpatti](#) on Thu, 11 Sep 2008 18:42:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Reading the documentation further, would I be correct in using `_cm_filterWhere()` method to prevent the additional field being removed in the initialise?

---

---

Subject: Re: Access non key data from parent table  
Posted by [AJM](#) on Thu, 11 Sep 2008 18:58:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

You are confusing me. You said that `$where` was being initialised when in fact it is passed to the `initialise()` method, which is something else entirely.

If the input string to `initialise()` contains the values that you want, but some are being filtered out with the `filterWhere()` function, then you are correct in saying that the list of fields which should not be filtered out can be customised with the `_cm_filterWhere()` method. When you step through with your debugger the comments I have in my code should explain what each function does.

When you report a problem such as this it is important to identify where you are having the problem - in the parent task or the child task - and whether you are having a problem putting values into the selection criteria *\*before\** they are passed to the child task, or having values filtered out *\*during\** the processing of the child task. It is no good me investigating what is going on in the processing of the parent task if the problem actually lies in the processing of the child task.

---

---

Subject: Re: Access non key data from parent table  
Posted by [AJM](#) on Thu, 18 Aug 2011 10:26:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

For more information on how to use the `_cm_getPkeyNames()`, `_cm_getWhere()` and `_cm_filterWhere()` methods please take a look at [Passing context between objects](#).

---