Subject: Multiple parent/child relations between 2 tables
Posted by ikatz on Fri, 23 May 2008 20:45:19 GMT
View Forum Message <> Reply to Message

I am working on a project tracking application, which keeps track of both the hierarchy of tasks (tree) and dependencies between tasks (graph).

```
mysql> desc task;
+----------------+- ...
| Field          |
+----------------+- ...
| task_id        |
| project_id     |
| parent_task_id |
| name           |
| estimate_days  |
| description    |
| start_date     |
| end_date       |
+----------------+- ...
```

```
mysql> desc task_order;
+-----------------+- ...
| Field           |
+-----------------+- ...
| blocked_task_id |
| waiting_task_id |
+-----------------+- ...
```

The task hierarchy is taken care of through the task.parent_task_id field, and is working fine. Relating the task table to itself using an alias was no problem.

The dependencies are represented using task_order, where task_order.waiting_task_id represents the item that task_order.blocked_task_id is waiting on. These fields together are the primary key and they both refer to task_id. I feel like this is a reasonable way to represent the graph edges.

I added 2 relations on task in the dictionary:
task (aliased as task_blocked) --> task_order (aliased as task_order_blocked)
task (aliased as task_waiting) --> task_order (aliased as task_order_waiting)
I am not confident that I did this correctly, so please set me straight here if need be.

The problems emerged when I tried to set up a list2 transaction using task as the outer table and task_order as the inner table. When I tried to add a new record (add2), it complained that I had not selected a parent. I then tried using task_order_waiting and task_order_blocked for the inner table, which didn't help.

So, I tried instead to set up a list1 transaction for task_order. This time I was able to get to the add1 screen, which displayed a textbox for both blocked_task_id and waiting_task_id. I modified the task_order.dict.inc file to use task(popup1) for both fields, but this didn't work the way I had hoped. Both popups return a value to the blocked_task_id field, so I am unable to complete the add operation.

I suspect that I am making some combination of simple mistakes, but I didn't find a good example

of this type of design pattern in the documentation -- specifically, what the dict file should look like. If you have a page (or previous forum post) on this, please point me to it. Otherwise, do you see where I am going wrong here? My goal is to set up task_order as a list2 transaction that I can get to from a list1 or list2 of task records.

Subject: Re: Multiple parent/child relations between 2 tables
Posted by AJM on Fri, 23 May 2008 23:19:09 GMT
View Forum Message <> Reply to Message

I see that between the two tables you have various different versions of task_id, namely:
- task_id
- parent_task_id
- blocked_task_id
- waiting_task_id

How are you converting 'task_id' to any of the other values? When passing context between database objects there is no automatic conversion of column names, so you will have to do it manually.

If you look in the menu system at the relationships between the MNU_TASK and MNU_MENU tables you will see that the column name 'task_id' has to be translated to either 'menu_id' or 'task_id_jnr', and this is done by custom code as it cannot be done automatically by the framework.

Subject: Re: Multiple parent/child relations between 2 tables
Posted by ikatz on Thu, 29 May 2008 18:44:21 GMT
View Forum Message <> Reply to Message

Quote:If you look in the menu system at the relationships between the MNU_TASK and MNU_MENU tables you will see that the column name 'task_id' has to be translated to either 'menu_id' or 'task_id_jnr', and this is done by custom code as it cannot be done automatically by the framework.

That is interesting.

What I took from the documentation was that the translation between field names in different tables was done by the dict file (this is a portion of my task_order.dict.inc):

```
    // parent relationship details
    $this->parent_relations[]       = array('parent' => 'task',
                                    'alias' => 'task_blocked',
                                    'parent_field' => 'name as taskblocked_name',
                                    'fields' => array('blocked_task_id' => 'task_id'));

    $this->parent_relations[]       = array('parent' => 'task',
                                    'alias' => 'task_waiting',
```

'parent_field' => 'name as taskwaiting_name',
'fields' => array('waiting_task_id' => 'task_id'));

I am having trouble seeing what the mnu_menu.dict.inc and mnu_task.dict.inc files are doing differently from mine (unless parent_field is causing my problem... is it?). Also, nothing in the mnu_task.class.inc or mnu_menu.class.inc files is jumping out at me as a solution to the problem I am seeing. Where should I be looking for the code that does the translation?

Is there a cut-down version of this in the documentation somewhere? I feel like I am missing something obvious.

Subject: Re: Multiple parent/child relations between 2 tables
Posted by AJM on Thu, 29 May 2008 19:15:05 GMT
View Forum Message <> Reply to Message

When you say that the "translation between field names in different tables was done by the dict file" you are mistaken. This can only be done internally within a table object when it is creating the 'ON (...)' clause of a JOIN within a SELECT statement.

However, the same logic does not apply when a field name is passed into a table object from an external source because there is nothing to identify which relationship should be used, only the field name itself. There is no way to identify which table the input field comes from, which means that there is no way to identify the relationship, which in turn means that there is no way that the framework knows how to convert the input name to a different name so that it can be applied to the current table.

When I said that "this is done by custom code as it cannot be done automatically by the framework" I did not mean custom code within the '*.dict.inc' file as this does not contain customisable code - it is generated from the contents of the data dictionary. I meant custom code within a table class, as in the '_cm_pre_getData()' method of file 'mnu_menu_snr.class.inc'.

The 'mnu_menu' table contains two foreign keys, named 'menu_id' and 'task_id_nr', but the field name which is passed in is named 'task_id'. This has to be changed to one or other of the two different names, but which one? The framework cannot do it automatically, so it has to be done manually.

Subject: Re: Multiple parent/child relations between 2 tables
Posted by ikatz on Thu, 29 May 2008 22:53:17 GMT
View Forum Message <> Reply to Message

Quote:However, the same logic does not apply when a field name is passed into a table object from an external source because there is nothing to identify which relationship should be used, only the field name itself. There is no way to identify which table the input field comes from, which means that there is no way to identify the relationship, which in turn means that there is no way

that the framework knows how to convert the input name to a different name so that it can be applied to the current table.

Thanks! That pretty much sums up the problem I saw. I found the solution to my list1 issue at http://www.tonymarston.net/php-mysql/infrastructure-faq.html #faq82. The problem was that both of blocked_task_id and waiting_task_id were using the same mnu_task popup (referenced in the $fieldspec array) to select their values.

Based on some of the articles you've written, I get the sense that not many people tell you how great your software is. Well, Radicore is f-ing fantastic; I've worked with it for several months now and am still amazed (after resolving issues like this one) at how much code I don't have to write.

I'll try to get the list2 problems resolved through the .class files.

Subject: Re: Multiple parent/child relations between 2 tables
Posted by AJM on Thu, 29 May 2008 23:08:21 GMT
View Forum Message <> Reply to Message

I'm glad your problem is solved, and I'm pleased you like my software (even if you are in a minority).
