Subject: multi2
Posted by interop on Mon, 09 Apr 2007 20:36:02 GMT
View Forum Message <> Reply to Message

I've created a multi2 task and it is failing in updateRecord() at line 3176 because _dml_ReadBeforeUpdate() is passed the updated where clause not the original where clause.

e.g. I have an employee with a number of phones in a table emp_phone. I want to change the number for a phone from 555-1234 to 555-5678 and the _dml_ReadBeforeUpdate() is passed $where='employee_id='10' AND phone_number='555-5678' and returns the error Could not locate original EMP_PHONE record for updating (employee_id='10' AND phone_number='555-5678')

edit: radicore 1.23.0

Subject: Re: multi2
Posted by AJM on Mon, 09 Apr 2007 21:50:32 GMT
View Forum Message <> Reply to Message

When _dml_ReadBeforeUpdate() is called in the updateRecord() method the $where string is constructed using ONLY the primary key of the current record, and the radicore framework does not allow the updating of primary keys in its default behaviour. If you wish to update a primary key then you can only do so with custom code, in which case you should use the updateSelection() method.

There is a function built into the framework which updates primary keys - take a look at the Task Rename function.

Subject: Re: multi2
Posted by interop on Tue, 10 Apr 2007 14:27:44 GMT
View Forum Message <> Reply to Message

Ok, I've added a separate primary key to the child table (emp_phone) and all is working fine. Is this an appropriate solution?

Thanks

Subject: Re: multi2
Posted by AJM on Tue, 10 Apr 2007 14:37:15 GMT
View Forum Message <> Reply to Message

I assume you mean a candidate or unique key as a database table can have only one primary key.

If your design says that a single phone number can only be used by a single employee, then specifying this is a candidate key would work. Because it is separate from the primary key the framework will allow it to be updated at any time, and will automatically check any new value for uniqueness.

Subject: Re: multi2
Posted by interop on Wed, 11 Apr 2007 15:04:19 GMT
View Forum Message <> Reply to Message

In my original design emp_phone did not have a primary key. My design was Employee(one) --> emp_phone(many). Employees could have more than one phone (cell, home, work,...). What I did was add a new column "emp_phone_id" to emp_phone (an auto_increment, primary key) and that seemed to fix my problem.

Subject: Re: multi2
Posted by AJM on Wed, 11 Apr 2007 15:59:25 GMT
View Forum Message <> Reply to Message

Every table in a relational database is supposed to have a primary key. EMP_PHONE should therefore have a primary key of EMPLOYEE_ID + PHONE_ID which will prevent the same PHONE_ID from being added to the same EMPLOYEE_ID more than once. This will also allow an EMPLOYEE_ID to have more than one PHONE_ID, and allow a PHONE_ID to be shared by more than one EMPLOYEE_ID.

When you want to change an employee's phone number using this design what you have to do is DELETE the old one before INSERTING the new one. Some DBMS engines do not allow primary keys to be updated (which is why technical primary keys were invented in the first place), so this is the way to do it. The Radicore framework follows this traditional convention by not allowing any part of a primary key to be updated in its default processing. However, it is possible to update a primary key with custom code as I mentioned in a previous post.