Subject: std.update3.inc confusion
Posted by ljkbrost on Sat, 14 Nov 2009 16:30:47 GMT
View Forum Message <> Reply to Message

Hi,

I'm trying to use the std.update3.inc pattern to create a dynamic report generation screen and I've run into a problem that I have not seen before and I'm not sure of the fix.

To give an idea of what I'm trying to do:

I have a rpt_report and rpt_parameter table. The rpt_report table contains the description of the report and the rpt_parameter table contains the description of the parameters (one parameter per row).

1) Display list1 of the reports
2) Select a report from the list1 screen.
3) Click the run_report (an update3 screen)
4) Inside the _cm_changeConfig fuction:
4.1) Grab a list of all the parameters based on the passed in report_id (populated by the list1->update3 selection process)
4.2) Build the fieldspec based on the data pulled in from this report.

So this is all pretty simple/straight forward, but I have a problem the somehow my where clause in the query to get the parameters has an extra '=' operator.

Query:

```
SELECT *
FROM
  rpt_report
 JOIN
  rpt_version
 ON (rpt_report.report_default_version_id =
    rpt_version.version_id)
 RIGHT JOIN
  rpt_parameter
 USING (version_id)
WHERE
 rpt_report.report_id='='060a-7bb8-36e67021fcb81fe-01224c''
```

Notice the WHERE clause has an extra "=" in the front.

I've traced the problem to the std.table.class.inc's initialise() method. Inside this class it there is the following call:

```
    // convert $where string to an array
```

```
$fieldarray  = where2array($where, false, false);
```

This function ultimately returns a fieldarray with

```
Array
(
   [report_id] => ='060a-7bb8-36e67021fcb81fe-01224c'
)
```

Stepping through the where2array function everything works upto the point when the function strips the operators. In this case the function is passed $strip_operators=false, so it leaves the extra '=' in the fieldarray.

Ultimately when I'm in _cm_changeConfig, my $fieldarray is not what I would expect. Changing the initialise() method to strip the operators fixes the problem.

So, should the where2array be change to strip_operators=true?
Or, am I missing something?
