

---

Subject: RADICORE V2.12.0 released

Posted by [AJM](#) on Wed, 01 May 2019 15:55:20 GMT

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

---

This version adds a new option to the processing of the DETAIL view in PDF documents, the ability to add a blank line when creating CSV documents, plus a few small enhancements and bug fixes.

This version contains the following changes:

Database changes:

- none

Other changes:

fixed a bug in 'dict\_database\_s01.class.inc' in the Data Dictionary which caused a problem when importing from a PostgreSQL database. See <https://forum.radicore.org/index.php?t=msg&th=3416> for details.

fixed a bug in 'dict\_related\_column.class.inc' in the Data Dictionary which failed to show any entries in the \$inner area in the update screen if none were initially created in the insert screen.

updated processing for the DETAIL view in PDF documents so that when using the additional data option, which uses code in the `_cm_output_multi()` method, it will allow multiple iterations through the 'multi1' to 'multi99' entries using different WHERE strings. Multiple iterations can be requested by putting code in the `_cm_output_iterations()` method to provide an array of different WHERE strings. By default an empty array will be output which will signify a single iteration with an empty WHERE string.

updated 'std.table.class.inc' to include the new `_cm_output_iterations()` method.

updated 'std.table.class.inc' to add a third argument called \$iteration to the `_cm_output_multi()` method. This will contain the WHERE string for each iteration. If nothing is set in the `_cm_output_iterations()` method then by default `_cm_output_multi()` will only cycle once through the 'multi1' to 'multi99' entries with a NULL value for the \$iteration argument.

updated 'std.table.class.inc' so that the `array_update_indexed()` method calls the `$this->array_update_associative()` method for each row in the array instead of the `array_update_indexed()` function for the whole array.

updated 'std.table.class.inc' so that the `customButton()` method calls the `$this->array_update_associative()` method instead of the `array_update_associative()` function.

updated 'std.table.class.inc' so that the `updateLinkData()` method will operate correctly when more than one row of the outer entity has been selected.

updated 'std.table.class.inc' so that the `updateMultiple()` method calls the `$this->array_update_associative()` method instead of the `array_update_associative()` function.

updated 'std.table.class.inc' to fix a bug in the `updateRecord()` method which caused it, in some circumstances, not to read the record that was due to be updated, thus causing an error.

updated 'std.table.class.inc' to include the `_cm_pre_pasteData()` and `_cm_post_pasteData()` methods which will be called by the `pasteData()` method when the PASTE button is pressed.

updated 'std.table.class.inc' so that the `getData()` and `getData_serial()` methods will examine the SQL\_FROM string to see if the current table name has been given an alias. If so then the alias name will be passed to the `$DML->getData()` method instead of the original name, thus preventing an error when the SQL query is constructed.

updated 'std.table.class.inc' so that the `_sqlAssembleWhere()` method will not filter out a

fieldname from the WHERE string if it is used as an alias for a field which exists in a table which is mentioned in the FROM string.

updated 'std.table.class.inc' so that when the `_sqlAssembleWhereLink()` method constructs its query it will start with whatever is specified in the component script.

updated 'include.library.inc' to correct some issues encountered in the `array2where()` function.

updated 'include.library.inc' to correct an issue encountered in the `convert_parent_id_to_child_id()` function with the stripping of operators from the contents of the `$fieldarray` argument.

updated 'include.library.inc' to correct an issue encountered with the `getLanguageArray()` method when a `require_once` fails due to it already having been processed.

updated 'include.session.inc' to correct an issue in the `initSession()` function which caused the CSRF check to fail when it should not have.

updated 'include.session.inc' to correct an issue in the `scriptNext()` function which failed to check that an entry extracted from the `$_SESSION['script_sequence']` array was actually valid.

updated 'std.validation.class.inc' so that the `validateField()` method uses a static variable for the `$date` object so that it does not need to instantiate it more than once in any processing cycle.

updated the various 'dml.???.class.inc' files so that when filling in any autoinsert or autoupdate fields with the current timestamp they use the server's time instead of the client's time.

updated 'std.csv.class.inc' so that if the contents of the variable `$dbobject->insert_blank_line` is set to a non-zero value following a call to `$dbobject->formatData()` then a blank line will be inserted into the CSV file.

updated 'std.list2.inc' to insert a call to `$dbinner->restart()` after the call to `$dbinner->initialise()`.

updated 'std.search1.inc' to correctly deal with boolean values which may have values containing either true/false, 0/1, T/F or Y/N.

updated 'std.multi4.inc' to deal correctly with the situation where `$dbouter->skip_getData` is set to TRUE.

updated 'std.add1.inc', 'std.add2.inc' and 'std.update1.inc' so that they use the `$dbobject->pasteData()` method instead of the `pasteData()` function. This allows the use of the `_cm_pre_pasteData()` and `_cm_post_pasteData()` methods.

updated 'std.update1.inc' to allow an empty `$where` string if `$dbobject->allow_empty_where` is set to TRUE.

updated 'std.add7.inc' to correct an issue which did not place an error following a popup call on the correct line in the inner zone.

updated 'std.link1.inc' to remove the call to `$dblink->array_update_indexed()` as the necessary processing is performed within `$dblink->updateLinkData()`.