

---

Subject: RADICORE V2.17.0 released  
Posted by [AJM](#) on Mon, 09 Nov 2020 15:06:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

This version contains some enhancements, including the way in which tables are locked during a database transaction, and a few bug fixes.

Database changes:

- none

Other changes:

AUDIT subsystem: updated the search screen to allow filtering by tran\_seq\_no.

DICT subsystem: fixed bug in dict\_column\_s01.class.inc where the \_cm\_getInitialDataMultiple() method returned \$insert\_array instead of \$rows.

MENU subsystem: fixed bug in mnu\_user.class.inc which had an extraneous ')' in an SQL query.

MENU subsystem: fixed bug in mnu\_subsystem\_s01.class.class.inc so that it does not execute the \_cm\_commonValidation() and \_cm\_validateUpdate() methods in the parent class.

MENU subsystem: fixed bug in mnu\_task\_s02.class.class.inc so that it does not execute the the \_cm\_commonValidation(), \_cm\_post\_updateRecord() and \_cm\_validateUpdate() methods in the parent class.

WORKFLOW subsystem: fixed bug in wf\_workitem.class.inc where it referred to 'mnu\_task.task\_desc' instead of 'mnu\_task.task\_name'.

MYSQL driver: fixed bug in buildKeyString() so that if any component in a primary key is null then the entire key is treated as null.

MYSQL driver: fixed bug in getCount() so that it will allow any number of whitespace characters between the leading '(' and the word 'SELECT'.

MYSQL driver: fixed bug in getData() so that it builds the correct query when there is a WHERE and/or HAVING string.

MYSQL driver: fixed bug in updateRecord() where it reported the wrong WHERE string following an error with a candidate key.

MYSQL driver: fixed bug in updateRecord() where it failed to abort if there was an error with the UPDATE query.

MYSQL driver: for MySQL version 8.0.17 and above it will allow the use of column constraints instead of the deprecated 'unsigned' attribute.

std.ADD2.inc: avoid generating message "N records were inserted into XXXX" is \$this->no\_display\_count is set to TRUE;

std.ADD2.inc: include call to \$dbobject->setFinalAction() at end of processing of GET request.

std.ADD2.inc: include call to \$dbobject->runImmediateTask() prior to generation of HTML output.

std.ADD7.inc: add code to regenerate \$inner\_where when processing the 'SubmitStay' button.

std.ADD7.inc: add call to \$dbouter->getFieldArray() after processing popup for \$dbinner in case its data has changed.

std.MULTI4.inc: fix bug to deal correctly with setting of \$dbobject->use\_scrollarray.

std.MULTI4.inc: keep \$inner\_where separate from \$outer\_where.

std.MULTI4.inc: include calls to \$dbouter->getFieldArray() and \$dbinner->getFieldArray() after processing a popup form in case any data has been changed.

std.OUTPUT2.inc: add call to ob\_flush() at end of script.

std.OUTPUT3.inc: add call to ob\_flush() before returning to previous screen.

std.OUTPUT5.inc: add call to ob\_flush() at end of script.

std.SEARCH1.inc: if a field is empty but has both \$fieldspec['optionlist'] and \$fieldspec['initial\_value'] set then load that initial value.

std.CSV.class.inc: fix bug which caused \$dbobject->formatData to be called twice on the first row.

std.DATEVALIDATION.class.inc: remove redundant 'return false;' lines after throwing an error.

std.PDF.class.inc: fixed bug which prevented a barcode or image from appearing in a line without other cells.

std.SINGLETON.php5.inc: added code to throw an error if the requested file does not exist.

std.VALIDATION.class.inc: added code to deal with a decimal number with precision = 38 and scale = 0.

include.XML.php5.inc: amended addData2XMLdoc() so that if \$dbobject does not contain a value for \$this->zone then the default value of 'main' is used instead.

include.XML.php5.inc: amended addData2XMLdoc() so that when \$fieldspec is refreshed then so is the screen structure in \$this->structure. This will allow the screen structure to be amended in the \_cm\_formatData() method.

include.XML.php5.inc: amended addData2XMLdoc() so that when a button for another task is created it is in the format 'task#{ \$zone }#{ \$task\_id }'. If it is in a zone which contains multiple rows then '[offset={rownumber}]' will be appended.

include.XML.php5.inc: amended addData2XMLdoc() so that when loading the contents for a checkbox with an optionlist which contains '\_no\_undef' then it is displayed as a checkbox instead of a dropdown list with an additional 'undefined' option.

include.XML.php5.inc: amended buildXML() to use \$this->javascript instead of \$javascript so as not to lose any entries.

include.LIBRARY.inc: amended append2ScriptSequence() to force \$next to be the first in the queue if \$next['immediate'] is set.

include.LIBRARY.inc: amended array2string() to add an option 2nd argument called \$separator with a default value of '&'.

include.LIBRARY.inc: amended array2where() so that \$dbobject->getFieldSpec() is not called if that method does not exist.

include.LIBRARY.inc: amended check\_task\_button() to include "[offset={ \$offset }]" in the output value if \$osset is not zero.

include.LIBRARY.inc: amended extractSelectList() to deal with more combinations in the input string.

include.LIBRARY.inc: amended several functions to replace code with a call to splitWhereByRow().

include.LIBRARY.inc: amended findClosingParenthesis() to simplify the code using a better regular expression.

include.LIBRARY.inc: amended qualifyField() to deal with an entry in the SELECT list which is in the format 'func(field)'.

include.LIBRARY.inc: amended removeDuplicateFromSelect() to deal with a select string containing 'tablename.\*'.

include.LIBRARY.inc: amended selection2PKeyOnly() to deal correctly with an entry which is not in \$fieldlist.

include.LIBRARY.inc: amended selection2where() to verify that the selected rows have an entry in \$pkeyarray.

include.LIBRARY.inc: amended where2indexedArray() to fix bugs in some regular expressions.

include.SESSION.inc: amended childForm() to extract 'offset' if it is available in the input string. This deals with a button which is pressed in a zone containing multiple rows.

include.SESSION.inc: amended childForm() so that calls to \$dbobject->getWhere() include the \$offset value.

include.SESSION.inc: amended childForm() so that calls to \$dbobject->getSearch() include \$task\_array.

include.SESSION.inc: amended scriptNext() so that when processing an entry from \$\_SESSION['script\_sequence'] which contains the option 'no\_messages' the contents of \$this->messages are not forwarded to the next script.

std.TABLE.class.inc: added \$save\_initial\_where variable so that if a WHERE string for an INNER entity is validated and modified in the \_cm\_initialise() method then this value will override the one constructed from its parent entity.

std.TABLE.class.inc: amended commit() so that if \$this->fieldarray is empty but \$this->parent\_data is not then it will call \_examineWorkflow() on the parent object.

std.TABLE.class.inc: added convertSearchExpression() and \_cm\_convertSearchExpression() will be called in the initialise() method in order to allow the contents of \$search to be modified.

std.TABLE.class.inc: amended \_cm\_customButton() to include a 3rd argument to identify the row number for use in a zone which contains multiple rows.

std.TABLE.class.inc: amended various method calls so that if they return FALSE it will be converted into an empty array.

std.TABLE.class.inc: amended getData() method to replace contents of \$where argument with \$this->save\_initial\_where if it is not empty.

std.TABLE.class.inc: amended getExtraData() method to remove the check of \$this->no\_foreign\_data before calling getForeignData().

std.TABLE.class.inc: amended getForeignData() method so that after calling \_cm\_getForeigndata() it will skip further processing if either \$this->no\_foreign\_data or \$fieldarray['rdc\_no\_foreign\_data'] is set.

std.TABLE.class.inc: added \_cm\_adjustColumnNames() method to allow list of field names in the primary key to be adjusted.

std.TABLE.class.inc: amended getPkeyArray() to call \_cm\_adjustColumnNames() after \_cm\_getPkeyNames().

std.TABLE.class.inc: added getSearch() and \_cm\_getSearch() methods to allow the \$this->sql\_search string to be adjusted before it is passed to the next script.

std.TABLE.class.inc: amended getWhere() to include a 2nd argument called \$offset so that when \$this->fieldarray contains multiple rows the \$where string is constructed from the current/selected row instead of the first row.

std.TABLE.class.inc: amended initialise() method so that when called in a search screen it will pass any search criteria through the convertSearchExpression() method before the screen is displayed.

std.TABLE.class.inc: amended insertRecord() method so that \_cm\_post\_insertRecord() method is called even if \$this->errors is not empty.

std.TABLE.class.inc: amended popupCall() so that if contents of \$popupname argument is changed it will update the relevant \$this->fieldspec entry to record the name change.

std.TABLE.class.inc: amended popupReturn() so that when searching through \$this->parent\_relations to see if there are any differences between the column names used in the parent table and those used in this child table. By default it will stop at the first entry which contains the returned column names, but if there are multiple entries the correct table name can be identified by setting \$fieldarray['rdc\_table\_id'] = 'whatever' in the \_cm\_popupReturn() method.

std.TABLE.class.inc: added runImmediateTask() method.

std.TABLE.class.inc: added setFinalAction() and \_cm\_setFinalAction() methods.

std.TABLE.class.inc: amended updateRecord() method so that \_cm\_post\_updateRecord() is called even if \$this->errors is not empty.

---