Subject: Bug with 'require_once' file path
Posted by kong on Thu, 10 Mar 2016 02:34:02 GMT

Replication:
1. Using Radicore v1.93, create 2 subsystems
 Subsystem A has 1 table Subsystem B has 2 tables that are related to each other with parent/child relationship
2. In subsystem A, create a component script in which subsystem B's child object is invoked using the singleton function and call getData() function using that object.
3. When this component script is executed the web application will hang with a white browser screen.

Problem trace:
The problem has been traced back to the file std.table.class.inc, function _sqlProcessJoin(), specifically the following lines: if (array_key_exists('subsys_dir', $reldata)) {

```
    ...
} else {
    $dir = NULL;
} // if
if (!class_exists($parent_table)) {
    require_once $dir ."classes/$parent_table.class.inc";
} // if
```
The web app hangs at the execution of the require_once statement. It appears that all this code has been called as part of subsystem B child object's process to obtain its parent record information. Since both child and parent objects are in the same subsystem, the variable $dir will be NULL, which makes the 'require_once' file path invalid because the component file was executed under the directory of subsystem A.

Solution:
In the code shown above, change this line: $dir = NULL;into this: $dir = dirname($this->dirname) .DIRECTORY_SEPARATOR;