



CODING STANDARDS

General

1. All variables should be properly scoped and variable calls should reference said scope (e.g. arguments.firstname or request.user_id)
2. CFCs: All code should be contained within functions
3. Functions: Pass all variables into your functions—don't rely on global variables being available when a function needs it.
4. Code should be efficient but not overcomplicated
5. Use functions to break up processes into practical methods
6. Keep readability in mind when coding and commenting—again, avoid overly complex code, instead, utilize your coding expertise to make code processes obvious and readable
7. No queries or loops on client side pages/templates (e.g. the output). Do your processing within the CFC and create a clean "method" that can be invoked.

Example:

```
<cfscript>
    objTask = CreateObject('component', 'taskScript');
    myTasks = objTask.GetTasks();
</cfscript>
```

8. Use tag notation when creating functions. Script notation within a function is fine when obviously the better way to go. For instance when setting variables because there's so much less to type etc.

Example:

```
<cffunction name="AddTask" access="public" output="true" returntype="boolean" hint="I'm a task function" >
    <cfargument name="title" type="string" required="true" default="">
    <cfscript>
        try {
            var tasks = EntityLoad('ormtasks');
            var newTaskObj = EntityNew("ormtasks");
            newTaskObj.settitle(arguments.title);
            EntitySave(newTaskObj,true);
            ormflush();
        }
        catch (Exception ex) {
            WriteOutput("<p>#ex.message#</p>");
        }
    </cfscript>
    <cfreturn true>
</cffunction>
```

Prepared by Steven Edward Smith, CTO

If you decide to copy this document for your own use, please mention us as your source.

Thanks!