

# Rx Script Guidelines

## Where to add new functionality?

When adding new functions, should they be in the OcapAppDriver.java file, or in a Level1, Level2 or Level3 script? What is in the different levels of bean shell scripts? When should you write a new Level1, or Level2, or Level 3 script?

Level2 scripts are the traditional tests. They can pass or fail. Should a Level2 test not behave as expected perhaps there is a bug in the stack (or a bug in the script).

The test script levels map to CTP tests as follows:

- Level 1 = CTP Test Utilities - an aggregation (more than 1) of OAD calls which perform a specific task, provides common functionality which is reused by multiple scripts
- Level 2 = CTP Test
- Level 3 = CTP Test Scenario

Can a Level 2 script call OAD functions directly?

1. Level 1 scripts should be created to encapsulate a common set of calls to multiple OAD functions to reduce duplication of code
2. There should not be a Level 1 script which is just a wrapper to an OAD function
3. Level 2 scripts can call a single OAD function directly but calling multiple OAD methods from a Level 2 script implies that those calls should be moved into to a Level 1 script.

A Level1 script should always behave as expected. If not, the cause is likely the environment created by the sequence of other Level1 scripts. For example, if a Level2 script attempts to playback a recording (Level1) before publishing (Level1), the problem is not with the Level1 scripts. It is in the Level2 script. The Level2 script calls the Level1 scripts in the wrong order.

## Metaphors

How are the levels like the components in the HN integration tests in the spreadsheets?

Level 1 – like a single row in the spreadsheet. It is like a menu leaf in DvrE.

Level 2 – A sequence of level1 scripts that meet a test objective.

Level 3 – a suite of tests referenced by continuous integration.

## There are boundaries between the levels and OAD.

Only OAD can call OCAP methods.

Level1 bean shell scripts can call OAD methods

Level2 can call level 1 and single OAD methods directly. If a Level2 script calls a series of OAD functions, this implies a level1 script is needed.

Level 3 only call level 2

## Other conventions

No header comments in level 1. They should be self evident.

Grammar is verb followed by object of the verb or a noun.

No void returns on any level 1 scripts.

Whenever possible, static constants defined in the OcapAppDriverInterface classes should be used. These constants may be accessed as instance variables rather than using a static declaration. For example, it is acceptable to use rxDriverCore.TUNED instead of OcapAppDriverCore.TUNED.

## BEAN SHELL SCRIPT CODING CONVENTIONS:

The following are the conventions to follow when writing a script:

1. Level 3 scripts can only call Level 2 scripts (example would be the smoke test bean shell script), no OCAP API calls or OcapAppDriver method calls.
2. Level 2 scripts can only call Level 1 scripts (example would be a sequence of menu option selections using DvrExerciser to perform an integration test). Level 2 scripts can not call OCAP App Driver functions or OCAP APIs.
3. Level 1 scripts can only call OcapAppDriver functions (example would be "make a recording" bean shell script). Level 1 scripts can not call other scripts or OCAP APIs.
4. If a script needs to call an OCAP API which is not currently supported via OcapAppDriver API, a OcapAppDriver API should be added, see "[How to Add A New Function" Design Documentation](#)[/../../../../../../../../pages/viewpage.action?pageId=39846061]"
5. Scripts should be checked into svn: trunk\ri\RI\_Stack\apps\qa\org\cablelabs\lib\rxscripts into the appropriate subdirectory - level1, level2 or level3
6. Level 3 script is responsible for setting the rxReturn and rxReturnString to report the overall status of the level 3 test script
7. Level 1 and Level 2 scripts should return a String. If the string is empty, this implies the script ran without problems and should be considered "PASS". If the string is not empty, this implies the script ran into problems and should be considered "FAIL" with details provide in the returned string.
8. In Level 2 scripts, there should be a single "[name of the script]" method which will be used to call the script from Level 3 or other Level 2 scripts. If more than one is needed, the script should be divided into more than one script.
9. rxLog() statements do not need to include a logging guard since the RiScriptlet does this on behalf of the script.
10. rxLog() statements will start with the name of the bean shell script to help clarify log messages. The RiScriptlet appends the name of the script to the log message automatically.
11. If a script has methods within the file, the methods should have a prefix which is an abbreviation of the script file name, i.e. LiveStreamingTrick10 has a method call lst\_subtest10()
12. Although the scripts are not technically java, they should adhere to the RI coding standards, specifically to java standards where applicable.