Testing Framework Status

René Météry

Outline

- Purpose of the framework
- Status of tests integration
  - Testing Framework status
  - Tests status
- Tests definition
- New tests definition
  - What's new
  - 3 files for defining a test
- The 3 files
- How to define a test
- Conclusion

Purpose of the framework

- Define Development Conventions
  - reusable core functions
  - required options for high level scripts
  - HTML reports
  - Exit codes
  - predefined environment variables and configuration files
- Allow the integration of existing test scripts from INFN, IFAE...
- Keep the possibility to launch test scripts within or outside the framework
- Allow remote testing
- Organize tests : define level testing and tests steps
  - Need to modify INFN scripts to fit this purpose

Status of tests integration

- Testing Framework status
  - New -L and -t options which allow to run tests without passing by the menus
  - Added personal configuration files in the ~/.edg-tsft directory
  - Correcting few bugs
  - Modifying man page for the dg-tests-menu command
- Tests status
  - No more tests provided
  - Exercise remote testing
  - Adding new way of defining tests
  - Beginning the writing of a HOWTO file describing how to write a test

Tests definition

2 Kind of tests:

- Written with 'case': (IFAE test scripts)
  - +
    - **Organize by steps**
    - **Compatible with HTML reports**
  - -
    - **Hard task to add new tests**
    - **The tests information and command-line are in the same file**
    - **The test script is HUGE.**
- Not written with 'case': (original INFN test scripts)
  - +
    - **Easy to add new tests**
  - -
    - **Not organize by steps**
    - **Not compatible with HTML reports**

⇒ Need to find a new definition with both advantages

New tests definition

- What's new
  - a XML way of test definition
  - a new TEST_func.sh file containing parse functions
  - compatible with existing HTML parsers
  - easier to add or remove tests
  - organise by steps

- 3 files for defining a test
  - .conf file : configuration file for the HTML parser (already exists)
  - .sh file : test script launching the different steps
  - .def file : file describing the steps of the test (information and command)

The 3 files

- The .conf file
  - Contains variables for HTML parser
  - No new things added
- The .sh file
  - Read the .def file
  - The generic test script which launches the different steps
- The .def file
  - Containing all the steps describing with the below tags:
    - **<step>...<steps> : defining a step**
    - **<brf_info>...</brf_info> : containing brief information about a step (shows in the row)**
    - **<det_info>...</det_info> : containing detail information about a step (shows at the end)**
    - **<command>...</command> : containing the command to run**
    - **<title>...</title> : containing the title of new type of test (show in the row and at the end)(optional)**

How to define a test 1/3

Example : CE Installation and configuration tests

- 0D_CE_config_test.conf - Test Configuration file

```
#HTML output config
HTML_OUT=index.html
HTML_STYLE=$EDG_TEST_HTML_STYLE
HTML_DIR=$EDG_TEST_HTML_OUTPUT/$(date'+%y%m%d')_CE_config_test
#generic config
TEST_FILE=$EDG_TESTROOT/bin/installconfig/CE/CE_config_test.sh
TEST_ARGS=""
TEST_TITLE="Computing Element Install and Config Tests"
TEST_ROWS_LIST=`hostname`
```

How to define a test 2/3

- CE_config_test.sh
  - Generic script (the core is the same for all the test scripts)
  - Lines needed to be modified

```
export DEF_FILE=$EDG_TESTROOT/bin/installconfig/CE/CE_config_test.def
export TMP_LOG_STEP_FILE=$EDG_TEST_TMP/CE_cfg_step$$.tmp
export TMP_LOG_INFO_FILE=$EDG_TEST_TMP/CE_cfg_info$$.tmp
export TMP_LOG_CMD_FILE=$EDG_TEST_TMP/CE_cfg_cmd$$.tmp
```

How to define a test 3/3

- CE_config_test.def - Test definition file

```
// Step definition

<step>
      <brf_info>
      getting GLOBUS_LOCATION from $EDG_TEST_GLCF and check directory
      </brf_info>
      <command>
      checkglobusconfparam GLOBUS_LOCATION
      </command>
<title>
Mandatory Parameters in $EDG_TEST_GLCF
#####################################
</title>
<det_info>
Get the GLOBUS_LOCATION variable from $EDG_TEST_GLCF and check directory
</det_info>
</step>
```

Conclusion

- Rewrite Installation & Configuration tests
  - Need to write the .def file for all the tests
- Rewrite Functionality tests
  - Need to write the .sh file for 1D and 2D tests
  - Need to write the .def file
- Remote tests integration : work in progress
- Integrate new tests