Test Definitions

Contents

The test cases, both manual and automated, are written in the LAVA test definition file format, which stores the instructions to run the automated tests in YAML files.

Git is used as the data storage backend for all the test cases. The current Apertis tests can be found in the Apertis Test Cases[1] repository. The test cases are versioned using Git branches to enable functionality change without breaking tests for older releases.

The format has been extended to add all the required test case data to the tests definition files. A description of these changes can be found in the README.md[2] This approach avoids the issue of test case instructions differing from the executed steps in automated tests, since the test case and the definition file are the same document. The `atc` utility is provided with the test cases to render them to HTML definitions. Test cases labeled as automated are run in LAVA, those labeled as manual should be run by hand using the steps that are generated in the HTML definition.

Mandatory Test Definition Data Fields

A test case file should at least contain the following data fields for both the automated and manual tests:

```
1   format: This is used to identify the format version.
2   name: Name of test case.
3   type: This could be used to define a series of test case types (functional, sanity,
4        system, unit-test).
5   exec-type: Manual or automated test case.
6   image-type: This is the image type (target, minimal, ostree, development, SDK)
7        and the architectures that it's supported on.  description: Brief
8             description of the test case.
9   priority: low, medium, high, critical.
10  run: Steps to execute the test.
11  expected: The expected result after running the test.
```

The test case file format is very extensible and new fields can be added if necessary.

---

[1] https://gitlab.apertis.org/tests/apertis-test-cases

[2] https://gitlab.apertis.org/tests/apertis-test-cases/-/blob/apertis/v2021/README.md