



Magento Functional Testing Framework

Improvements
& Future Direction



Tom Erskine

Quality Lead, Community Engineering

Since MFTF 1.0.0

- We're all a little older.
- Some of us are a little wiser.
- We've grown from 1.0.0 to 2.3.14.
- Magento Core quality focusses on MFTF testing.
- Bugs are investigated and resolved with the help of MFTF reporting.
- Community Engineering projects are embracing MFTF as a critical part of their quality process.



2.0.0 & 2.0.2

2.0.0

Cest name removed. Tests can now be called tests

2.0.2

New CLI action - `<magentoCli>`. This requires some additional setup (`.htaccess`). Enables you to set and use `bin/magento` commands in your tests. This is very useful for setting the configuration needed for your tests.

2.2.0

Tests now read from app/code directories. This allows tests to live directly with modules and aids modularity and extension testing support.

- **Improved Javascript support.**
 - Javascript errors are shown in the test output.
 - `<executeJs>` action now returns the variable (if any) returned by the script executed
- **Suites:**
 - Allows for grouping of test pre and post conditions for a set of tests
 - Inclusion and exclusion by name , module, and group.
- **Reporting:**
 - Allure reports now collect and correctly display Skipped tests.

2.3.0

Logging:

- mftf.log file now outputted for debugging, particularly in a CI system.
- `--debug` flag will output all debug information in test generation and execution
- Skipping - `<skip>` annotation replaces `<group value="skip">`
- Annotations required as best practice. Deprecation warning shown if not used.
- Plan to enforce annotations in 3.0.0

Running without Magento instance:

- `--force` flag ignores MFTF checking that Magento instance is up

<extends> attribute:

- Allows extending a Test, ActionGroup, Page, Section or Data.
- This creates a copy, allowing both extension and original file to run
- Extension can be modified as if it were a merge but remains independent.

2.3.0 continued

.credentials file:

- MFTF supports a .credentials file, an example is generated by *build:project*
- This allows *key=value* pairs that can be referenced in test files
- The values are not used in output or reporting, so that secrets can stay secret
- Only supported in *<fillField>* action for now.

2.3.0 was kind of a big deal

***Bin/mftf* command:**

- MFTF no longer requires Robo to run commands
- We now use a `bin/mftf` command to run:
 - Generation
 - Test runs
- The syntax is like standard `bin/magento` command
- This is more intuitive
- We've removed our dependency on Robo
- We can edit, fix, and improve the commands available more easily and quickly
- **This change began from a community contribution**

2.3.6

Suites:

- Tests can now reference data created in a suite pre/post condition which aids consolidating all conditions in the suite

--remove:

- This flag will clean out `_generated` directory when used

2.3.10

Run failed:

- `mftf run:failed` command reruns previous failures only

The Future of MFTF and Quality in Magento



MFTF Priorities

1. Support

2. Community contribution

3. Deprecation and incremental improvement

4. Marketplace and extension testing



Support

MFTF is being used internally and successfully.

The MFTF team's first priority now is to make it simple, useful and valuable to our community.

We want you to use this, build with it, and contribute your ideas and fixes.

Our best measurement of success is not bugs found by us, but productivity and quality in our community.

Community contribution

Involvement and contribution from our community of developers and testers will be a driving force in the future of MFTF and quality in Magento.

We've made a framework that we're proud of and we know can help you.

Over the course of 2019, we plan to hold trainings and webinars.

We're encouraging pull requests to our framework and will have up-for-grabs issues and the same community contribution recognition as other projects.

Deprecation and incremental improvement

Version 3 will allow us to follow semantic versioning and release some breaking changes.

The initial release will focus on deprecating outdated features that have been replaced.

In this way, the framework becomes clearer and easier to use.

Marketplace and Extensions

MFTF has been an asset to the quality of Core Magento.

It's being relied on for community engineering projects extensions, like Multi-Source Inventory.

Over the course of 2019, we will be releasing best practices and information on how to use MFTF and develop tests for your extensions.

Marketplace will be integrating MFTF tests that you submit into the Marketplace workflow and frontend so that your commitment to quality can be demonstrated to potential customers.

Functional Tests Migration

MTF -> MFTF Conversion project

Led by : Oleksii Kolesnyk

Our regression coverage of Magento currently is split.

There's growing coverage in MFTF – and all the newest features have been tested first-and-foremost with MFTF tests.

We still have a great deal of coverage in our previous framework – MTF.

There is an ongoing project internally to convert this MTF test coverage to MFTF.

MTF -> MFTF Conversion Community

This is an excellent opportunity to get involved.

You can help submit conversion of MTF tests:

- This will give you a great learning point for MFTF
- It's a simple way to become a contributor
- You can focus on MFTF testing that will have the greatest impact on your org
- Once we accept a converted test, it will be part of our Core regression coverage and will be maintained by us in future versions