

Concepts and ideas for improving the UI test infrastructure

- [Test fixture data](#)
- [Use more REST](#)
- [Quick-fix](#)
- [Remove delay in favor of waitUntil](#)
- [Integrate Jetty9 branch for CE/EE](#)
- [Use page objects](#)
- [Next Steps](#)
 - [Short-term](#)
 - [Intermediate steps](#)
 - [Long term goal](#)

Test fixture data

Change/add MVH default tasks to load fixture data. Could be a specific task that bootstraps e.g. when `magnolia.properties magnolia.load.testdata=true` (similar to `mgnl-bootstrap-samples`)

General question: How to load fixture data?

Idea: Only test the C of CRUD once! Make tests faster!

Use more REST

Use REST (and custom endpoints) once jetty9 is merged in master

Get rid of servlets...?

Quick-fix

Shaving off minutes (17minutes) when checking for `NonExistingElements` and whether they are gone.

- `implicitWait` of the driver
- `waitUntil`
 - each have ten seconds

Remove delay in favor of waitUntil

We still use delay for publication related stuff. We should aim for less delay and more `waitUntils`.

Integrate Jetty9 branch for CE/EE

[MAGNOLIA-5536](#) - Getting issue details... STATUS

Use page objects

- Jan: use css instead of xpath (faster more reliable)
- See Michael's branch (tba)

Next Steps

Short-term

- Integrate jetty9
 - [MAGNOLIA-5536](#) - Getting issue details... STATUS
- Integrate `implicitWait` reduction for `NonExistingElements`

- Start with first UI testframework + first page object implementation

Intermediate steps

- UI test framework that contains Page Objects (abstract)
- Continously "migrate" all UI tests to using page objects

Long term goal

- Modules provide their own UI test and test fixture data
- Remote test execution on Amazon?
 - There are plugins for jenkins...
- Parellelization of UI tests