



Testing and DITA-OT

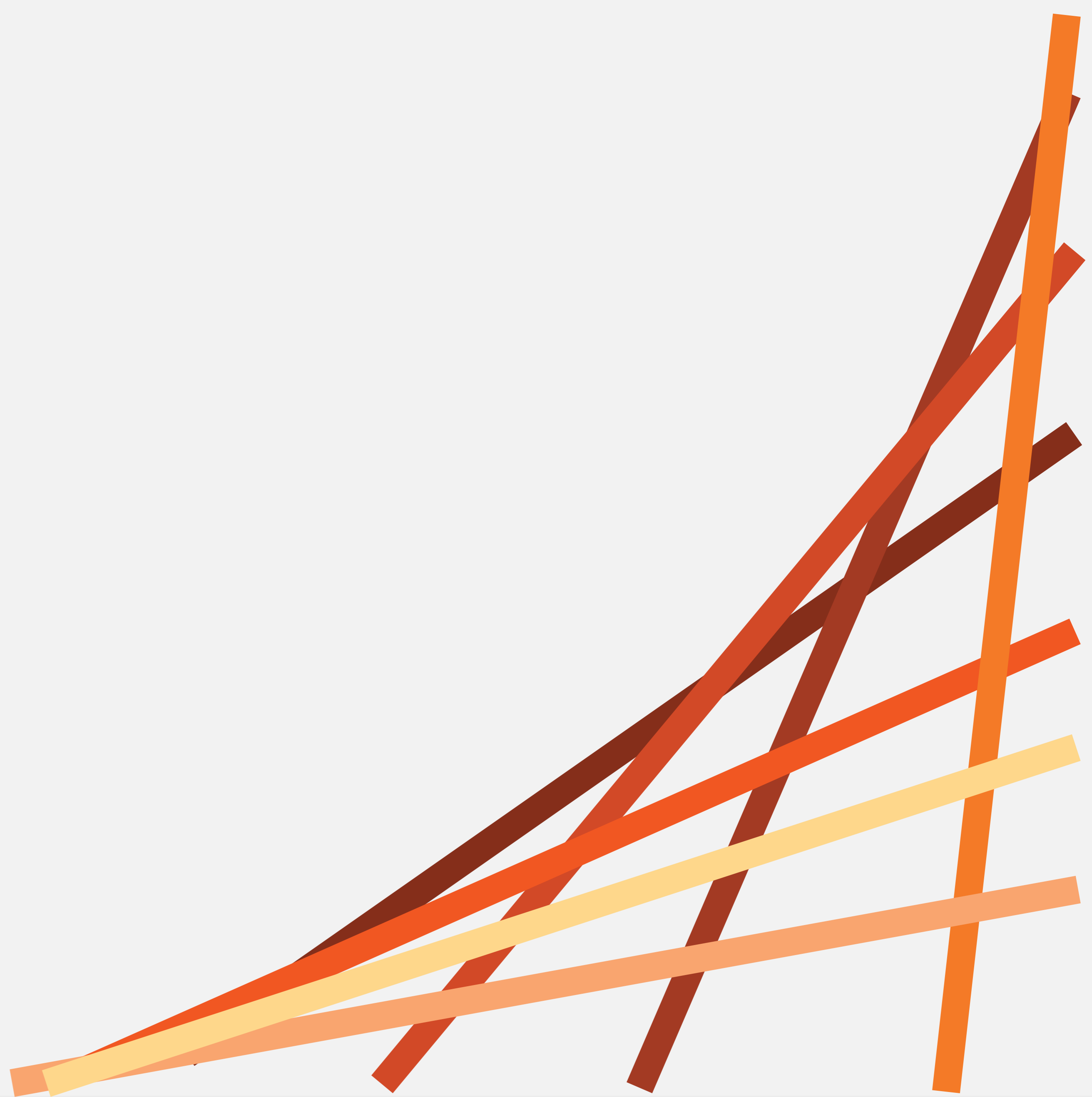
~~Should we maybe do that?~~ How do we do that?

@jelovirt (Jarno)

@robander (Robert)

Agenda

1. We talk about DITA-OT testing
2. You tell us how to do it better



Goals...

1. Fully automated
2. Widespread coverage
3. Easy to use and improve
4. ...



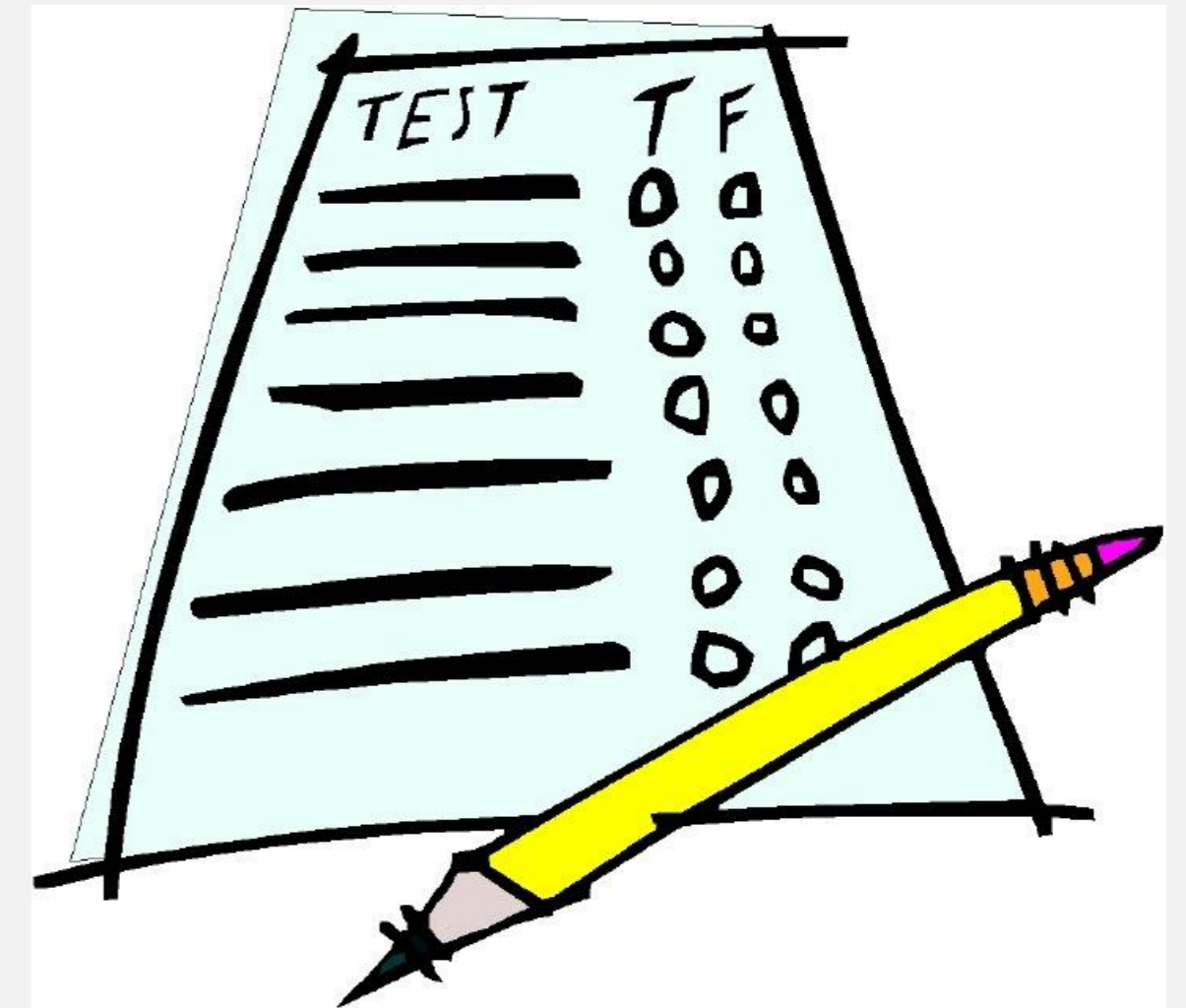
In ancient times...

- Dedicated test team in IBM. Yay!
- Did everything manually. Boo.
 - Integration test: run 100s of builds, compare output to previous release
 - Slow process, difficult to test, easy to miss changes
 - Only done at end of release cycle
- So many integration test files ... More on that later.



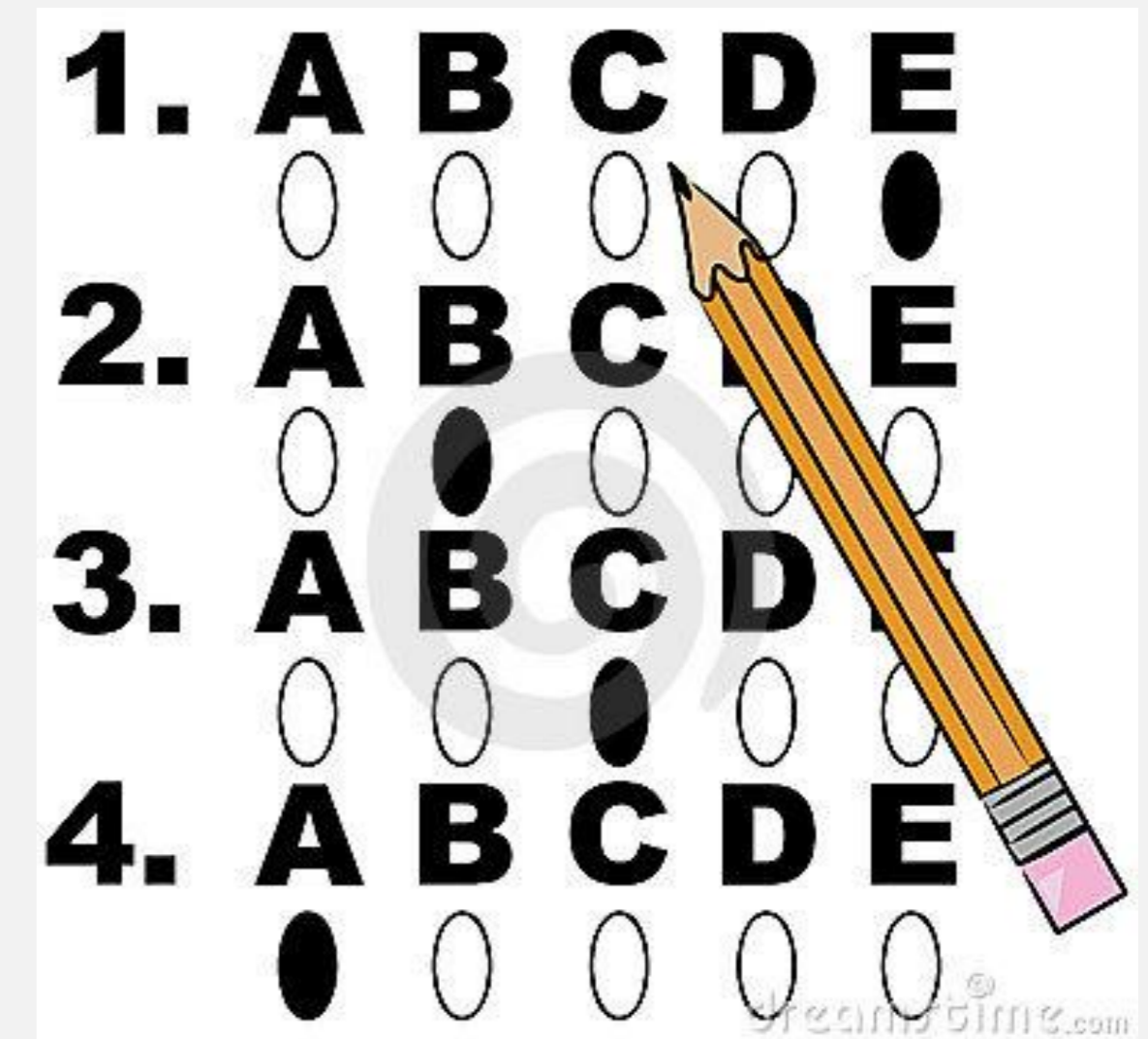
JUnit

- Very common way to test Java functions
 - Define an expected input/output for some portion of code
 - If output changes as a result of Java change, test fails
- Coverage is incomplete due to legacy code
- Goal: should have tests for every new feature



XSLT Unit tests

- Similar idea to JUnit
 - Send some XML construct into a function or template
 - If result changes, test fails
- Mostly covers recently added functions
- New features should provide tests

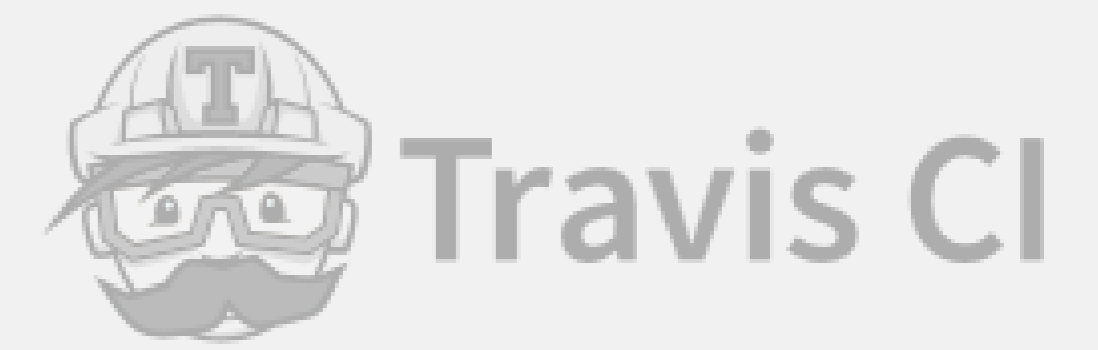


Integration test suite

- Failure = (code is bad) or (test is bad)
- Known issues:
 - Incomplete coverage
 - Overlapping tests
 - Has been difficult to update
- Jarno improving the tests so they're easier to update

pushBefore_with_crossRef	Split conref_push integration test to multiple methods
pushBefore_without_conref	Split conref_push integration test to multiple methods
pushReplace_between_Specialization	Split conref_push integration test to multiple methods
pushReplace_with_InvalidTarget	Split conref_push integration test to multiple methods
pushReplace_with_crossRef	Split conref_push integration test to multiple methods
pushReplace_without_conref	Split conref_push integration test to multiple methods
simple_pushAfter	Split conref_push integration test to multiple methods

Travis CI (Continuous Integration)



- Automatically builds with each commit and with each pull request
- Runs all automated tests and sends report

Travis also handles our nightly builds and release builds

Questions from the audience...?



Questions for the audience...!



How do you test DITA-OT updates?

- *Vendors: how do you test before bundling new version (or do you?)*
- *Anybody: do you test basic DITA-OT functions? Plugin extensions?*



What can we do better...?

- To increase reliability, automation?
- To make it easy for others to contribute test cases (without duplicating existing tests)?
- To guide you on testing your own plugins?



DITA-OT: Further study

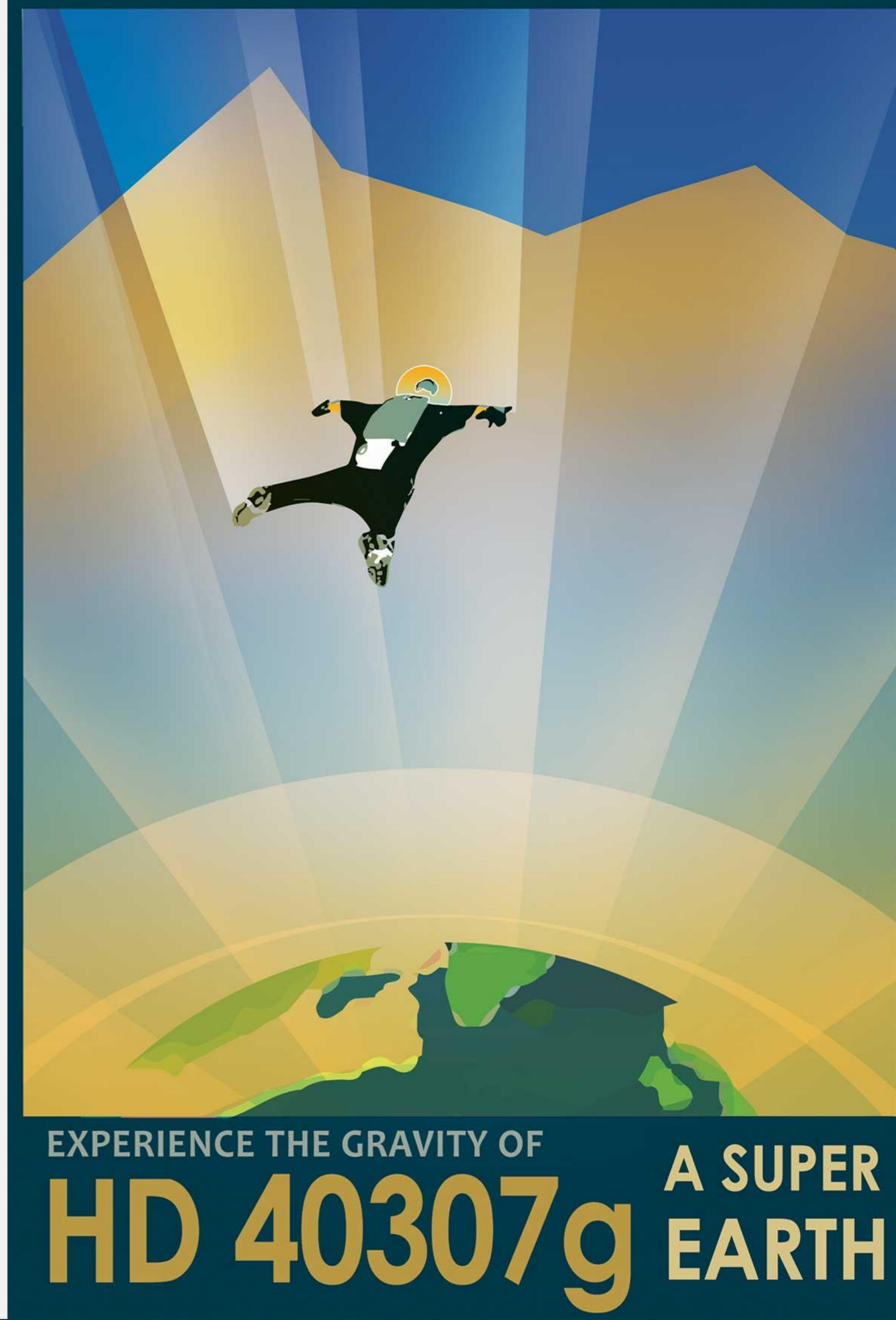
Monthly DITA-OT Contributor calls: hosted by Syncro Soft, open to anyone

Monthly DITA-OT Docs calls: hosted by Eberlein Consulting, open to anyone

Github project: <https://github.com/dita-ot/dita-ot/>

Everything else at <http://dita-ot.org>

Get involved! Please!



Super secret DITA-OT shortcut resources!

Maybe these are useful? If so, tell your friends!

If you'd like to keep your friends, just tell your co-workers!

- <http://code.dita-ot.org> → redirects to <http://github.com/dita-ot/dita-ot/>
- <http://issues.dita-ot.org> → redirects to <http://github.com/dita-ot/dita-ot/issues/>
- <http://wiki.dita-ot.org> → redirects to <http://github.com/dita-ot/dita-ot/wiki>
(Used for contributor meeting minutes)

And in summary...

<https://twitter.com/SwiftOnSecurity/status/685898921063874560>