

# Managing a build environment with [shudder] 50+ custom plugins

DITA-OT Day Berlin, 2017

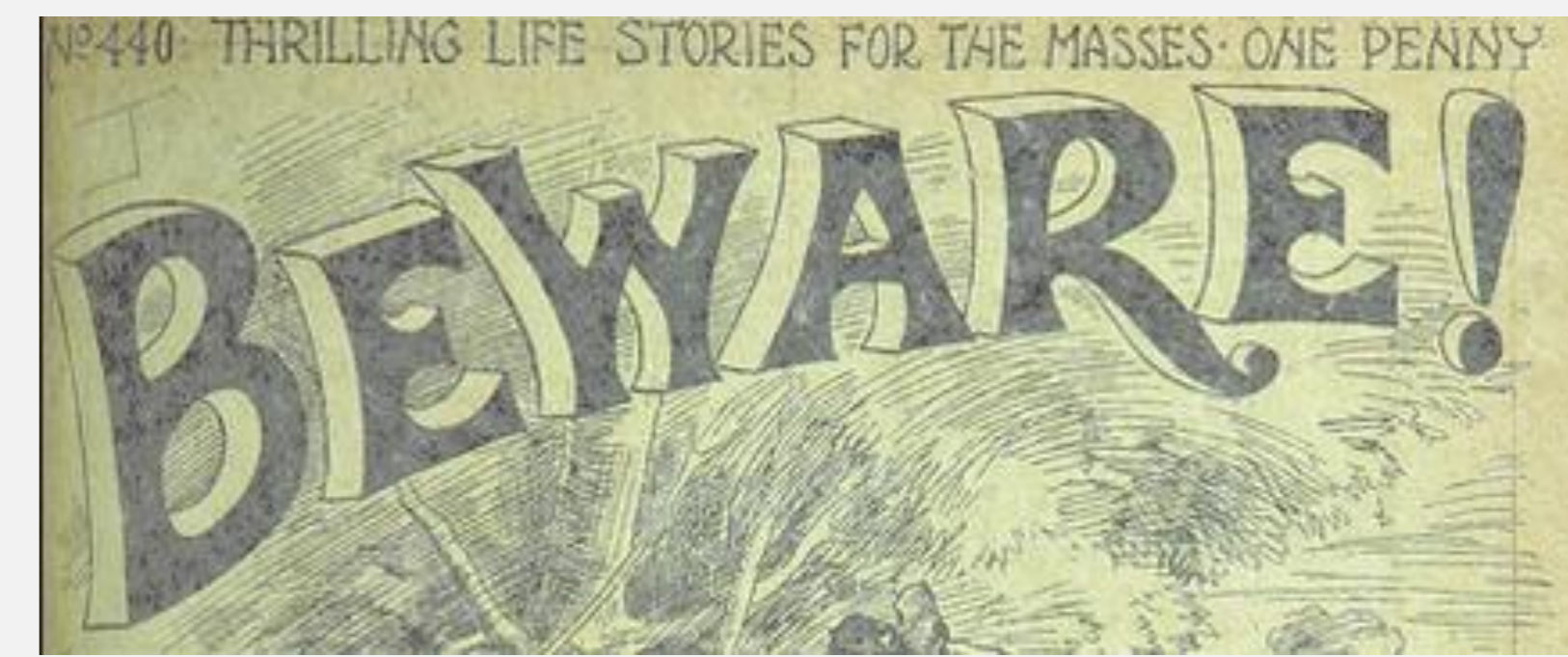
Robert D. Anderson, IBM

[@robander](#)



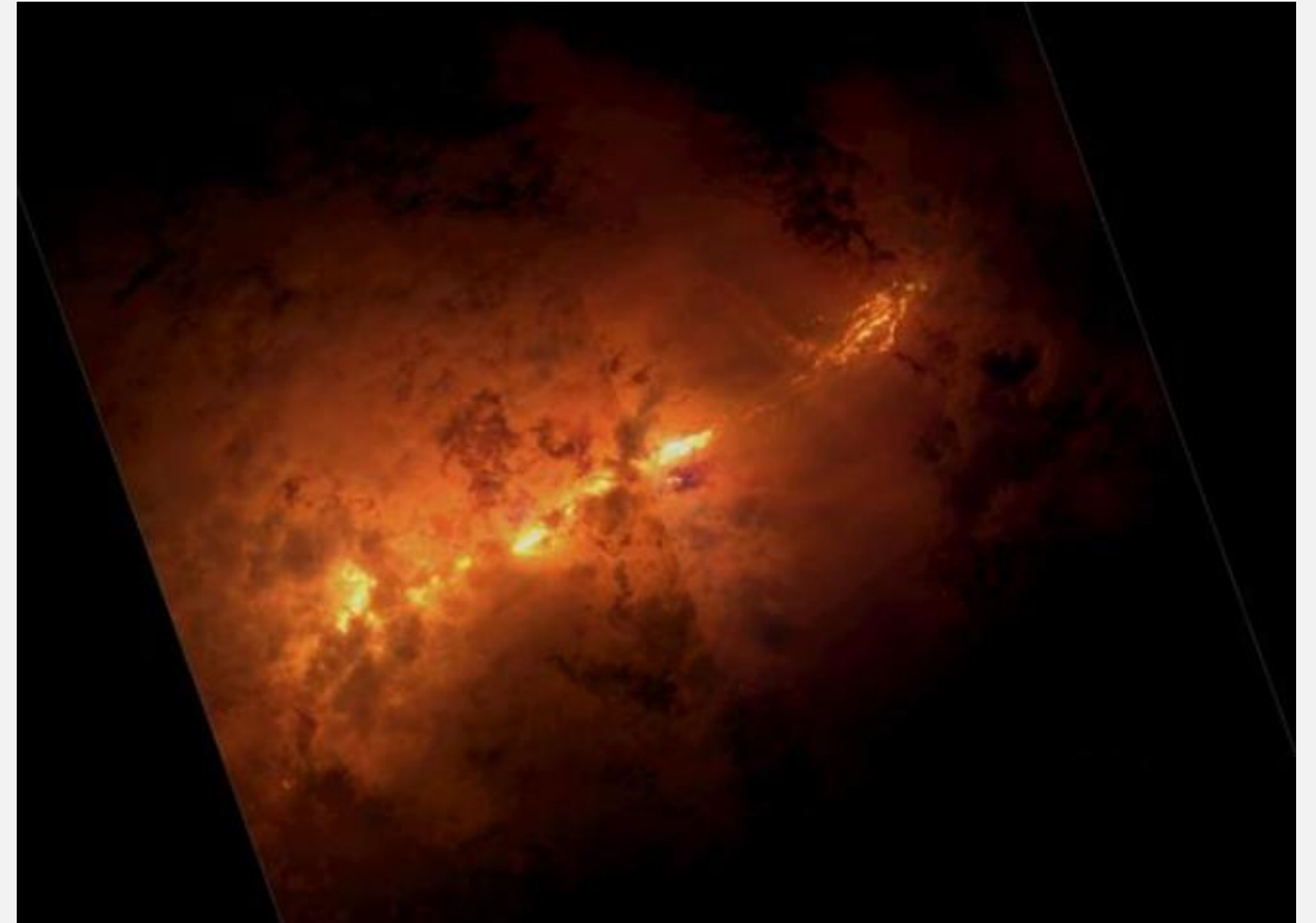
# What I expect you are expecting...

- Plugins!
  - Lots of them!
  - Really, lots!
- Lessons learned!
- A few tips!
- Robert's vacation photos!



# In the beginning (of DITA)

- We (IBM) used an SGML based toolset
- DITA-OT pushed in; used but not “used” directly
  - Pain
  - Frustration
  - Got stuck on DITA-OT 1.7
- More recently:
  - Call DITA-OT directly
  - Everything plugins



# Problems began early

- Adapting “plugins” against 1.7 to ***plugins*** against 2.0
  - We customized everywhere
  - We had “plugins” from before there were ***plugins***
  - We were not in a good spot
- Years of bad design patterns caught up to us
- Few of you should encounter this



# Time to rebuild

- ✓ Make everything a plugin. **Yay!**
- ✓ Refactor for latest code. **Yay!**
  - But ... only as time permits
- ✓ Chance to drop obsolete code. **Yay!**
  - Never as easy as hoped / expected
- ✓ Continuous delivery. **Yay!**
  - Server builds: everyone gets latest!
    - For better, and worse



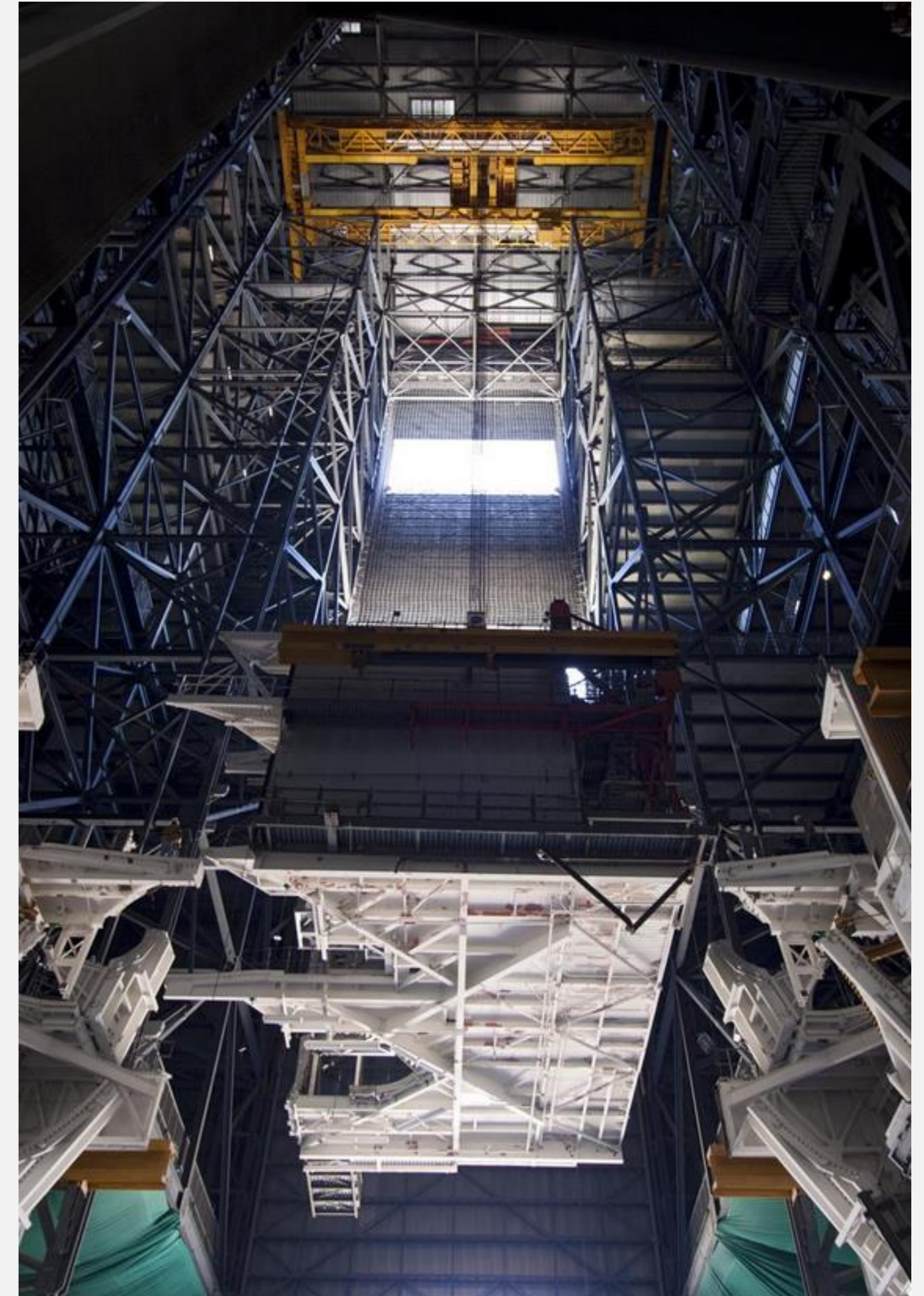
# How far did we get?

- Currently: DITA-OT 2.5.3
- Total plugins today: 66
  - 13 from DITA-OT
  - 2 from OASIS
  - 1 Markdown plugin
  - 1 Syncro plugin
  - 7 related to EPUB
  - 9 Related to Syntax diagrams
  - 33 IBM plugins



# 33 IBM plugins?

- Everybody has access to everything
- Custom document types
  - IBM Topic, IBM Map, IBM Book map...
- Specializations
  - Message topic, Quick Start Guide topic...
- Transform types
  - PDF: Basic IBM Book, Media insert, Quick Start guide...
  - HTML5 for IBM Knowledge Center
  - Plain text
  - ...



# How do we manage all of that?

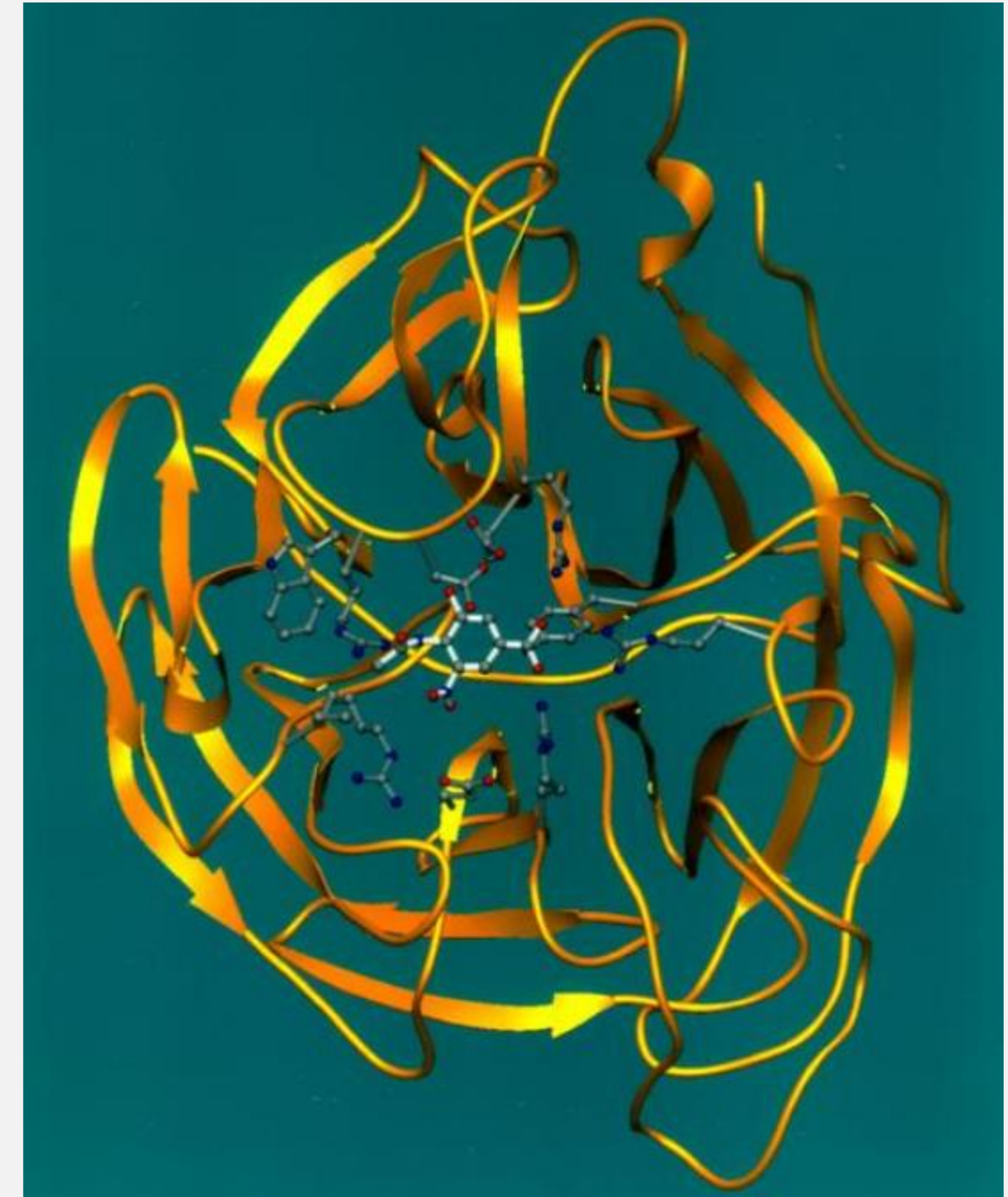
- With occasional difficulty
  - *But so much easier than before!!!*
- Keep a fully-integrated copy of DITA-OT in source code repository, with plugins
- Automated testing of updates
  - Not as automated as DITA-OT itself
  - Would really, really like to do better





# Plugin dependencies: arrrrrgggggghhhh

- HTML+ was full of them
- EPUB used to be full of them
- I added my own, and regretted it
  - IBM topic in one, domains in another
  - IBM topic required domains
  - Message topic required domains
  - Message topic also required processing
  - Processing required...
  - Ended up back in a tangled web



# Debugging: more arrrrrrggggggghhhh

- *Other People's Code...*
- Hard to fix SVG – searching across 9 plugins
- Hard to override EPUB – same
- Hard to find my own rule for `topic/image` processing



# What to try, what to watch out for

1. Remove dependencies when possible, or when practical
2. Upgrade base DITA-OT often
  - Even better: test upcoming releases!
  - *(Should have tested preprocess2 sooner...)*
3. Create new plugin directories
  - Update `configuration.properties`
  - We have `plugins`, `pluginsepub`, `pluginsplus`
  - Could use more? ... but maybe not
4. Only override what's needed



# Does anything go *well* with so many plugins?

Yes!

Incompatibilities are rare

- Documented for recent releases
- Jumping from 1.7 was hard;  
2.0 through to 2.5 not so much

The `plugin:` syntax for imports  
makes shuffling plugins easy



# Further study

Project page and documentation:

<http://dita-ot.org>

GitHub project:

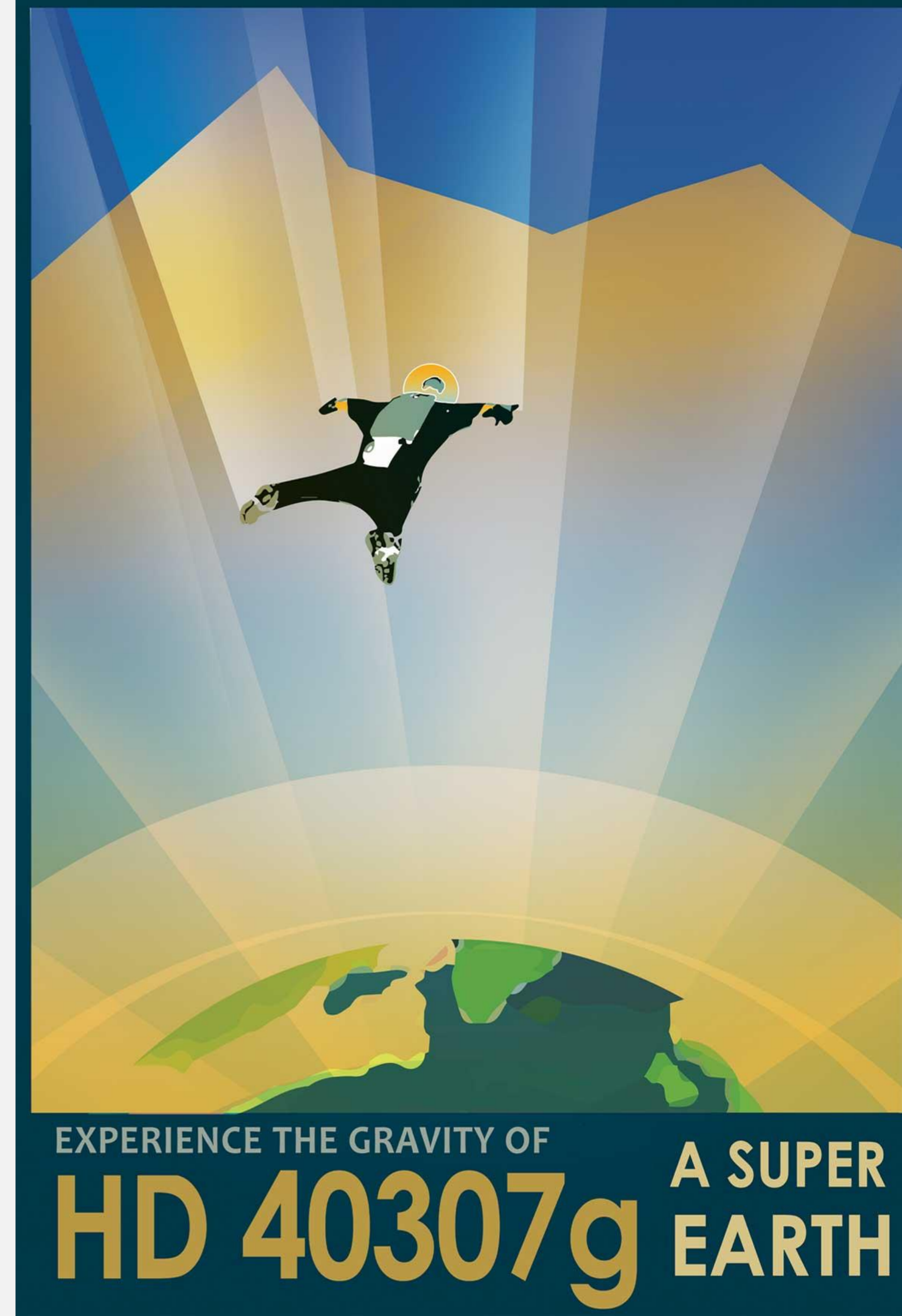
<https://github.com/dita-ot/dita-ot/>

Slack channel:

<http://slack.dita-ot.org/>

Silly stuff (including sample plugins):

<http://metadita.org/toolkit/>



# I am ready for your ideas!

*...I'll also take advice, or questions!*



# Image credits

NASA/JPL-Caltech

<http://www.jpl.nasa.gov/visions-of-the-future/>

<https://images.nasa.gov/>

British Library Flickr stream

[www.flickr.com/photos/britishlibrary/](http://www.flickr.com/photos/britishlibrary/)

New York Public Library Digital Collections

<https://digitalcollections.nypl.org/>

Thank you

Digital Services Group