Third-Party Tools Used at the IETF

draft-nottingham-wugh-services
draft-thomson-github-bcp
History

People started using GitHub for managing their drafts

HTTP started using it “officially” for HTTP/2

Some other working groups copied the model

These were all successful, so more working groups followed
Why does this matter?

The question of what we do - and why - is crucial.

This is about the *how*.

We talk about working groups and finding consensus.

These are important parts of the process.

…and we have a pretty good story for that.
Build RFCs

The same way we build software
Good

Open source

Software
What does that mean?
Quality Assurance

This could mean a lot of things

idnits is useful

The different directorates fill a role analogous to quality assurance or verification; so do area directors

We have a shepherd who is tasked with basic verification tasks

We have some automation, but very little on document content

...YANG sets a good example here
Continuous Integration

Validate as much as we can, as often as possible

1. the document produces valid output (a low bar)
2. code in the document compiles (low value)
3. code in the document is correct (awesome)
4. examples in the document are correct

Keep feedback loops short
4 MONTHS!

Discussion

Build Consensus

Edit

Publish Revision

Meet

Implement

Deploy

Comment/Review/etc…
Real Validation

Comment/Review/etc...

Discussion

Build Consensus

Edit

Deploy

Implement

Meet

Publish Revision
Humans in the Loop

Not a lot we can do to shorten implementation and deployment phases

The hackathon helps (sometimes a lot)

NETVC avoids this problem (not an option for everyone)
Community Involvement

Reach a very large pool of contributors

Allows for distribution of the task of writing text

Easy to contribute -> more contributions

Easy to handle contributions -> more contributions isn’t bad
Additional Factors

Accountability and traceability (issues and PRs record decisions and tie directly into revision history)

Threading and context (discussion on issues is in context)

Minor changes are much easier to handle (inline reviews allow for more contextual comments, which are filtered well; editorial pull requests for grammar and spelling are easy)
The Draft

guide rails for git
Synopsis

What this is

How to decide to use it, what to use it for, how to use it

Guidance on management (overlap with Mark’s draft)

Some advice