

moz://a

Continuous Localization

2.2.2017

Axel Hecht



Agenda

1. Brief History

How we shipped Firefox

2. Continuous Localization

Localizing Firefox Nightly

3. Android

Analysis of our Android Apps

4. Q&A

Brief History

Shipping Localized Firefox

over time

The OLD Days

Firefox < 5

- Localize mostly on the release branch
- We all forgot most of that
 - It was year(s) per branch

Rapid Release

getting closer in 2011

Nightly

- small locale list
- localized builds each day
- directly what's on central
- localizers merge to Aurora

Aurora

- all locales
- 6-ish weeks of string freeze
- localized builds each day
- l10n-drivers merge to Beta

Beta

- release-ready locales
- weekly builds
- signed off by localizers
- reviewed by l10n-drivers
- release-engineering merges to release

Today

post-"Dawn"

- one localization for Nightly, Beta, Release, ESR
- "Herald" notifies on patches touching l10n
- Localization of Nightly, Beta, Release in Pontoon
- Nightly builds twice a day with latest localization
- Beta twice a week with sign-offs by l10n-drivers
- Release/ESR with latest state of that release during Beta, plus fixups

Herald

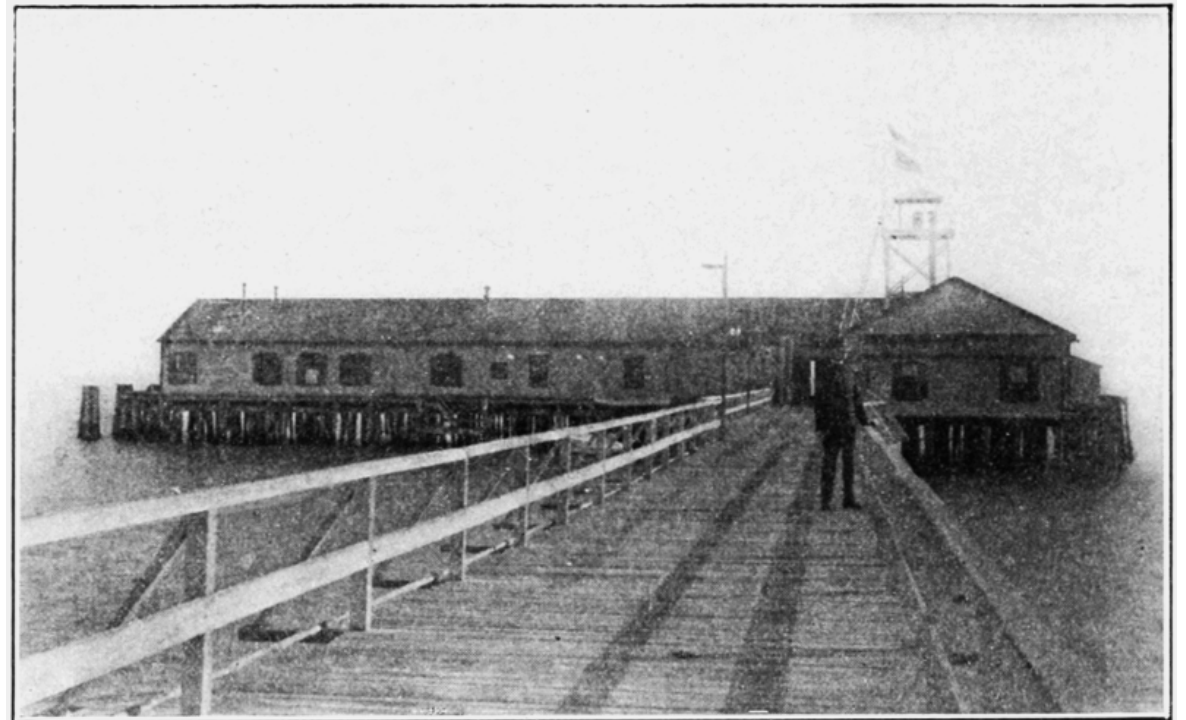
catching localization issues during review

- Herald is part of Phabricator
- CCs l10n-drivers on patches when they get submitted for review
- Developers can't hide ;-)
- We don't know when developers don't touch l10n files, but should.
- Use CODEOWNERS on github to achieve similar results

Quarantine

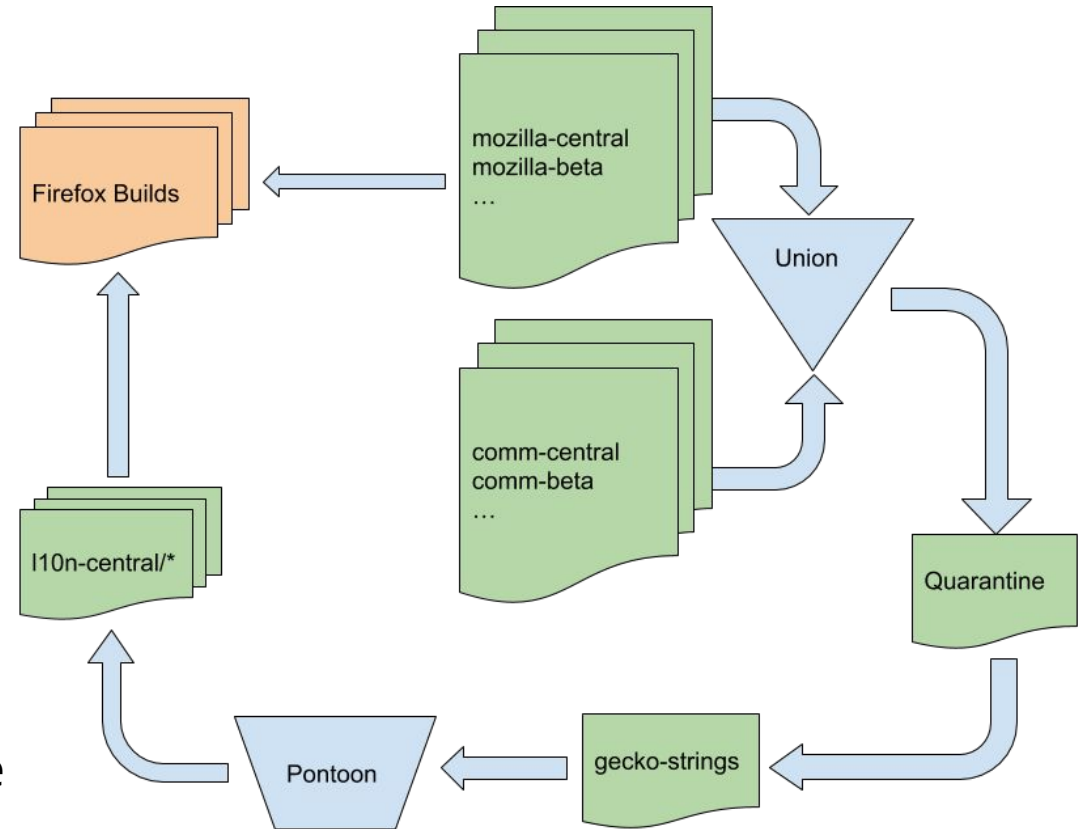
not everything is healthy

- problems for localization still reach mozilla-central
- keep them to just a few localizers to verify
- ask for help if problems arise
- publish known-good states to gecko-strings



gecko-strings

- en-US strings
- contains the unit (super-set) of
 - mozilla-central, mozilla-beta, mozilla-release
 - comm-central, comm-beta, comm-release
- we localize this, and have strings for each channel
 - thus **cross-channel**
- used to build localized versions of Firefox, Fennec, Thunderbird, SeaMonkey
- will include ESR at some point in the future



Android

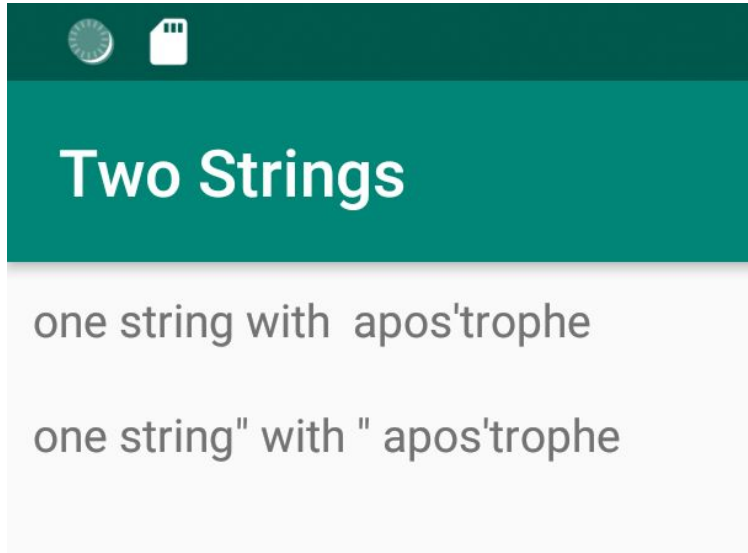
Mobile Apps & More

The good parts

- Android app development puts localizable strings into `strings.xml` files
- Strings have comments, and are maintained as part of the patch
- Engineering teams are set on a linear history in git

All set for continuous localization!

... and the Ugly
... if they'd ever fix that bug ...



```
<resources>  
  <string name="app_name">Two Strings</string>  
  <string name="textView1">one string" with &quot; apos""trophe</string>  
  <string name="textView2">one string\" with \" apos\'trophe</string>  
</resources>
```

Still Ugly

`app/src/main/res/values/strings.xml` is localized in

- `app/src/main/res/values-de-rDE/strings.xml` or
- `app/src/main/res/values-b+sr+Latn/strings.xml`

Locale codes: **in** vs **id**, etc

Documentation only describes best practices, but lacks definition of edge cases

The Way Forward

- Establish a mozilla-supported subset
- Automated tests to enforce that subset
- Support some quirks like locale codes

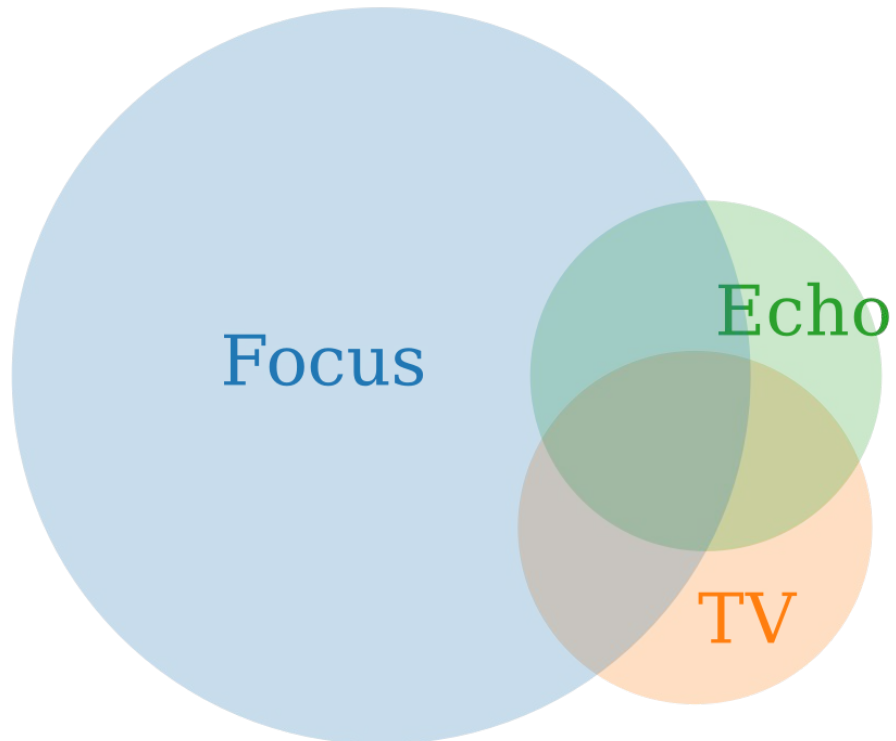
=>

Pontoon can localize Android projects directly.

Mozilla's Android Apps

Many Things Firefox

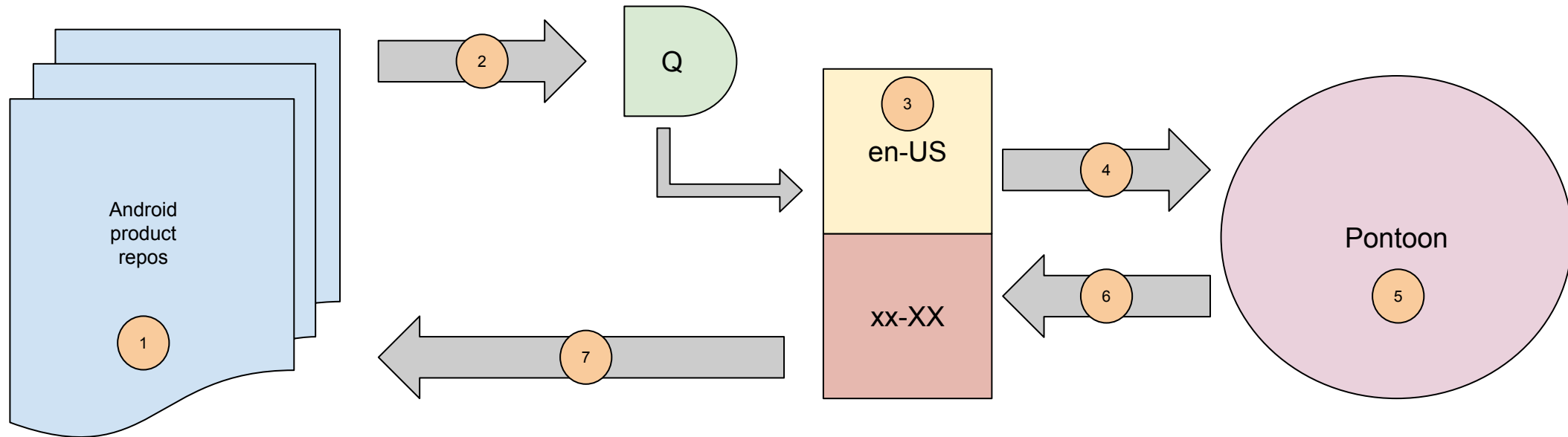
Android Apps



And

- Android Components
- Firefox Lockbox
- Firefox Reality
- Firefox Lite
- Fenix
- ...

Workflow



Builds

- Builds with all languages
- Published on a developer channel with updates
- Some projects will have automated screenshots for testing

moz://a

Thank You