What's New in Gerrit 2.15

Dave Borowitz <dborowitz@google.com>

Google

Dave Borowitz, Gerrit User Summit 2017

I'm the last (real) talk of the day.
I promise to talk quickly and sound excited.
Hopefully some of it rubs off.
Or at least we'll get out early and check out the bar.

Banner says "what's coming" but 2.15-rc0 definitely exists as of 10pm last night

2.15 at a Glance

Time	158 days (Apr 26 - Oct 1)
Commits	2102 (2.15-rc0)
Contributors	53
***New* contributors :)**	7 Brian Bamsch Chih-Hung Hsieh Jean-François Geyelin Mike Samuel Owen Li Vladimir Moskva anatoly techtonik

Commits: git log --oneline --no-merges stable-2.14..v2.15-rc0 | wc -l
New contributors: comm -13 <(git log --no-merges --oneline --format=%an
stable-2.14~ | sort -u) <(git log --oneline --no-merges --format=%an
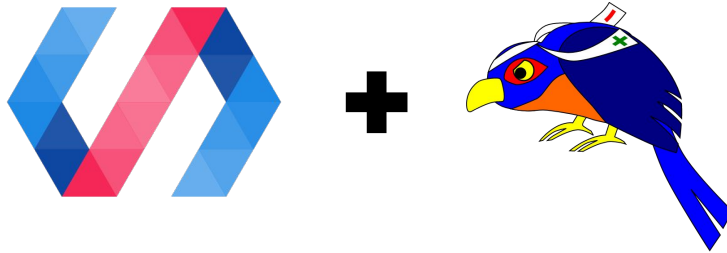stable-2.14..v2.15-rc0 | sort -u)

This isn't the biggest release, based on David P's slides.
I am really only touching on highlights here. Hopefully this is exciting enough to get
you to migrate.

Frontend

I'm happy to say that Gerrit at Google has grown enough that there is a whole *team*
of PolyGerrit people
What I say is my viewpoint as a BE dev/maintainer, but if you are excited about web
UI and frontend dev, you should talk to those fine folks <points at PG team>

Modern   Usable

Modern:
* Nobody accuses GWT of being modern
* Uses real JS
* Easy to hack on, from the browser

Usable:
* Almost a clean slate; GWT UI has lots of inertia
* Easy to hack -> easy to polish (PG team completely rewrote the CSS last night before I cut rc0)

TODO: retake screenshot
I had a screenshot here that I had to redo this morning because of the CSS changes
Show editing

Backend

Dave Borowitz, Gerrit User Summit 2017

NoteDb

Gerrit admin != SQL admin

When I say NoteDb I usually mean for changes
Not going to go into detail on the data model here, I did a talk 1 or 2 years ago that has more detail on the low-level design.
I'm also happy to answer any questions

NoteDb

- **Simplicity**

- **Consistency**

- **Auditability**

- **Extensibility**

- **New features**

Cribbed from Gerrit docs (it's fair, I wrote them)
Simplicity: All data stored in one place (git). No more DB management.
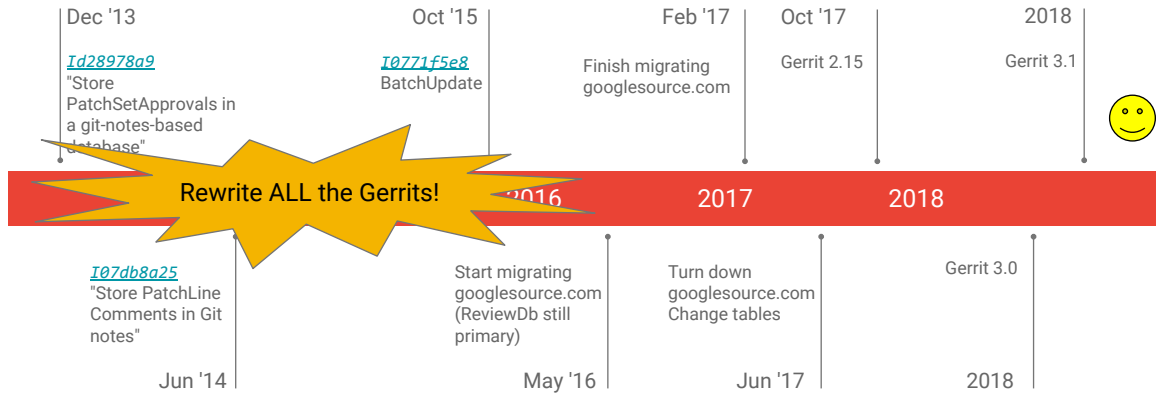Consistency: Replication & backups are consistent snapshots of ref state. Ref
updates are atomic!
Auditability: No mutable records; every modification is a git change. Default
assumption is that every action is logged
Extensibility: Plugins can add new fields without core having to know about them.
New features: Federation; offline code review; interoperation other tools.

NoteDb Timeline

| Dec '13 | Oct '15 | Feb '17 | Oct '17 | 2018 |

Id28978a9
"Store PatchSetApprovals in a git-notes-based database"

I0771f5e8
BatchUpdate

Finish migrating googlesource.com

Gerrit 2.15

Gerrit 3.1

Rewrite ALL the Gerrits!

2016   2017   2018

I07db8a25
"Store PatchLine Comments in Git notes"

Start migrating googlesource.com (ReviewDb still primary)

Turn down googlesource.com Change tables

Gerrit 3.0

| Jun '14 | May '16 | Jun '17 | 2018 |

Google

Off to a slow start in late 2013
In 2014 we got an intern. Yacob was great, but was just about the only help I got on the project for 1.5 years.
Then came a period I like to call "rewrite all of Gerrit"
Started migrating googlesource.com in 2016 (more on that later)
Finished 9 months later--but don't worry, your migration will be faster. We took the pain on ourselves so you don't have to.
Dirty secret: when we started migrating in May, I didn't actually know how to do the primary storage migration

NoteDb Migration

Past	Future
● **googlesource.com**	● **3.0:** Offline only
● **2.15:** Online/offline	● **3.1:** NoteDb only

Documentation

googlesource.com was a hairy migration, and we discovered a lot of issues.
We have a process (not open-source) that test-migrates every single change and
comapres the results between ReviewDb and NoteDb for all our millions of changes.
This was crucial, and gives us pretty high confidence that the migration process will
work in the wild.
Online migration has a phase where we write to both ReviewDb and NoteDb but
ReviewDb is still the source of truth.
I think this is interesting, happy to answer questions about it.

You can also stop in this phase, which we call "trial mode"

Bottom line: we will have the same upgrade process for 2.x -> 3.0 as we have for
other releases
* You can migrate 2.14 -> 2.15 -> 3.0 using only online migrations
* You can migrate 2.14 or earlier directly to 3.0 but it will require an offline migration

Documentation describes the option and their tradeoffs
This doc used to be extremely technical. Now it is less technical.
We still need a "NoteDb quickstart docs"

Demo?

Depending how I feel like I'm doing on time, I might try a demo.
I want to emphasize that this is not super scary!

NoteDb Incentives

- Separate Reviewer/CC
  - With history
  - Add CC by email address
- Hashtags
- Robot comments API
- Ignore change/mark change as reviewed

What I've said so far might make you think migration is scary.
I want to give you some incentives too.
I'll talk about these later:
*Add CC by email
*Ignore/mark as reviewed

NoteDb: Lessons Learned

- Touching every single line of code is hard
  - Shawn hardly recognizes Gerrit internals anymore

- Don't mix storage changes with semantic changes

- Writing tests at head isn't good enough

Mixing changes examples:
* PatchSetApproval 0 votes are treated completely differently, and it's a huge headache.
* Not implementing hashtags in ReviewDb was a mistake
But, we are spinning these as an incentive for you to upgrade, so maybe it'll work out.

Android Gerrit is ancient and has stored data exhibiting every single Gerrit bug you can imagine. It wouldn't be possible to

NoteDb Accounts

- Smaller problem space

- Auto-migrated in 2.15

NoteDb Accounts

```
$ git fetch -q https://gerrit.googlesource.com/All-Users refs/users/self && \
  git cat-file -p FETCH_HEAD:account.config

[account]
        fullName = Dave Borowitz
        preferredEmail = dborowitz@google.com

        status =
```

Thanks Edwin

Features

...that people who don't <3 backend development care about

We have more people working on Gerrit now thinking about the core code review experience.
Many of these features are of the form, "maybe you've always been annoyed that Gerrit does this annoying thing"

Improved difference of patch sets

Highlight diff hunks due to rebase

Original patch set

Patch set 1

my_file.txt	+1 -1
another_file.txt	+4 -4

Base   my_file.txt   PS 1

Base	PS 1
Line 1	Line one
Line 2	Line 2
Line 3	Line 3
Line 4	Line 4
Line 5	Line 5
Line 6	Line 6
Line 7	Line 7
⋮	⋮
Line 999	Line 999
Line 1000	Line 1000

Rebased + corrected patch set

Patch set 2

my_file.txt	+2 -2
another_file.txt	+4 -4

Base   my_file.txt   PS 2

Base	PS 2
Line 1	Line one
Line 2	Line 2
Third line	Third line
Line 4	Line 4
Line 5	Line five
Line 6	Line 6
Line 7	Line 7
⋮	⋮
999th line	999th line
Line 1000	Line 1000

Difference of patch sets: overview level

Patch set 1

| my_file.txt | +1 -1 |
| another_file.txt | +4 -4 |

Patch set 2

| my_file.txt | +2 -2 |
| another_file.txt | +4 -4 |

Patch set 1 → Patch set 2

?

Difference of patch sets: overview level

Patch set 1

my_file.txt   +1 -1
another_file.txt   +4 -4

Patch set 1 → Patch set 2

my_file.txt   +420 -420
another_file.txt   +10 -7

Patch set 2

my_file.txt   +2 -2
another_file.txt   +4 -4

Google

Difference of patch sets: file level

PS 1   my_file.txt   PS 2

PS 1	PS 2
Line one	Line one
Line 2	Line 2
Line 3	Third line
Line 4	Line 4
Line 5	Line five
Line 6	Line 6
Line 7	Line 7
⋮	⋮
Line 999	999th line
Line 1000	Line 1000

Comparison of file contents

Base   **my_file.txt**   PS 1   Base   **my_file.txt**   PS 2

Line 1	Line one		Line 1	Line one
Line 2	Line 2		Line 2	Line 2
Line 3	Line 3		Third line	Third line
Line 4	Line 4		Line 4	Line 4
Line 5	Line 5		Line 5	Line five
Line 6	Line 6		Line 6	Line 6
Line 7	Line 7		Line 7	Line 7
⋮	⋮		⋮	⋮
Line 999	Line 999		999th line	999th line
Line 1000	Line 1000		Line 1000	Line 1000

Google

Difference of patch sets: new version

Patch set 1 → Patch set 2

my_file.txt   +1 -1

PS 1   my_file.txt   PS 2

Line one	Line one
Line 2	Line 2
Line 3	Third line
Line 4	Line 4
Line 5	Line five
Line 6	Line 6
Line 7	Line 7
⋮	⋮
Line 999	999th line
Line 1000	Line 1000

Things to know

- Only available on PolyGerrit

- Reliability?
  - orange/blue: definitely introduced by other changes
  - red/green: no guarantee

Real example

Good example:
https://gerrit-review.googlesource.com/c/gerrit/+/117511/9..10/gerrit-server/src/main/java/com/google/gerrit/server/query/change/ChangeData.java

Change of picture:
https://gerrit-review.googlesource.com/c/gerrit/+/106342/5..6/polygerrit-ui/app/elements/diff/gr-diff-processor/gr-diff-processor_test.html

Draft Changes

- "I only want reviewers to see this"

- "I don't want to spam reviewers/myself until I'm ready"

- "I want to make small edits stored on the server"

- "I wish there were mysterious gaps in my patch set numbers"

  (said nobody ever)

Who thinks they know what draft changes are supposed to be for?
Speaking of things we've been talking about forever
In our experience (Google, community discussions), people were using drafts for
different things...and getting confused when it didn't do what they want

~~Draft Changes~~ Private/WIP Changes

- "I only want reviewers to see this"
  ⇒ **Private change**

- "I don't want to spam reviewers/myself until I'm ready"
  ⇒ **Work-in-progress**

- "I want to make small edits stored on the server"
  ⇒ **Edit** (not new in 2.15)

Private: only visible to reviewers; applies *at the change level*; if you can see one patch set, you can see all patch sets
Work in progress: nobody gets email notifications (even reviewers)

Mark Changes Reviewed

☆	Dissolve gerrit-cache-h2 top-level directory		David Ostrovsky	gerrit	master (flatten-directory-structur...
☆	MoveChange: remove necessary votes after moving	Merge Conflict	Changcheng Xiao (🇩🇪)	gerrit	master
☆	Add StartTLS support for LDAP		Alon Bar-Lev	gerrit	master (ldap)
☆	PolyGerrit: Translate some things into french using new localisation modu...		Paladox none (🇫🇷+🇬🇧)	gerrit	master
☆	Move project dashboard url from /project/ to /admin/project/		Paladox none (🇫🇷+🇬🇧)	gerrit	master
☆	Use SQL for group related tables in schema upgrades		Alice Kober-Sotzek	gerrit	master (sql-for-group-db-access...
☆	SubmitRules: discontinue permission checks	Merge Conflict	Changcheng Xiao (🇩🇪)	gerrit	master
☆	LabelNormalizer: stop squashing based on permitted range	Merge Conflict	Changcheng Xiao (🇩🇪)	gerrit	master
☆	PostReview: Remove support for 'strictLabels' option		Changcheng Xiao (🇩🇪)	gerrit	master
☆	Update gson to 2.8.1		Paladox none (🇫🇷+🇬🇧)	gerrit	master
☆	Add support for a language preference option		Paladox none (🇫🇷+🇬🇧)	gerrit	master
☆	Adapt 'Create Change' client to create WIP rather than draft changes	Merge Conflict	Changcheng Xiao (🇩🇪)	gerrit	master
☆	Add project index tests for both Lucene and ElasticSearch		Xin Sun	gerrit	master (project-index)
☆	Add query methods to project API		Xin Sun	gerrit	master (project-index)
☆	Implementation of project index in Elastic Search		Xin Sun	gerrit	master (project-index)
☆	REST API support for project query		Xin Sun	gerrit	master (project-index)

When there are too many bold changes on my dashboard, I feel like I'm getting yelled at.
Previously, I would have to respond on every one of these to un-bold them, which ironically creates extra noise for reviewees
Now there is a button to ignore
Uses "star labels" new in NoteDb

Ignore Changes

Dave Borowitz, Gerrit User Summit 2017

Or, I can go one step further and ignore changes, using the same mechanism.
This is the same screenshot, this time with all those changes ignored.

Publish Comments On Push

Change 1

Change 2
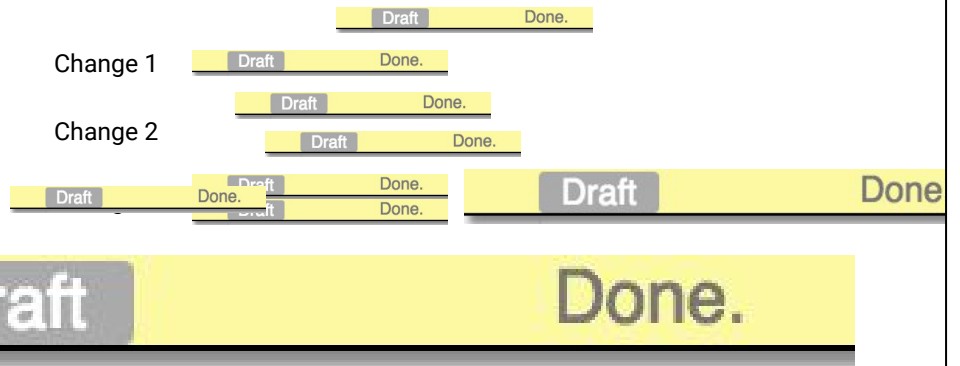
Publish Comments On Push

```
$ git push origin HEAD:refs/for/master%publish-comments
```

Publish comments on push ☑

We intentionally made the push option a little verbose to encourage you to set the preference instead.

CC users by email

Server error: mycoworker@mycompany.com does not identify a registered user or group

Google   Dave Borowitz, Gerrit User Summit 2017

Has this ever happened to you?
(Probably not exactly this because this screenshot is from PolyGerrit)

CC users by email

Reviewers	Add reviewer...
CC	mycoworker@company.com

Say something nice...

Only available on NoteDb
Thanks Patrick

Better error messages

```
$ git push origin HEAD:refs/for/master

...

  ! [remote rejected]      HEAD -> refs/for/master
        (prohibited by Gerrit)
```

Google

Dave Borowitz, Gerrit User Summit 2017

O...k...

Better error messages

```
$ git push origin HEAD:refs/for/master

...

  ! [remote rejected]      HEAD -> refs/for/master
        (prohibited by Gerrit:
         account 1005 lacks permission: forgeCommitter)
```

Thanks Han-Wen. He is good at looking at something and saying "that is dumb, this should be better" and then actually making it better.

Grab bag

- Robots can suggest fixes with one-click application

- conflicts:X queries are much faster

- Lots of email bugs fixed

- Delete inline comments

- Keep track of changes that revert this change

- More server consistency check endpoints

- New extension APIs for plugins, dashboards, …

Google   New URL scheme   Dave Borowitz, Gerrit User Summit 2017

Robot comments are not yet widely used but watch this space
Never had any tests that operation X sends emails to or CC users A, B, C.
Can check for pure reverts of a change, so you can write a bot that auto-submits reverts.
Thanks lots of other people!

Thanks!

Do I have time? Questions? Or we have the panel discussion coming up.