

Code review metadata in Git - Dave, eliminate SQL database, consistent backups, split and restore separate repos to servers
cross-repo atomic submit - work has started
Clustered/multimaster more being released

Consistent backups in Git
Offline review and rich client tools
federated review gerrit $\longleftrightarrow$ gerrit

Uses meta branch under refs/changes, records scores, file comments stored alongside files

cross-repo atomic submit groups related changes, syncs read/write atomically, using git submodule style manifests, XML manifests going away long term, requires cross-repo locking to accomplish (current implementation uses single-file locks)

Putting in more effort towards maturing git-submodule.

Multi-Master -- Horizontally Scale Gerrit
Get more Google Code into Open Source Gerrit

6 Data Centers

With Multiple Masters (18 masters at once , 3 per data center)
Would not recommend this config to others yet. Non-zero number of bugs currently.

140 virtual hosts, 69.3k accounts, 13,150 repos, 4.9TiB (10-12 GB/wk) -- pack only, most daily growth, very little is new repo growth (most repo imports are <1-2GB, growth weekly is an order of magnitude greater), Gerrit metadata is extremely small overall, despite large number of patchsets

Between 15M and 22M reads per day (avg 20M)
Between 2.5K and 8K pushes a day (avg 7K)

Largest 400 GB (Binaries) -- sharing with access controls (also a support issue), create a new branch every commit, disable delta compression (several gigs of history can be difficult, because it tries to process reachability)
Largest refs/repo: 69,295

Garbage Collection runs on dedicated garbage collection machines (2 pools 1 normal, 1 efing large pool) Delta compression is turned off in the repository for the binary project

Replication lag between clusters is usually less than 1 second. Pushes to a master stall until that master replicates to at least 2 other clusters. So usually this means that by the time the user's command finishes, it has already been replicated.

GC driven by an algorithm that watches pushes and keeps counters. Does many Tiny GCs and then accumulates to a larger Full GC.

Chromium 120x tiny GC day (last wednesday) 7 full GC a day

SQL - Google Megastore

Hand Coded Reviewdb 1,418 lines

Git → JGit DFS on Bigtable+GFS

Lucene → Google's Thing