GitLab deployment at CERN

Alex Lossent
IT-CDA-WF
In a nutshell

- **One-stop solution for collaboration on software projects**
  - Code hosting
  - Code review
  - Issue tracking
  - CI/CD
  - Container images

- **Operated by Version Control Systems team**
  - Also running Jira, SVN, Jenkins and CERNForge portal
Use cases (just a few highlights)

- **ATLAS Athena (offline software)**
  - 1.2GB, 30k commits, 100 branches, 4k tags, 1400 forks

- **CERN Datacenter Configuration Management**
  - 560 contributors, 700 projects

- **Docker containers**
  - Build and host Docker images

- **Machine Controls**
  - 1000+ Java projects

- **Preparing physics papers**
  - Collaboration on Latex documents
CERN GitLab in numbers

Projects: 36,949

Users: 13,688

Groups: 17,086

Statistics
- Forks: 7,793
- Issues: 92,742
- Merge Requests: 110,269
- Notes: 1,530,855
- Snippets: 492
- SSH Keys: 13,156
- Milestones: 3,256
- Active Users: 12,442

Features
- Sign up
- LDAP
- Gravatar
- OmniAuth
- Reply by email
- Elasticsearch
- Geo
- Container Registry
- GitLab Pages
- Shared Runners

Components
- GitLab: 11.0.5-ee (d139d22)
- GitLab Shell: 7.1.4
- GitLab Workhorse: v4.3.1
- GitLab API: v4
- Ruby: 2.4.4p296
- Rails: 4.2.10
- postgresql: 9.6.2
- Gitly Servers
Infrastructure

- 7 nodes (10 cores, 30GB)
- Containerized deployment
  - with HA
- Storage:
  - Git repos: 3TB (NFS)
    - 3 volumes, each 1000iops
  - Object storage: 10TB (S3)
    - Docker images: 9.5TB
  - CI artifacts, LFS: 0.5TB
History

- **2014: first experimental deployment**
  - Evaluation of code review capabilities

- **2015: production deployment**
  - Using EE for LDAP groups
  - Contributed Kerberos, SAML support upstream

- **2016: introduced GitLab CI/CD**
  - Complete migration of git repositories out of gitolite

- **2017: move from VMs to Containers in Openshift**
  - Plan for closing SVN service

- **2018: scale out**
  - Infrastructure growing to 200GB RAM, 3 NFS filers
  - Fully automated deployment and integration tests
Version updates

- GitLab releases minor versions on the 22\textsuperscript{nd} of every month
  - No LTS branch: new issues as well as fixed issues in each release
- Usually deployed on gitlab.cern.ch 1-2 month after release
  - Wait for patches to minor releases
  - ...but GitLab only supports the latest 3 releases
- Fully automated process
  - GitLab deploys itself via CI/CD
- Automated tests
  - Account/egroup integration
  - Basic workflow
  - ATLAS workflow

<table>
<thead>
<tr>
<th>OTG0045617</th>
<th>GitLab upgrade to 11.1</th>
<th>29-08-2018 13:30:00</th>
</tr>
</thead>
<tbody>
<tr>
<td>OTG0045370</td>
<td>GitLab upgrade to 11.0 with Global Search</td>
<td>15-08-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0044868</td>
<td>GitLab update to 10.8</td>
<td>10-07-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0044538</td>
<td>GitLab update to 10.7</td>
<td>19-06-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0044104</td>
<td>GitLab upgrade to 10.6 with Git-LFS support</td>
<td>29-05-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0043270</td>
<td>GitLab upgrade to 10.5</td>
<td>09-04-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0042622</td>
<td>GitLab upgrade to 10.4</td>
<td>28-02-2018 13:30:00</td>
</tr>
<tr>
<td>OTG0042136</td>
<td>GitLab upgrade to 10.3</td>
<td>05-02-2018 10:30:00</td>
</tr>
</tbody>
</table>
GitLab CI/CD

- **Rapidly increasing adoption**
  - 140k jobs in 2016
  - 650k jobs in 2017
  - 1.3M jobs in 2018 (so far)

- **GitLab deploys itself via GitLab CI**

![Pipeline Diagram](image_url)
Shared runners at CERN

- Many options tried!
  - Puppet, Swarm, Kubernetes, Openstack VMs via Docker-Machine…
  - VM flavors and various options for Docker storage

- Now using Openstack Magnum Swarm clusters + Ansible
  - Easily provision a set of machines with Docker
  - Leverage Openshift’s Ansible Playbook Bundle for automation

- Shared runner types:
  - Docker + CVMFS (default)
  - Docker Builds – locked on running a custom builder image
  - Docker Privileged (prototype) – using nested VirtualBox VM

- Several hundreds private runners
  - E.g. privileged docker, firmware testing…
Challenges

- **Manage growth of the Docker registry**
  - Registry design requires downtime to delete any image layer
  - Policy to define. Remove old tag versions?
  - Few reliable tools available for registry GC, but GitLab working on it

- **Starter vs. Premium features**
  - E.g. Jira dashboard, cross-project pipelines…

- **External accounts**
  - License costs, some technical difficulties
Challenges: performance

- **Changes in GitLab’s design**
  - All git repository access now handled by a dedicated ‘Gitaly’ service

- **Enables running without shared storage**
  - Gitaly server rather than a NFS server

- **Expected benefits:**
  - An order of magnitude reduction in latency for expansive operations

- **Drawbacks:**
  - Need to operate the storage infrastructure ourselves
  - Snapshots/Off-site replica from Filer service
  - Same SPoF as NFS server
Conclusion

- Our GitLab instance keeps growing steadily
- Opportunities for further performance improvement
Questions?