

Configuration Management with Puppet

puppet at DESY

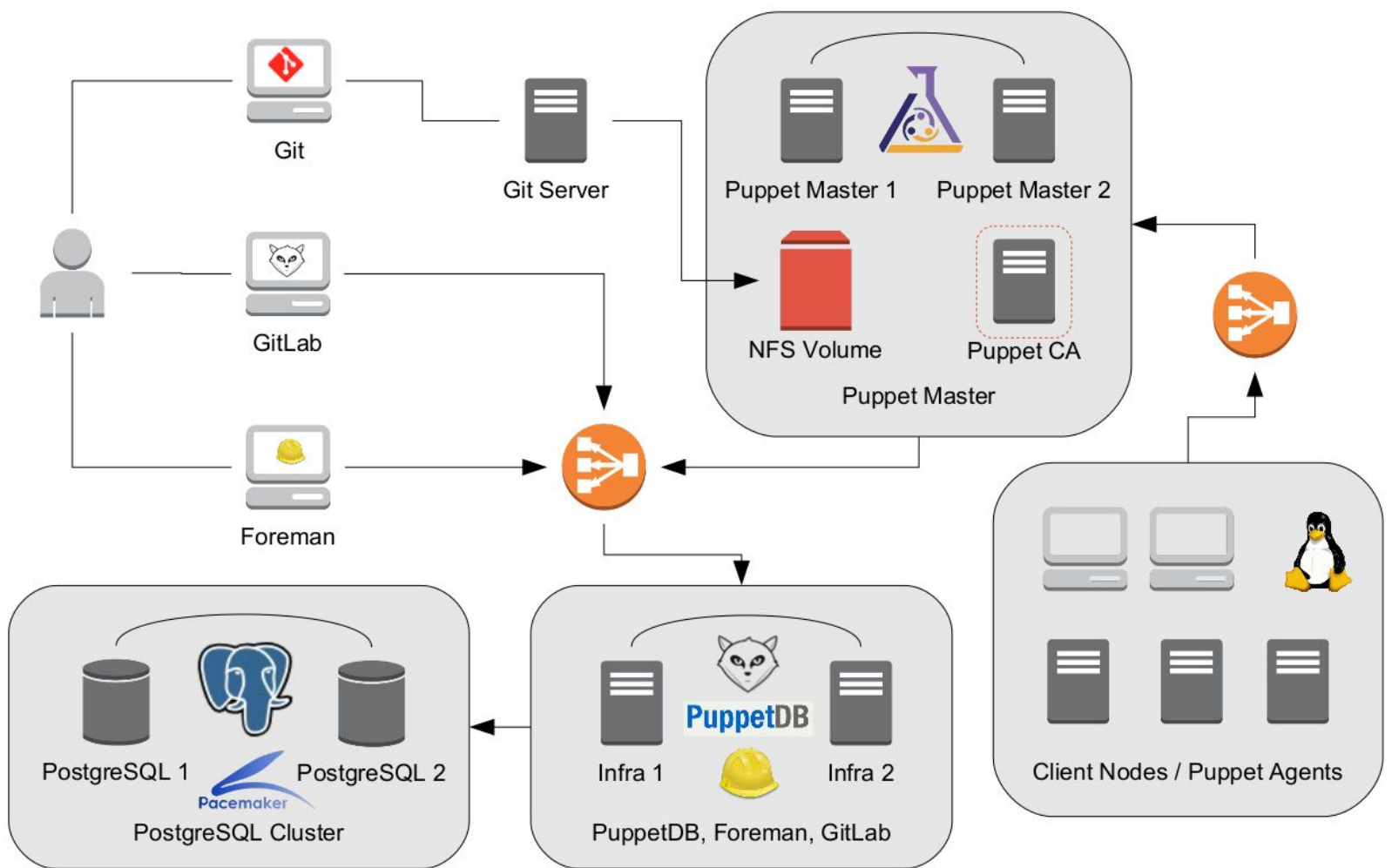


History



- 2012 plan to improve config management
 - A lot of different CM solutions (Salad, Quattor, FAI)
 - balkanized campus
 - OS groups independent (SL vs. Debian)
 - Central IT vs. different *group* IT
- 2013 Start to setup PUPPET
- 2013 Migrate GRID to puppet
- 2016 Replace legacy host registry with foreman

Infrastructure

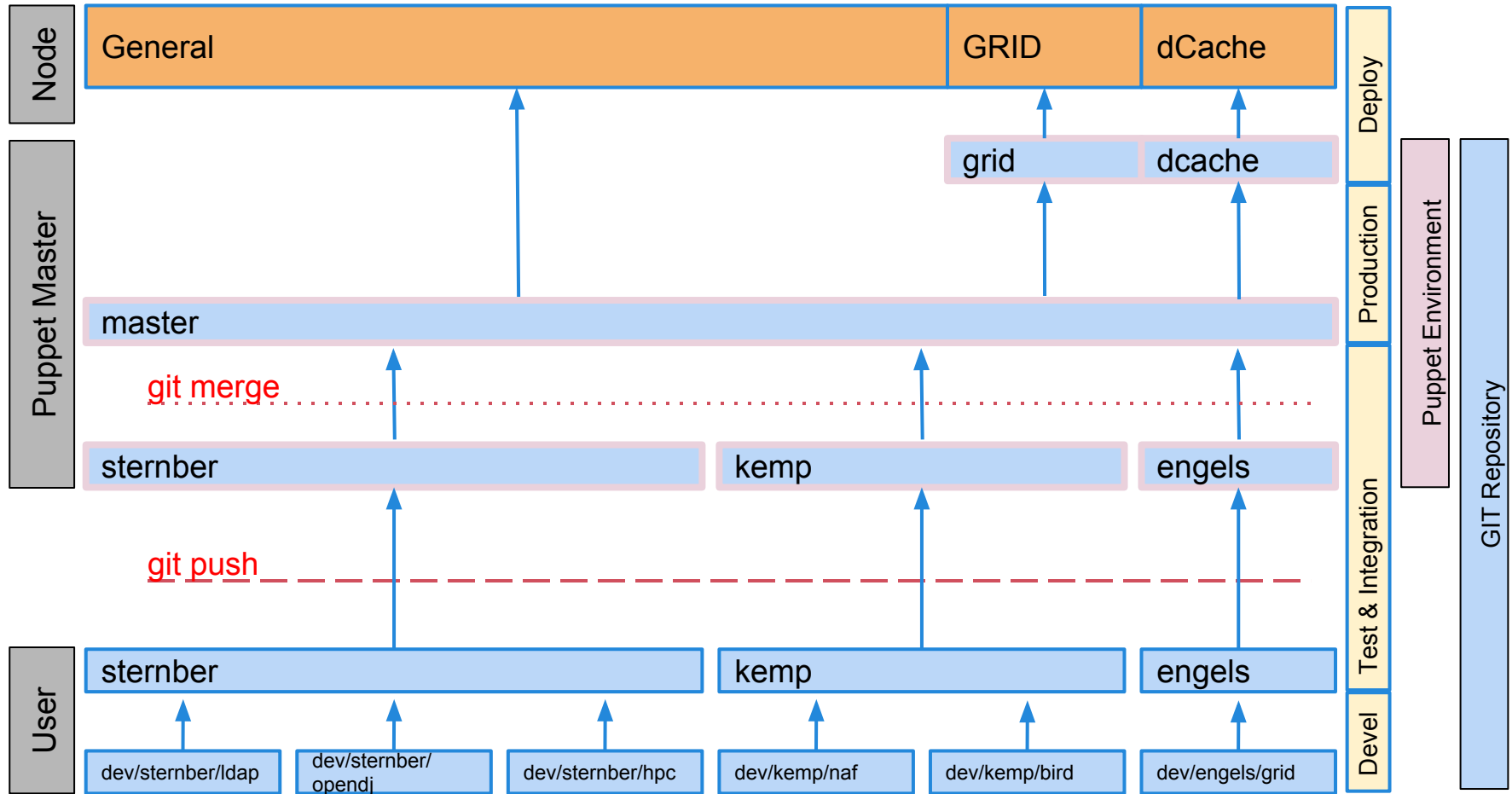




Problems / Remarks

- Quite complex
- Puppet (Ruby) is slow
 - but it scales
- Foreman / PuppetDB
 - redundant data management
 - both great products
- Certificate Handling is complex

Informationflow





Workflow for Puppet User

1. Develop/Improve/Debug feature
2. Test
3. Push feature to git server
4. Integration test with own environment
5. Send pull request to central IT
6. Central IT merge new feature
7. **Deploy commit**



Problems / Remarks

- Every pull request means work for central IT
 - high frequency changes are annoying for central IT
 - Time to wait for the merge is annoying for users
- User start to use their own environment for production
- User want access to the master
- GIT is for occasional user difficult to master
 - We hope GITLAB will help
- Deploy step *sounds* good ...
 - Need mechanism to control puppet run instead

Multiple Repositories



- One repository per feature
 - difficult to manage
 - how to configure environments?
 - how to update environments?
 - very flexible
 - changes to the master could be done by the developer of the feature
 - groups would be responsible for their features

Success



- One platform for Debian, Ubuntu, RedHat and Scientific Linux
- The integration of all IT groups starts very well.
 - Several trainings all well-attended
 - Version control helps a lot
- Puppet ecosystem is helpful
 - HEP community
 - PuppetForge
 - Documentation
 - Courses, Certification, Conferences ...