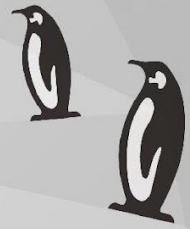
Integrating GitLab into the Red Hat kernel workflow

Don Zickus
Kernel Engineering

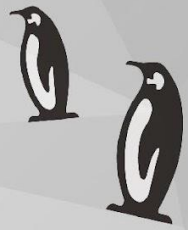
Prarit Bhargava
Kernel Engineering

LINUX PLUMBERS CONFERENCE › September 20-24, 2021

LINUX
PLUMBERS
CONFERENCE > September 20-24, 2021

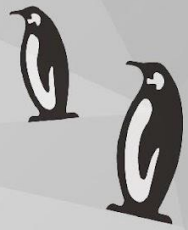


How to get Linus to accept a Pull Request from GitLab?



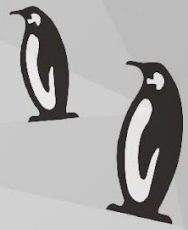
Motivation?

- Upstream discussions about using git forges
- Red Hat's modernization effort led to the migration of its kernel workflow to GItLab
- Red Hat offering to lead this effort



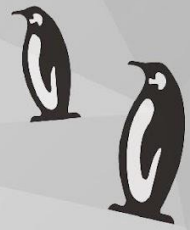
Workflow Issues?

- Maintainer and emails (Patchwork)
  - tracking reviews
  - tracking versions
  - tracking build/test results
  - High volume manual work

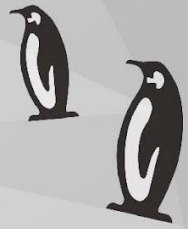


Workflow Issues?

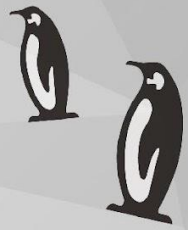
- Reviewers struggled to find what to review
- CI was a bolt on
- Business logic not front and center



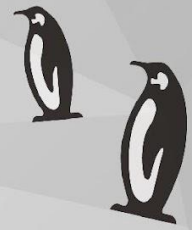
WHAT WE DID
(Why we transitioned a large development team to GitLab)



- GitLab WebUI is fine but command line is better
- https://github.com/zaquestion/lab
- Red Hat kernel development recommended tool
  - Open Source
  - Easy to use

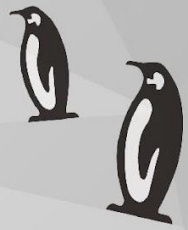


- Maintainers Scripts and Automation
  - Each have their own scripts
  - Replaced by webhooks
  - https://gitlab.com/cki-project/kernel-webhooks
  - Checks Signed-off-by:, Bugzilla:, etc.

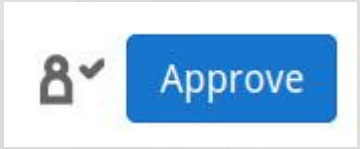


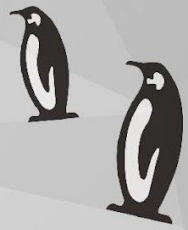
- GitLab Labels!
- WebUI (right)
- Command line ‘lab mr show’ (below)

```
Labels: Acks::NeedsReview, Acks::x86::NeedsReview, Bugzilla::NeedsReview, CKI::OK, CKI_RT::OK, CommitRefs::OK, Depende
ncies::OK, KABI, Signoff::OK, Subsystem:scheduler, Subsystem:x86
```

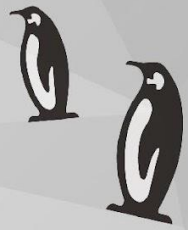


- Data tracking
  - **Not easy to track reviews**
  - GitLab organizes comments per Merge Request
  - Easy to track reviews through Approve button
    - ‘lab mr approve’

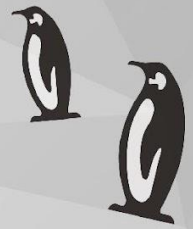




- Who can do a review? Who MUST do a review?
  - Upstream uses MAINTAINERS and get_maintainer.pl
  - Red Hat uses owners.yaml and approvers webhook
  - Reviews
    - Required approvers and reviewers
  - Acks:: labels
    - Acks::x86:NeedsReview
    - Acks::pci::OK

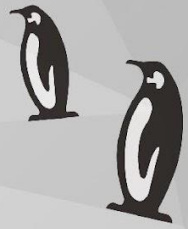


- CI
- Many different CI running upstream
  - Notifications come late
- GitLab easily integrates CI
  - Reporting failures to contributors
  - Excluding infrastructure failures
  - Test feedback and modifications
  - Red Hat CKI
    - https://gitlab.com/cki-project

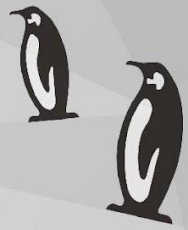


- Easier to do reviews
  - RSS NewsReader
  - Reviewers now work out of a git repository
  - Can comment on commits using lab
    - lab mr block --commit <gitCommitID>

```
[prarit@prarit rhel-8]$ lab mr checkout 1308
Deleted branch 1999550 (was 3dfbafcc49e8).
From gitlab.com:redhat/rhel/src/kernel/rhel-8
 * [new ref]                 refs/merge-requests/1308/head -> 1999550
Switched to branch '1999550'
[prarit@prarit rhel-8]$ git log --oneline -2
3dfbafcc49e8 (HEAD -> 1999550, prarit/1999550, origin/merge-requests/1308) x86/mce: Delay clearing IA32_MCG_STATUS to
the end of do_machine_check()
99354f626c16 x86/mce: Send #MC singal from task work
[prarit@prarit rhel-8]$
```



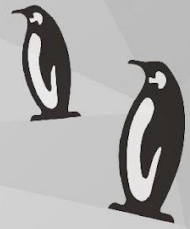
- ARK: https://gitlab.com/cki-project/kernel-ark (no BZ required!)
- c9s: https://gitlab.com/redhat/centos-stream/src/kernel/centos-stream-9
- documentation: https://red.ht/kernel_workflow_doc
- tools (git-backport): https://gitlab.com/redhat/centos-stream/src/tools
- lab: https://github.com/zaquestion/lab



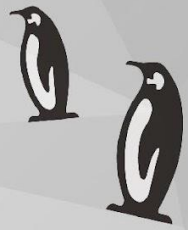
GitLab Relationship

- resolved over 15 issues in the last 6 months
- dedicated developer pushing changes (2 accepted contributions) [Lucas Zampieri]
- strategic partnership

LINUX
PLUMBERS
CONFERENCE > September 20-24, 2021

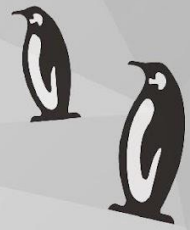


How to get Linus to accept a Pull Request from GitLab?



Known Issues

- Chain of trust
- Merge request logs
- Single Point of Failure
  - Bread crumb trail of discussions
  - Mirror of git trees



Thoughts?