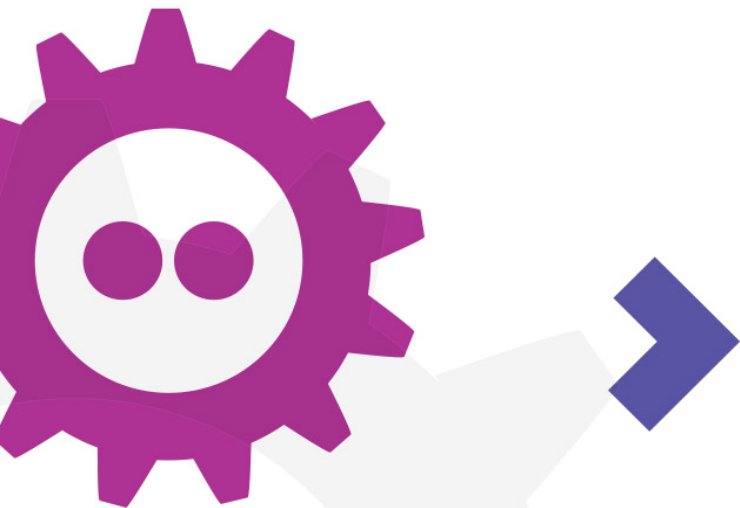




COLLABORA



Welcome to KernelCI

Guillaume Tucker
gtucker@collabora.com
01/02/2020

FOSDEM^{'20}

What is the purpose of KernelCI?

- Aspiring to the utopia of a bug-free mainline kernel
- Kernel code quality control
- Growing test coverage
- Community-driven
- A reusable tool

Mission Statement

« To ensure the quality, stability and long-term maintenance of the Linux kernel by maintaining an open ecosystem around test automation practices and principles »

A Linux Foundation project...

- Collabora
- BayLibre
- Red Hat
- Google
- Microsoft
- Foundries.io
- Civil Infrastructure Platform



Find out more on <https://foundation.kernelci.org/>

...serving the community

- kernelci.org remains focused on upstream
- E-mail reports sent to public mailing lists
- Engaging with subsystem maintainers
- Encouraging more contributors

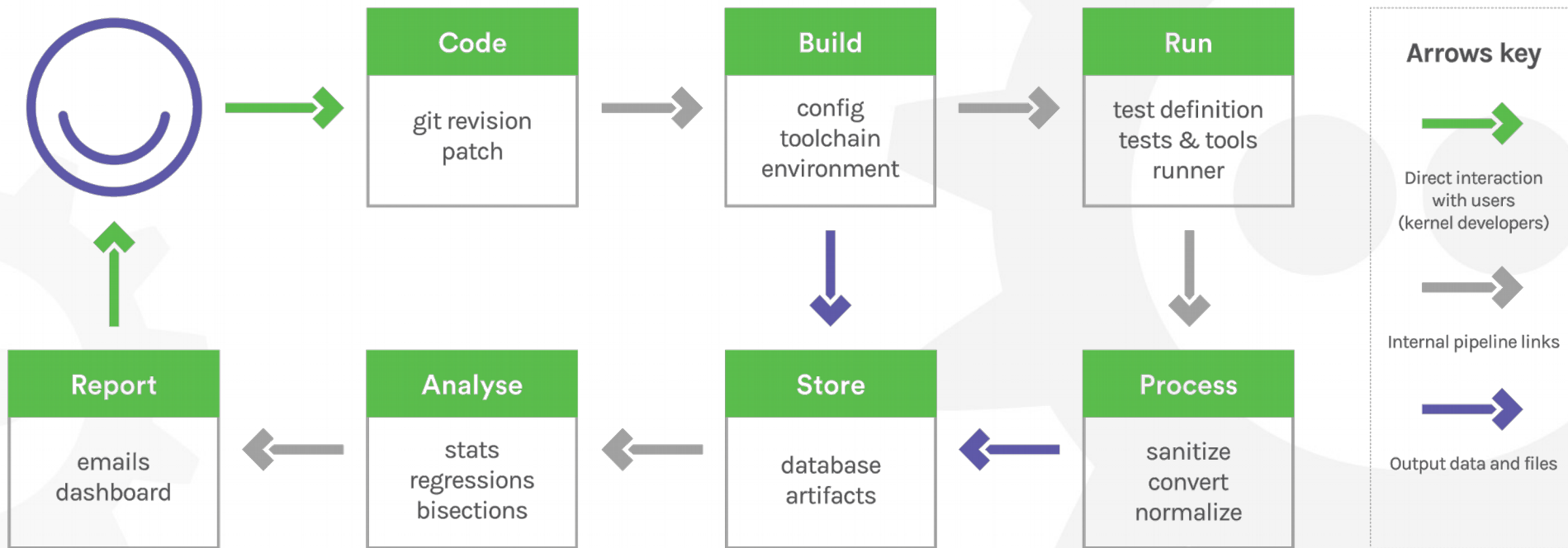


Open Testing Philosophy

- There is only one mainline kernel
 - Many very different use-cases
 - Large number of contributors
 - Common abstract APIs
 - Rich configuration system
- ⇒ Create a test system based on the same principles



KernelCI Modular Pipeline



For more details, see also the [KernelCI Modular Pipeline v2 document](#)

Modular in practice

- Common results database: kcidb and BigQuery
- Kernel builders from various systems (GCE, Azure...)
- Not tied to Jenkins, all the steps can be run anywhere
- Lab API abstraction: only LAVA for now, others possible



Current development goals

- Portable command line tools (`kci_build`, `kci_test...`)
- Enable developers to contribute tests and hardware
- Run more tests: `kselftest`, LTP, `i-g-t`, Kunit, `hackbench`
- Basic but clear web dashboard for test results
- Shorter email “notification” reports
- Advanced test bisection for complex test suites



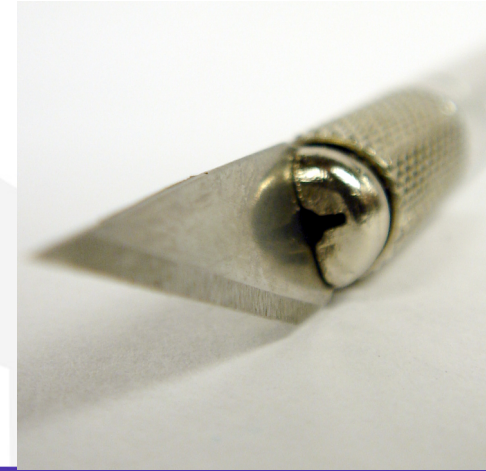
Reusable tools

- Core functions: `kci_build`, `kci_test`, `kci_rootfs`...
<https://github.com/kernelci/kernelci-doc/wiki/KernelCI-command-line>
- BigQuery interface: `kcidb` <https://github.com/kernelci/kcidb>
- Docker containers (work in progress):
 - LAVA
 - Jenkins with pre-configured jobs
 - kernelci-backend
 - kernelci-frontend



A new bisection tool: scalpel

- Proof-of-concept tool inspired by EzBench
<https://gitlab.collabora.com/gtucker/scalpel.git>
- Aiming to be generic better-than-git bisect
- Ready to start becoming part of KernelCI



Tests wish list

- Linux kernel test tools:
 - kselftest, KASAN, UBSAN, Kunit, coccinelle...
- Comprehensive test suites:
 - Linux Test Project
- Subsystem test suites
 - v4l2-compliance, i-g-t, xfstests...



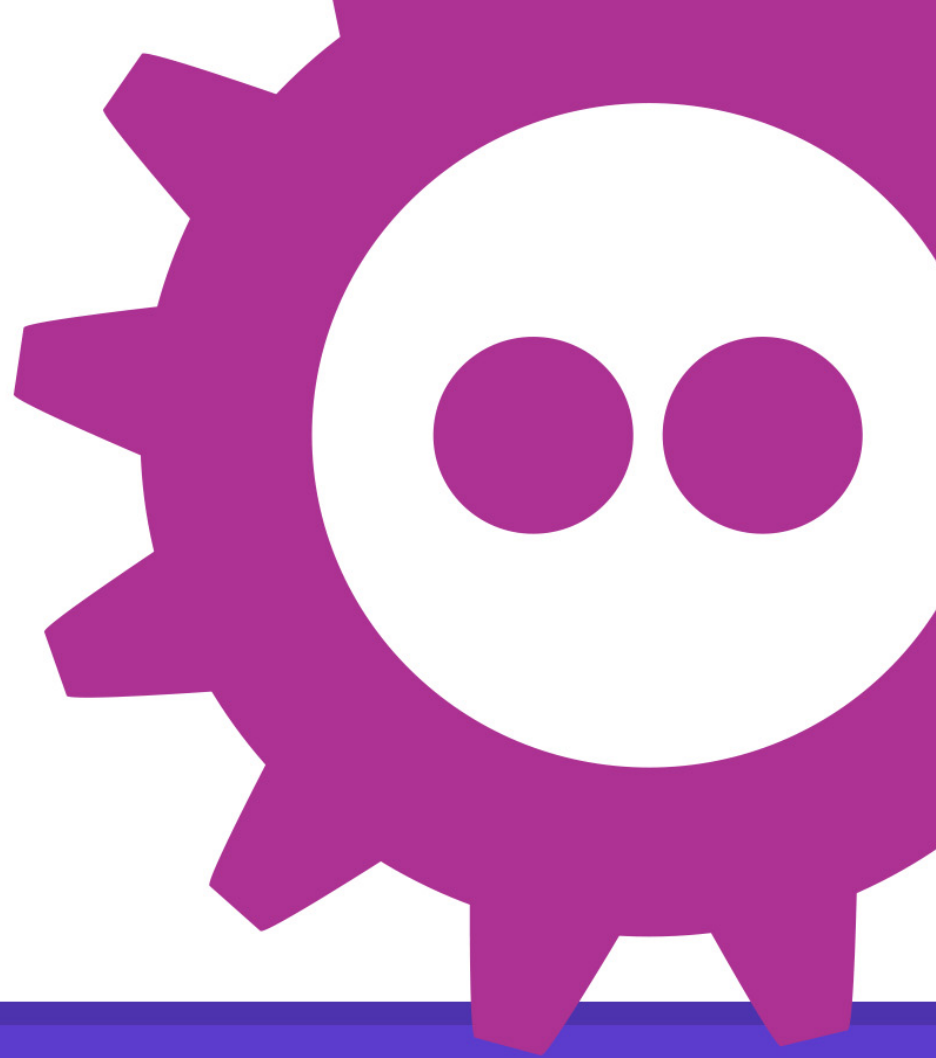
FOSDEM'20

Welcome to



KernelCI

Questions?



COLLABORA

