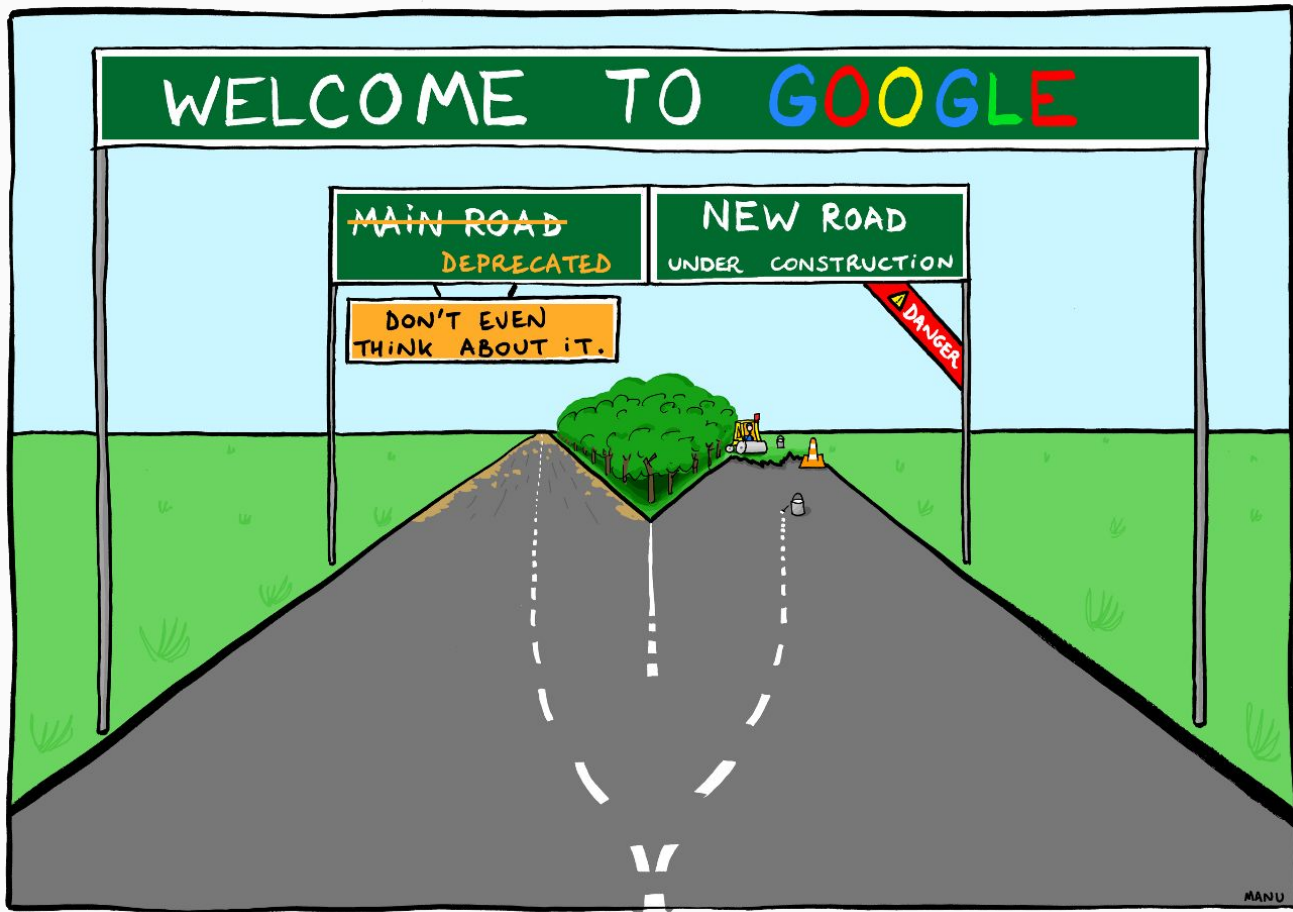




LLVM Premerge Testing: Current State and Next Steps



The Deprecated Thing

Premerge Testing: A Review



Some checks haven't completed yet

[Hide all checks](#)

2 skipped, 4 successful, and 1 pending checks

	Labelling new pull requests / greeter (pull_request_target) Skipped	Details
	Check code formatting / code_formatter (pull_request) Successful in 3m	Details
	Check for private emails used in PRs / validate_email (pull_request) Successful in 1m	Details
	Labelling new pull requests / automate-prs-labels (pull_request_target) Skipped	Details
	buildkite/github-pull-requests Pending — Build #108228 started	Details
	Graphite / ai_reviewer check — Graphite AI review summary	Details



This pull request is still a work in progress

Draft pull requests cannot be merged.

[Ready for review](#)



This branch is out-of-date with the base branch

Merge the latest changes from `main` into this branch.
This merge commit will be associated with `luci.the.rose@gmail.com`.

[Update branch](#)



[Squash and merge](#)



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Premerge Testing: A Review



Some checks haven't completed yet

2 skipped, 4 successful, and 1 pending checks

[Hide all checks](#)



Labelling new pull requests / greeter (pull_request_target) Skipped

[Details](#)



Check code formatting / code_formatter (pull_request) Successful in 3m

[Details](#)



Check for private emails used in PRs / validate_email (pull_request) Successful in 1m

[Details](#)



Labelling new pull requests / automate-prs-labels (pull_request_target) Skipped

[Details](#)



buildkite/github-pull-requests Pending — Build #108228 started

[Details](#)



Graphite / ai_reviewer check — Graphite AI review summary

[Details](#)



This pull request is still a work in progress

Draft pull requests cannot be merged.

[Ready for review](#)



This branch is out-of-date with the base branch

Merge the latest changes from `main` into this branch.
This merge commit will be associated with luci.the.rose@gmail.com.

[Update branch](#)



[Squash and merge](#)



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).


Premerge Testing: A Review

Buildkite LLVM project Pipelines Test Suites Package Registries Log In Sign Up Learn More...

Github pull requests Public Settings New Build
Pipeline running BuildKite pre-commit CI from Github Pull Requests

[clang] assume_aligned incorrectly diagnoses a dependent return type Running for 15m 12s
< #108258 / @ AmrDeveloper:clang_assume_aligned_dependent_ret_type / @ 0f1405efdbb6a9 (Pull Request #111573)

✓ 🌐 🕒 🐧 Linux x64 🔄 🪟 Windows x64

 **Amr Hesham**
Created today at 4:41 PM | Triggered from Webhook

Jobs Canvas New Waterfall

All Failures

✓ 🌐 `.ci/generate-buildkite-pipeline-premerge > pipeline.yaml && buildkite-agent pipeline upload pipeline.yaml` Waited 9s · Ran in 15s ⚙️ Agent

🕒 🐧 **Linux x64** `./ci/monolithic-linux.sh "clang;clang;lld;clang-tools-extra;compiler-rt;llvm" "check-all check-clang c...` Waiting for agent 15m 6s ⊞ Unclustered

🔄 🪟 **Windows x64** `C:\BuildTools\Common7\Tools\VsDevCmd.bat -arch=amd64 -host_arch=amd64 && bash .ci/monolithic-win...` Waited 9s · Running for 14m 57s ⚙️ Agent

🕒 Total Job Run Time: 14s

Premerge Testing: A Review

Buildkite LLVM project Pipelines Test Suites Package Registries Log In Sign Up Learn More...

Github pull requests Public
Pipeline running BuildKite pre-commit CI from Github Pull Requests Settings New Build

[clang] assume_aligned incorrectly diagnoses a dependent return type Running for 15m 12s

< #108258 / @AmrDeveloper:clang_assume_aligned_dependent_ret_type / 0f1405efdbb6a9 (Pull Request #111573)

✓ Linux x64 Windows x64

Amr Hesham
Created today at 4:41 PM | Triggered from Webhook

Jobs Canvas New Waterfall

All Failures

✓ Linux x64 `.ci/generate-buildkite-pipeline-premerge > pipeline.yaml && buildkite-agent pipeline upload pipeline.yaml` Waited 9s · Ran in 15s Agent

⌚ Linux x64 `./ci/monolithic-linux.sh "clang;clang;lld;clang-tools-extra;compiler-rt;llvm" "check-all check-clang c..."` Waiting for agent 15m 6s Unclustered


⌚ Windows x64 `C:\BuildTools\Common7\Tools\VsDevCmd.bat -arch=amd64 -host_arch=amd64 && bash .ci/monolithic-win...` Waited 9s · Running for 14m 57s Agent

© Total Job Run Time: 14s

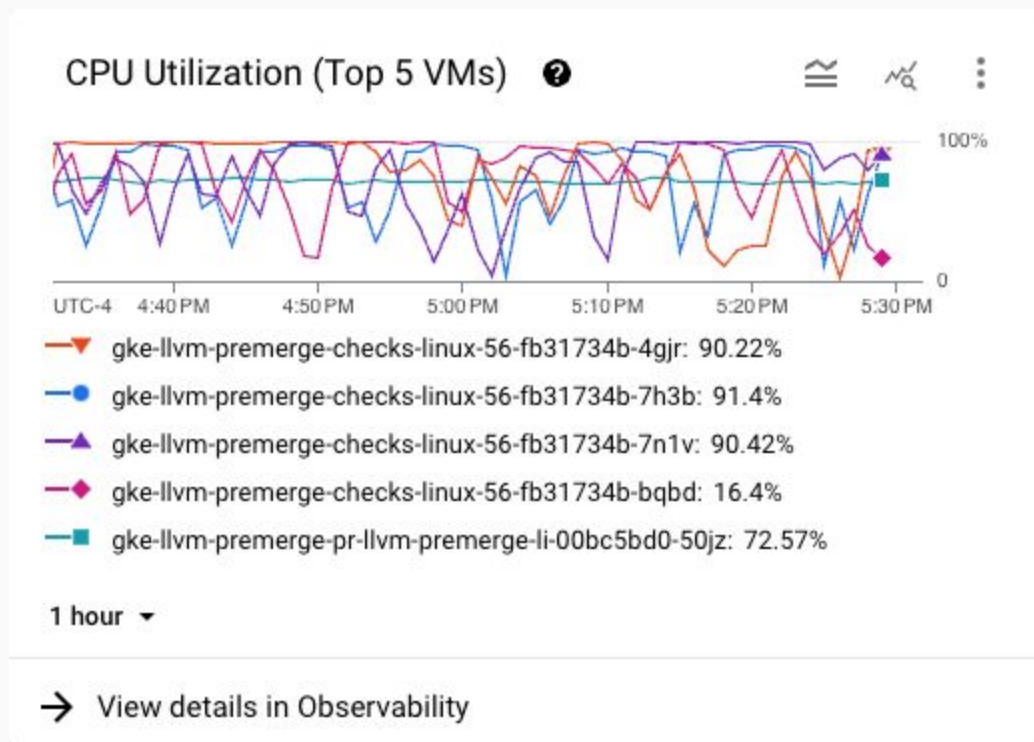
Premerge Testing: A Review

✓  `.ci/generate-buildkite-pipeline-premerge > pipeline.yaml && buildkite-agent pipeline upload pipeline.yaml` Waited 9s · Ran in 15s ⚙️ Agent

🕒  **Linux x64** `./ci/monolithic-linux.sh "clang;clang;lld;clang-tools-extra;compiler-rt;llvm" "check-all check-clang..."` Waiting for agent 36m 52s ☰ Unclustered

✓  **Windows x64** `C:\BuildTools\Common7\Tools\VsDevCmd.bat -arch=amd64 -host_arch=amd64 && bash .ci/monolithic-windows...` Waited 9s · Ran in 23m 32s ⚙️ Agent

Premerge Testing: A Review

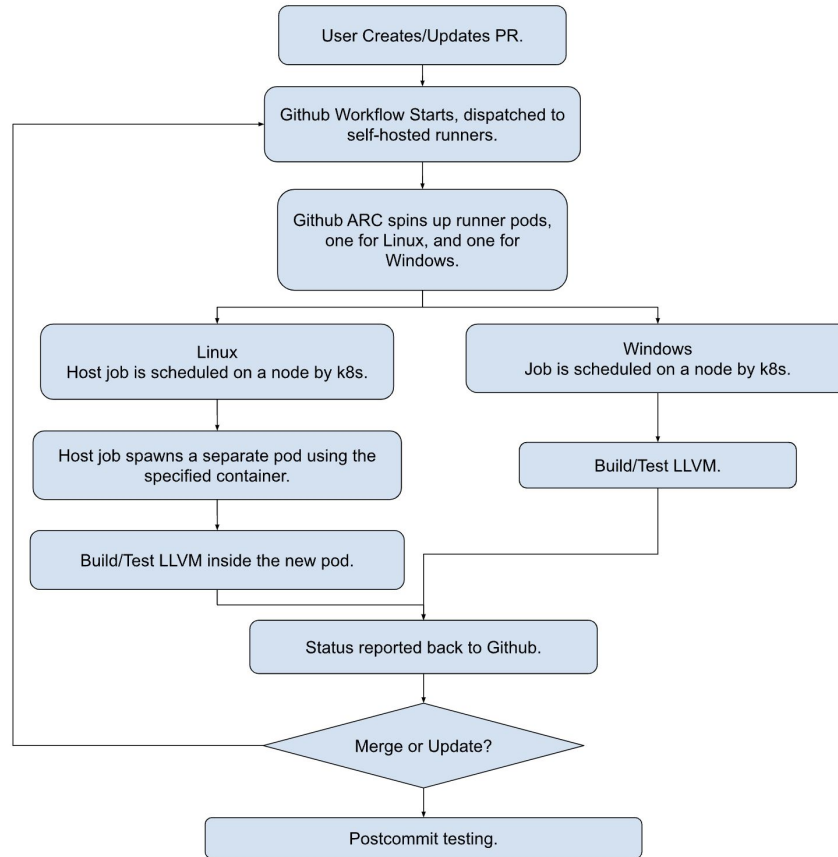


Problems With Current Infrastructure

- No autoscaling, so machines are on all the time, limiting our capacity planning at fixed costs
- No analytics, difficult to assess presubmit performance
- Stability gremlins requiring periodic manual intervention, e.g. reboot of Windows machines
- BuildKite deprecation after the move to GitHub, just adding complexity without adding notable value
- Unclear documentation, separate repository, opaque usability

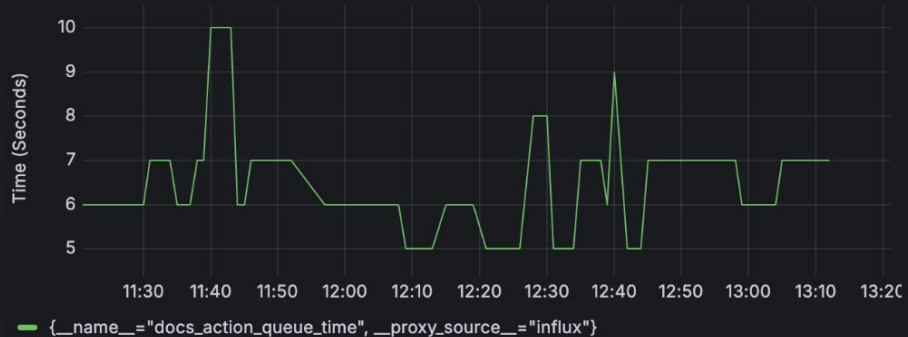
The
Not-Ready-Yet
Thing (But Soon!)

GitHub Runner Process Flow



Analytics on Grafana

Check Code Formatting Queue Time



Check Code Formatting Run Time



Check Code Formatting Success Rate



Last Hour Check Code Formatting Success Rate



Analytics Details

- Grafana public instance hosted on Grafana site
- Could self-host but Grafana is AGPL so Googlers generally can't engage, to ensure we respect the license
- Early prototype needs refinement
- We chose Grafana because of its great documentation and broad usage

Would like to track:

- Queueing delays on Linux and Windows
- Job Success Rates (how to tell infra failures from broken code?)
- Server utilization
- ??

Next Steps

Launch Plan

- Finalize Analytics 1st implementation
- Finish documentation 1st draft
- ~1 week “shadow” launch phase towards end of year
 - Both old and new systems online, fix any issues
 - Commit new premerge code to main LLVM repository
- Turn down old system

Future Plans

- Community discussion about resource allocation, queuing delays, test coverage, trading off against total cost of infrastructure
- More community involvement in infrastructure maintenance and development
- Clear escalation pathways when there issues

Credits

- Aiden Grossman
- Guillaume Chatelet
- Lucile Rose Nihlen
- Nathan Gauër

Emeritus:

- Christian Kühnel
- Mikhail Goncharov

Questions?