Aspect

Achieving the promised 3x-10x Bazel Speedup

DPE Summit 2023

Aspect.build →

Speaker

Alex Eagle

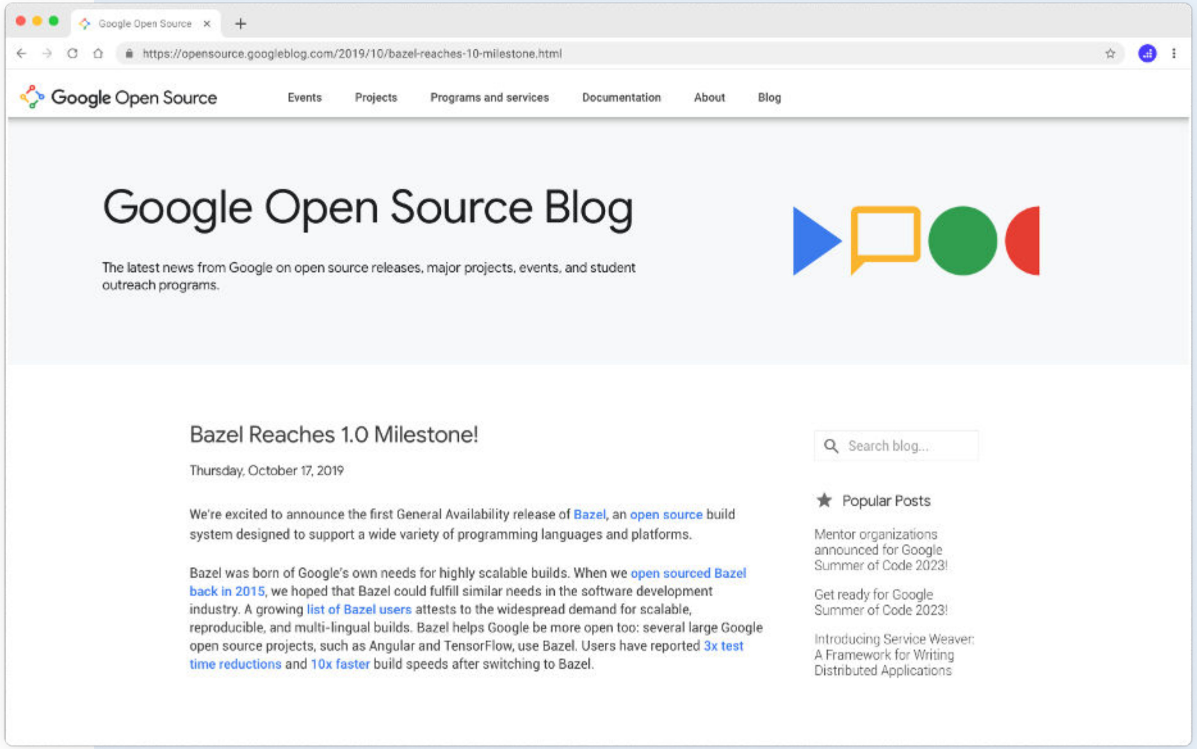
Founder & Co-CEO at Aspect

- Ex-Google 2008-2020
  Tech Lead on Bazel-adjacent systems: CI and Build Results viewer.
- Bazel Community Leader.
- Founded Aspect to bring Bazel's promised benefits to all developers!

Aspect

Bazel

“Users have reported **3x test time reductions** and **10x faster build speeds** after switching to Bazel.”



Google Open Source Blog

The latest news from Google on open source releases, major projects, events, and student outreach programs.

Bazel Reaches 1.0 Milestone!

Thursday, October 17, 2019

We're excited to announce the first General Availability release of Bazel, an open source build system designed to support a wide variety of programming languages and platforms.

Bazel was born of Google's own needs for highly scalable builds. When we open sourced Bazel back in 2015, we hoped that Bazel could fulfill similar needs in the software development industry. A growing list of Bazel users attests to the widespread demand for scalable, reproducible, and multi-lingual builds. Bazel helps Google be more open too: several large Google open source projects, such as Angular and TensorFlow, use Bazel. Users have reported 3x test time reductions and 10x faster build speeds after switching to Bazel.

Popular Posts

Mentor organizations announced for Google Summer of Code 2023!

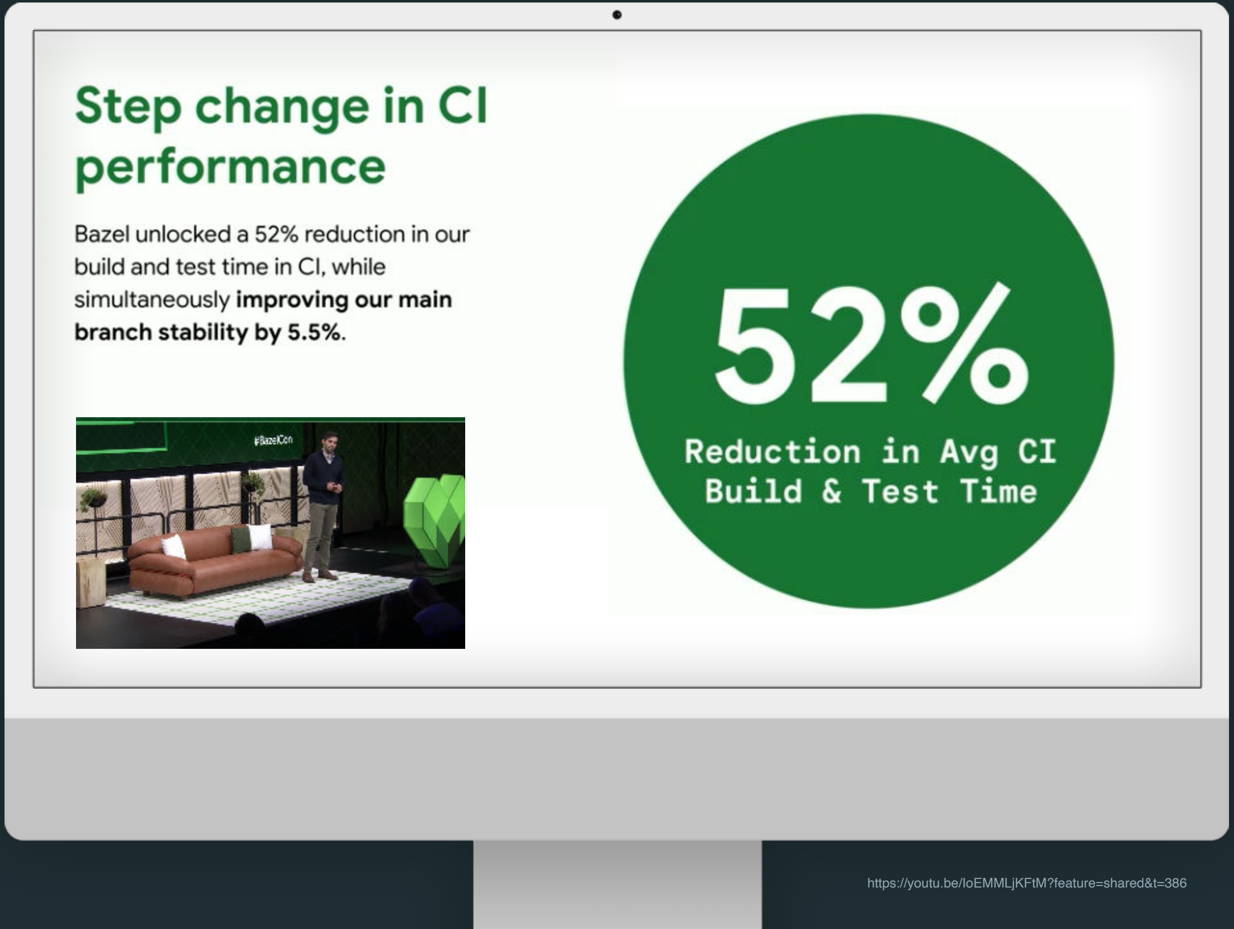
Get ready for Google Summer of Code 2023!

Introducing Service Weaver: A Framework for Writing Distributed Applications

Aspect

Who are these users?

Bazelcon 2022

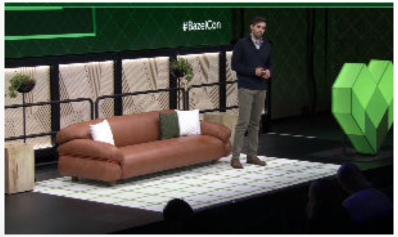


Step change in CI performance

Bazel unlocked a 52% reduction in our build and test time in CI, while simultaneously **improving our main branch stability by 5.5%.**

52%

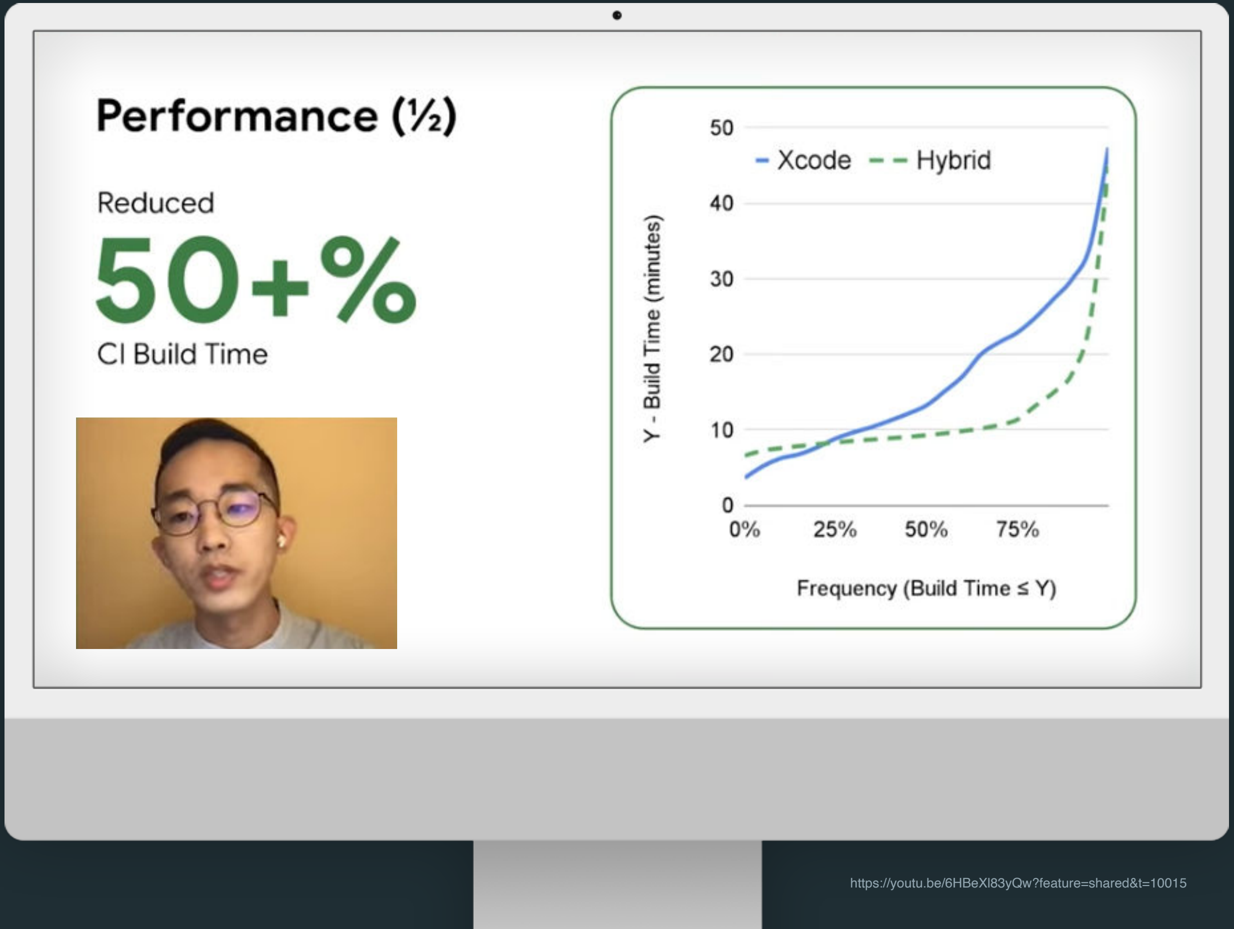
Reduction in Avg CI Build & Test Time



Aspect

Every year at BazelCon …

Bazelcon 2021



Performance (½)

Reduced

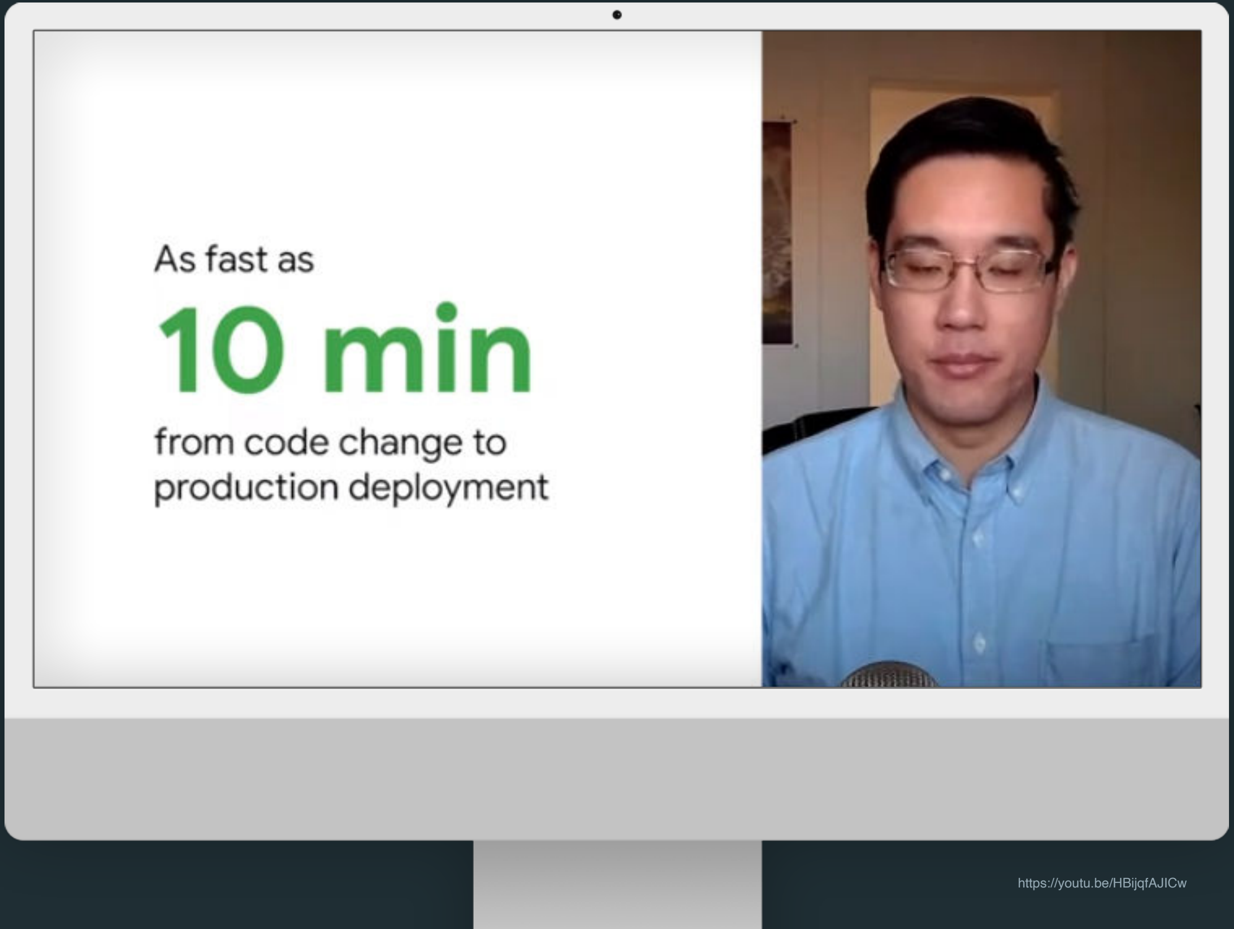
50+%

CI Build Time

https://youtu.be/6HBeXl83yQw?feature=shared&t=10015

… we're reminded it's possible!

Bazelcon 2020



As fast as

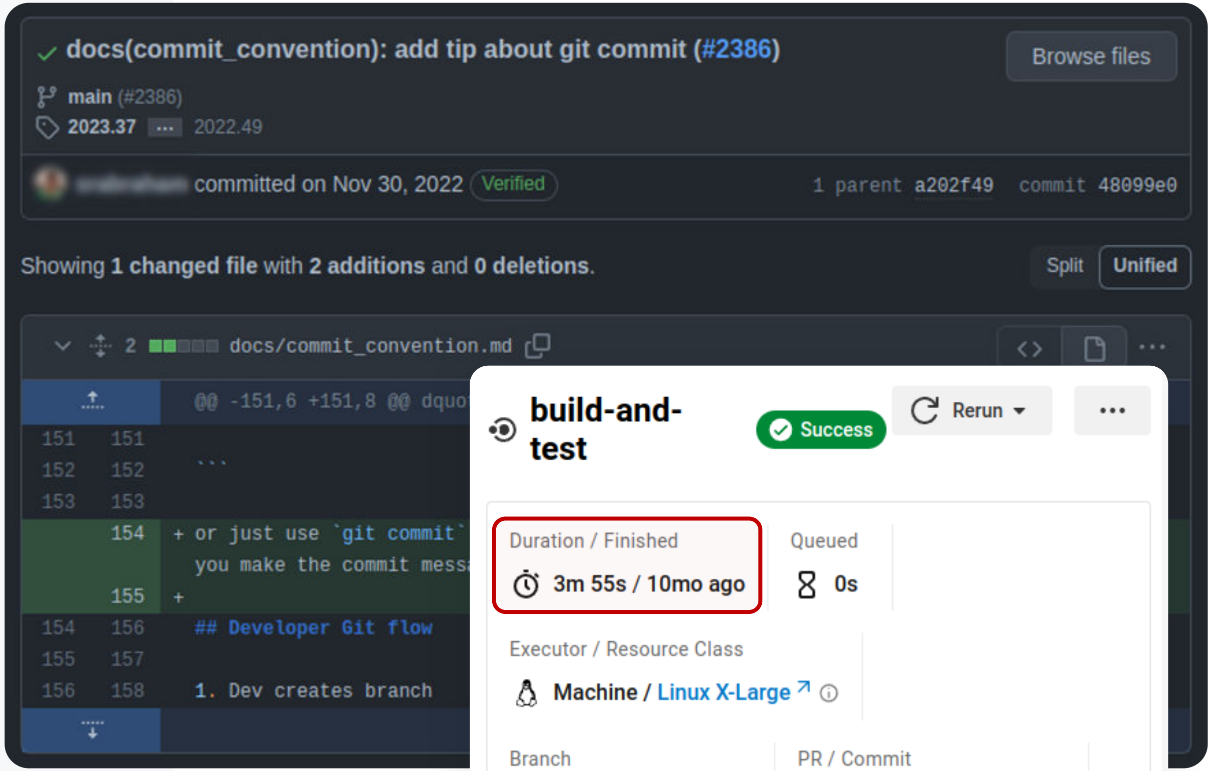
10 min

from code change to production deployment

https://youtu.be/HBijqfAJICw

But how fast are your Bazel builds?

Fastest no-op case?



docs(commit_convention): add tip about git commit (#2386)

Browse files

main (#2386)

2023.37 ... 2022.49

[redacted] committed on Nov 30, 2022 Verified

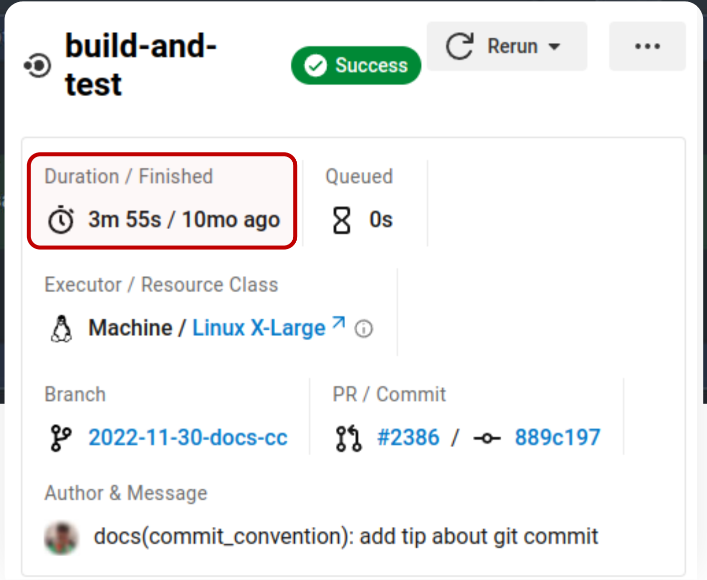
1 parent a202f49 commit 48099e0

Showing 1 changed file with 2 additions and 0 deletions.

Split Unified

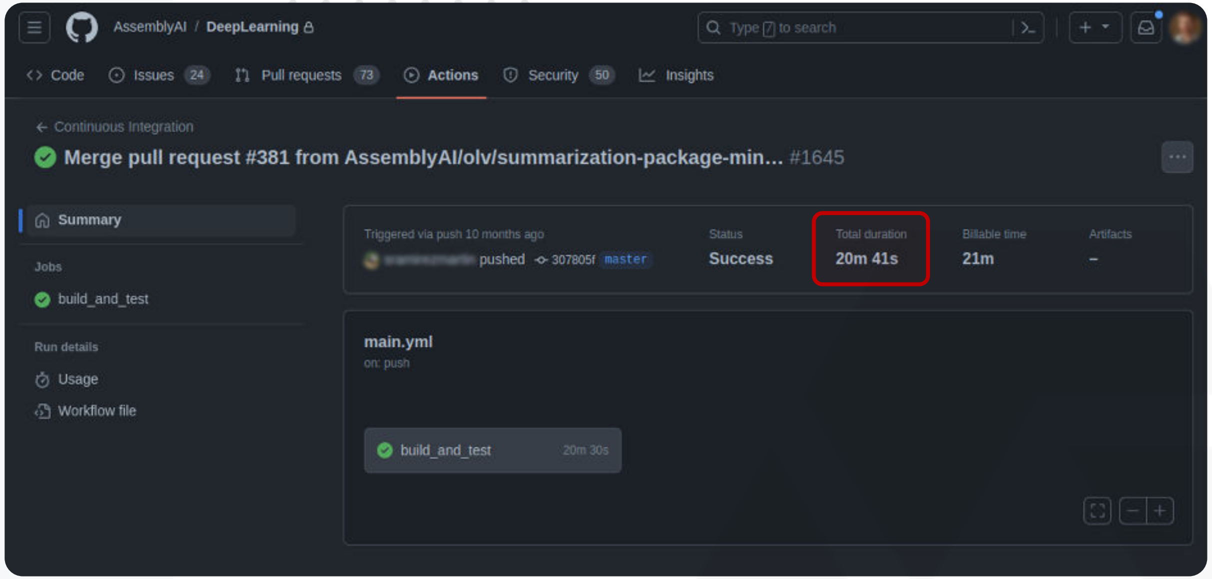
```
2 docs/commit_convention.md

@@ -151,6 +151,8 @@ dquo

 ```

+ or just use `git commit`
  you make the commit mess
+
 ## Developer Git flow

 1. Dev creates branch
```



build-and-test Success

Rerun

| Duration / Finished | Queued |
|---|---|
| 3m 55s / 10mo ago | 0s |

Executor / Resource Class: Machine / Linux X-Large

| Branch | PR / Commit |
|---|---|
| 2022-11-30-docs-cc | #2386 / 889c197 |

Author & Message: docs(commit_convention): add tip about git commit

Aspect
```

But how fast are your Bazel builds?

99%ile slow case?



AssemblyAI / DeepLearning

Code · Issues 24 · Pull requests 73 · Actions · Security 50 · Insights

← Continuous Integration

Merge pull request #381 from AssemblyAI/olv/summarization-package-min... #1645

Summary

Jobs

build_and_test

Run details

Usage

Workflow file

Triggered via push 10 months ago	Status	Total duration	Billable time	Artifacts
[redacted] pushed 307805f master	Success	20m 41s	21m	–

main.yml
on: push

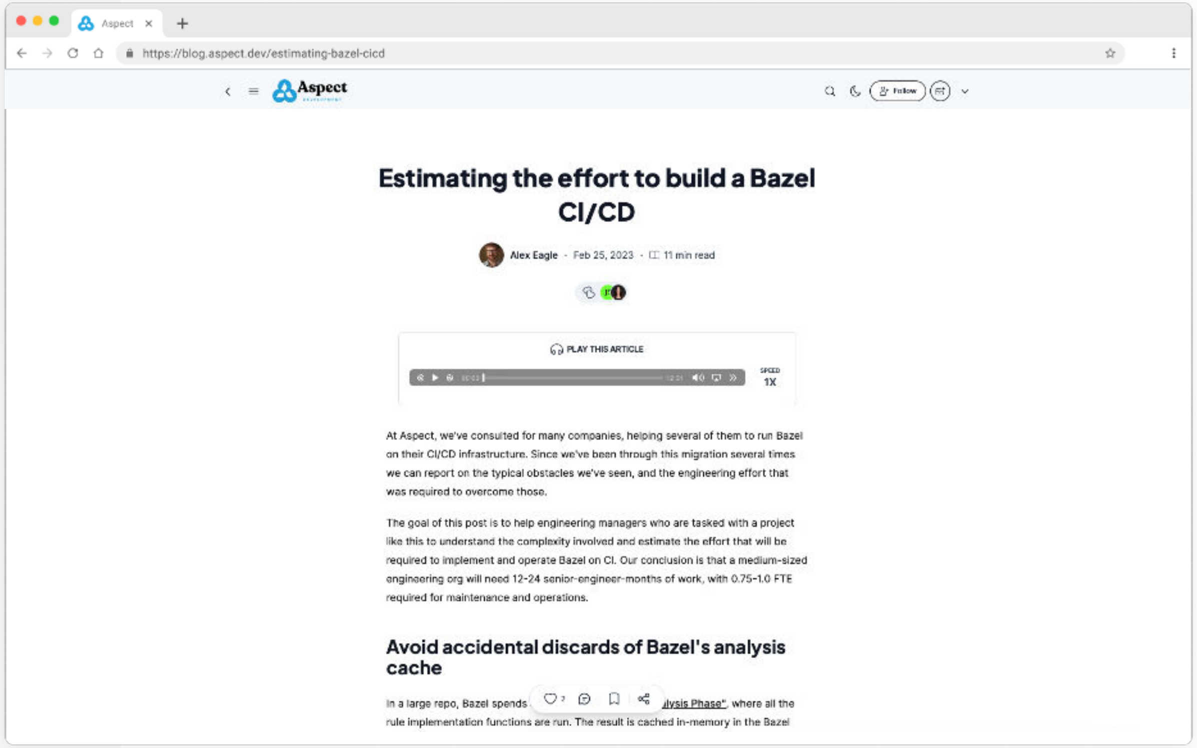
build_and_test 20m 30s

AssemblyAI

Aspect

Estimated effort to build a Bazel CI/CD

We built this for our early consulting clients and
know **how much work** it is
to build it yourself.



Estimating the effort to build a Bazel CI/CD

Alex Eagle · Feb 25, 2023 · 11 min read

PLAY THIS ARTICLE

SPEED 1X

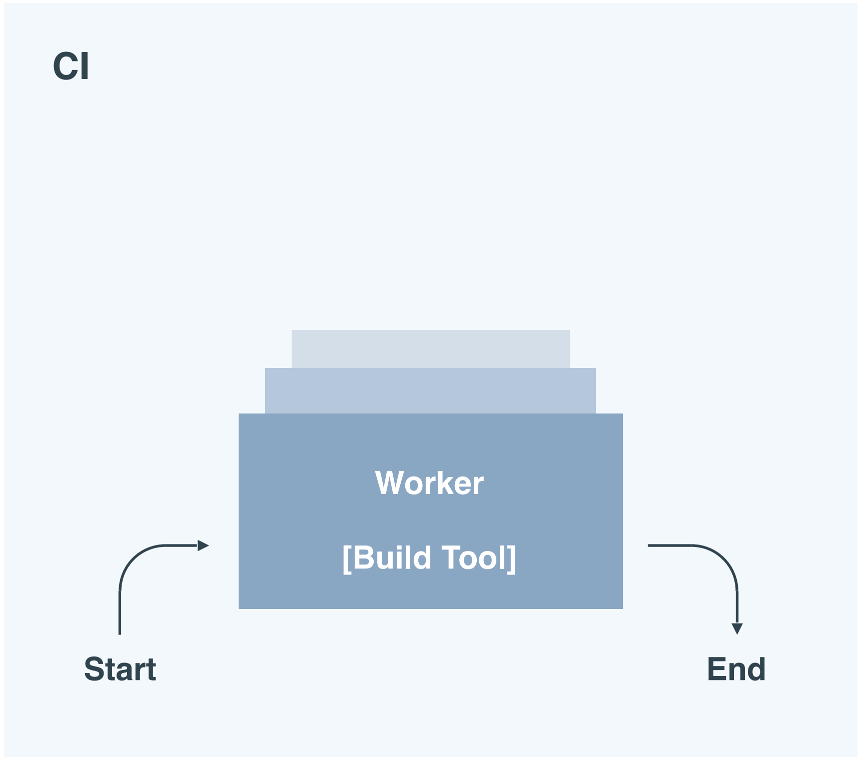
At Aspect, we've consulted for many companies, helping several of them to run Bazel on their CI/CD infrastructure. Since we've been through this migration several times we can report on the typical obstacles we've seen, and the engineering effort that was required to overcome those.

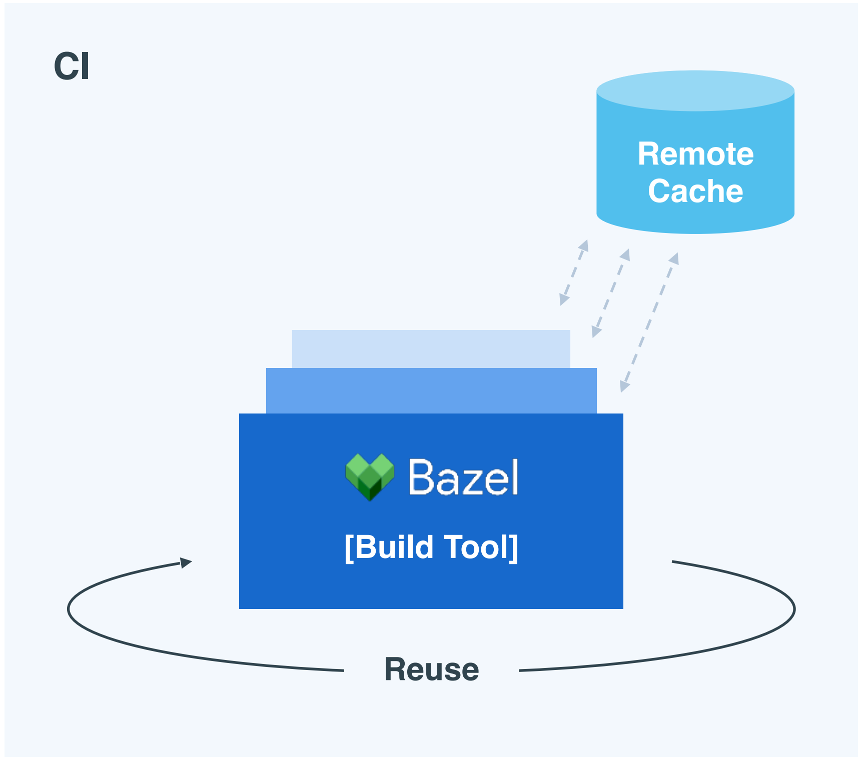
The goal of this post is to help engineering managers who are tasked with a project like this to understand the complexity involved and estimate the effort that will be required to implement and operate Bazel on CI. Our conclusion is that a medium-sized engineering org will need 12-24 senior-engineer-months of work, with 0.75-1.0 FTE required for maintenance and operations.

Avoid accidental discards of Bazel's analysis cache

In a large repo, Bazel spends ... lysis Phase", where all the rule implementation functions are run. The result is cached in-memory in the Bazel

Aspect





Avoid Stale Results

Rely on Bazel Correctness

So, I just need a persistent worker?

Aspect

So many de-optimizations

Analysis phase

Bazel needs Dep Graph

- Restarting Bazel JVM
- Accidental analysis cache discards

So many de-optimizations

Analysis phase

Bazel needs Dep Graph

- Restarting Bazel JVM
- Accidental analysis cache discards

Low cache-hit rate

Doing too much execution

- Non-determinism, due to
  - 3p installs
  - Tools
  - Stamping
- Cache has split-brain

Aspect

So many de-optimizations

Analysis phase

Bazel needs Dep Graph

- Restarting Bazel JVM
- Accidental analysis cache discards

Low cache-hit rate

Doing too much execution

- Non-determinism, due to
  - 3p installs
  - Tools
  - Stamping
- Cache has split-brain

Hosting mistakes

It's the machine

- Spinning disks, network volumes
- No RAID
- Resource leaks

Aspect

So many de-optimizations

Analysis phase

Bazel needs Dep Graph

- Restarting Bazel JVM
- Accidental analysis cache discards

Low cache-hit rate

Doing too much execution

- Non-determinism, due to
  - 3p installs
  - Tools
  - Stamping
- Cache has split-brain

Hosting mistakes

It's the machine

- Spinning disks, network volumes
- No RAID
- Resource leaks

Cluster mistakes

Distributed systems 101

- Checkout causes invalidations
- Not elastic / slow scale-out
- New workers are cold

Aspect

Remote Execution to the rescue?

The “performance optimization of last resort”

Broken

- Not all Bazel rules work when host platform != exec platform
- Much stricter hermeticity requirements

Expensive

- Used to “paper over” too much execution: increases costs by “throwing more machines”.
- Network ingress/egress costs, especially “bad” rules like rules_docker

Alternatives Exist

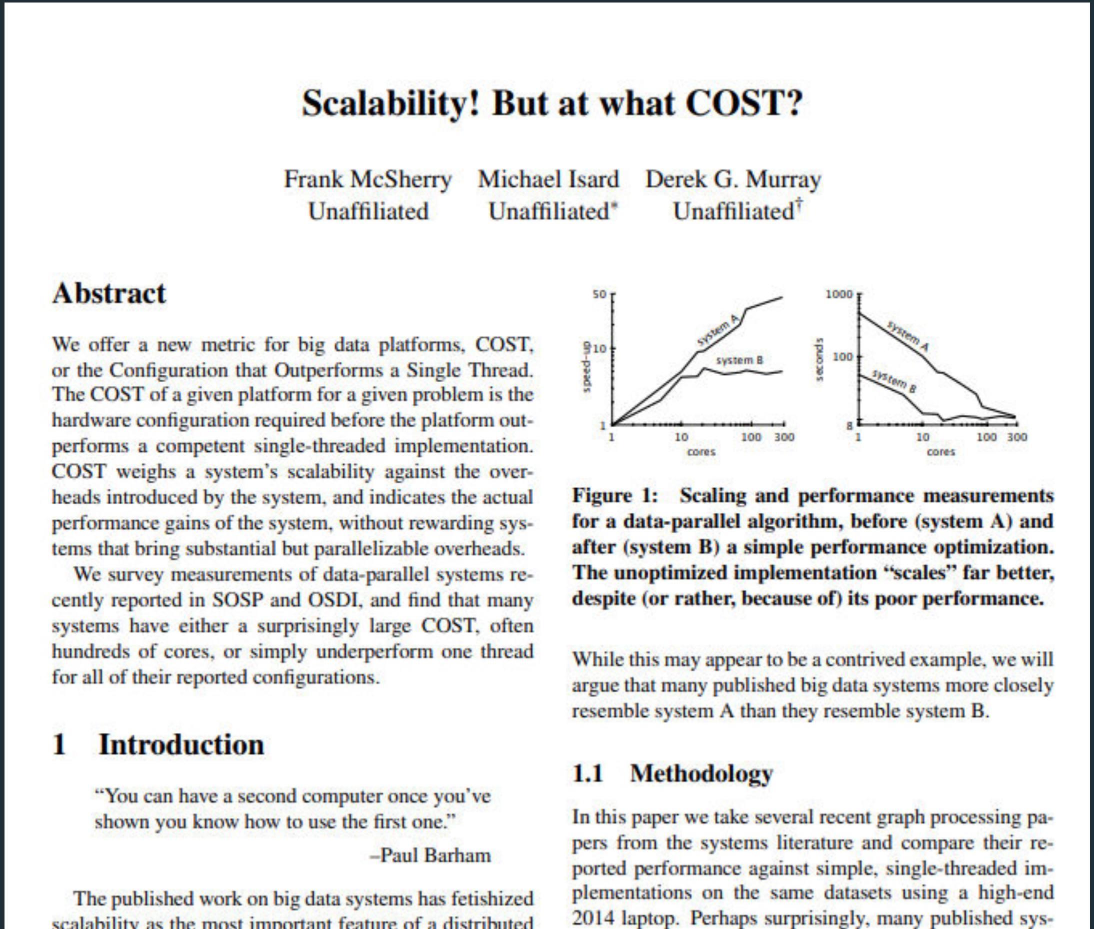
Test Selection → less-frequent triggers

Aspect

You can have a second computer once you've shown you know how to use the first one.

- Paul Barham

Aspect



Scalability! But at what COST?

Frank McSherry Unaffiliated
Michael Isard Unaffiliated*
Derek G. Murray Unaffiliated†

Abstract

We offer a new metric for big data platforms, COST, or the Configuration that Outperforms a Single Thread. The COST of a given platform for a given problem is the hardware configuration required before the platform outperforms a competent single-threaded implementation. COST weighs a system's scalability against the overheads introduced by the system, and indicates the actual performance gains of the system, without rewarding systems that bring substantial but parallelizable overheads.

We survey measurements of data-parallel systems recently reported in SOSP and OSDI, and find that many systems have either a surprisingly large COST, often hundreds of cores, or simply underperform one thread for all of their reported configurations.

1 Introduction

"You can have a second computer once you've shown you know how to use the first one."

–Paul Barham

The published work on big data systems has fetishized scalability as the most important feature of a distributed

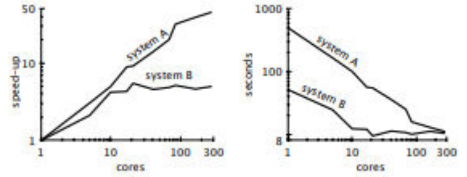


Figure 1: Scaling and performance measurements for a data-parallel algorithm, before (system A) and after (system B) a simple performance optimization. The unoptimized implementation "scales" far better, despite (or rather, because of) its poor performance.

While this may appear to be a contrived example, we will argue that many published big data systems more closely resemble system A than they resemble system B.

1.1 Methodology

In this paper we take several recent graph processing papers from the systems literature and compare their reported performance against simple, single-threaded implementations on the same datasets using a high-end 2014 laptop. Perhaps surprisingly, many published sys-

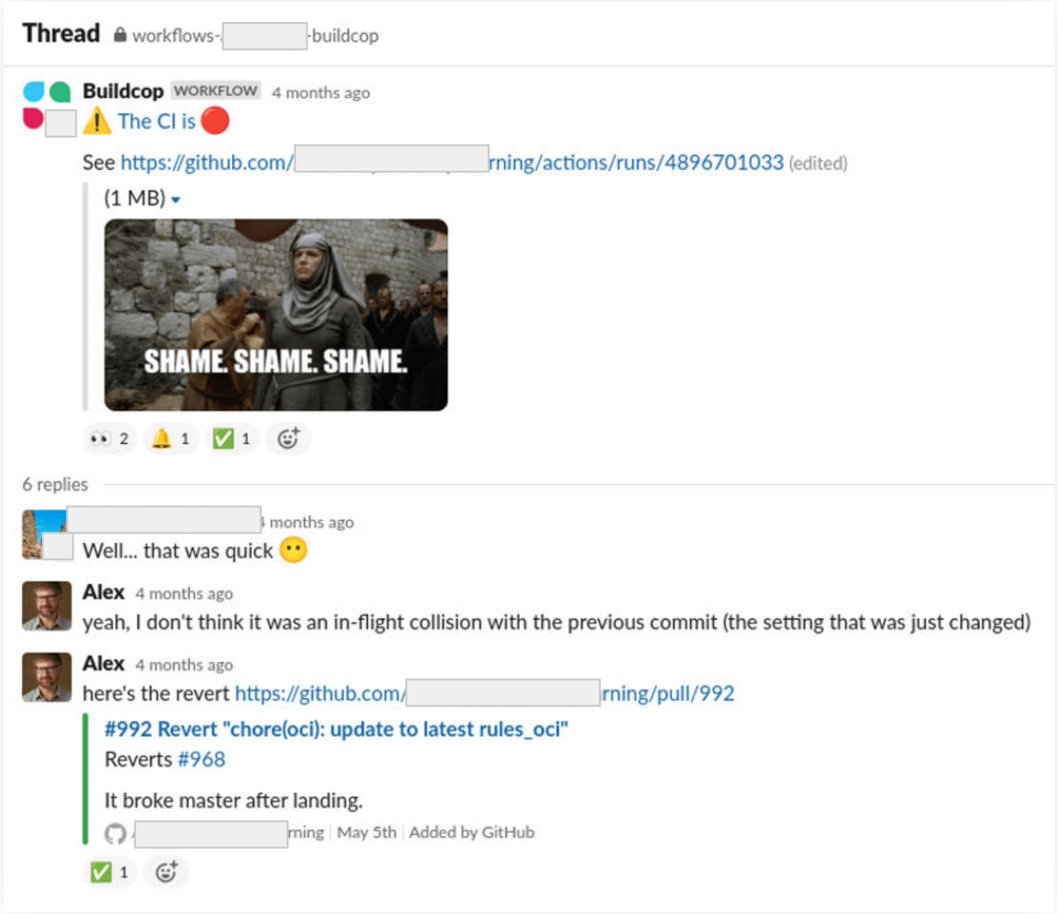
Features

Aspect

Buildcop

Monorepo -> Monobuild

Any developer can break everyone's releases



Thread workflows-[redacted]-buildcop

Buildcop WORKFLOW 4 months ago
⚠️ The CI is 🔴

See https://github.com/[redacted]rning/actions/runs/4896701033 (edited)

(1 MB) ▾

👀 2 🔔 1 ✅ 1

6 replies

[redacted] months ago
Well... that was quick 😶

Alex 4 months ago
yeah, I don't think it was an in-flight collision with the previous commit (the setting that was just changed)

Alex 4 months ago
here's the revert https://github.com/[redacted]rning/pull/992

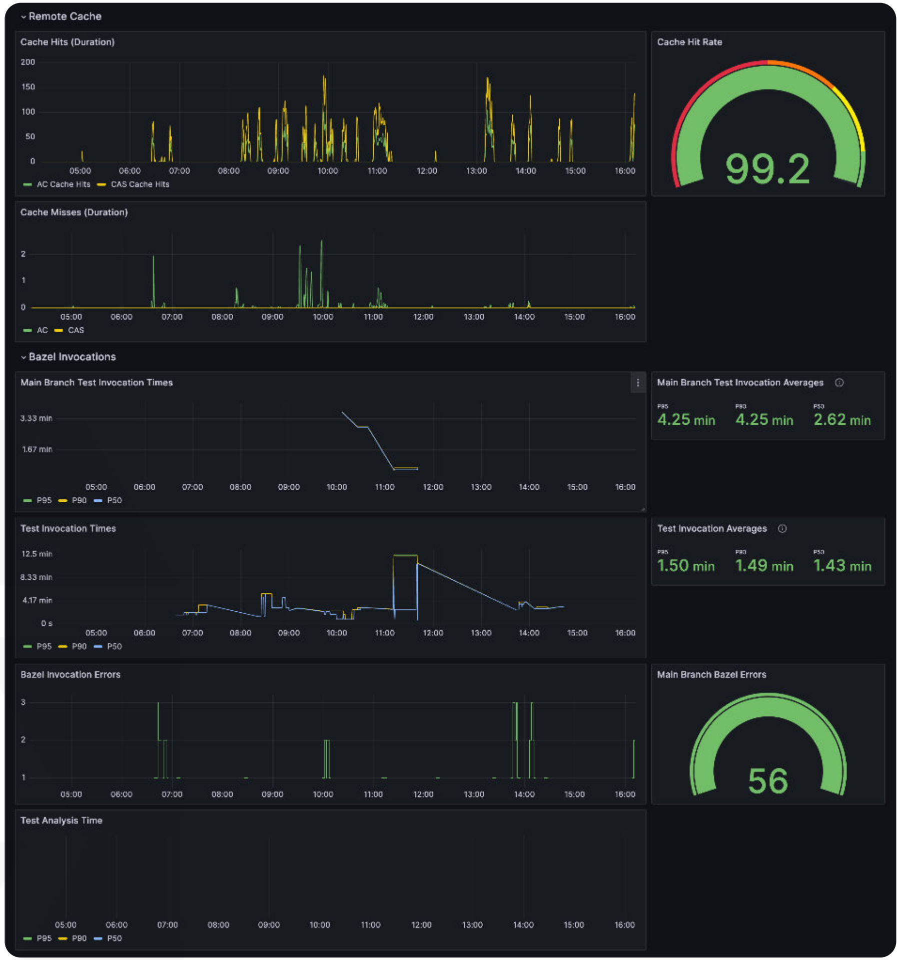
#992 Revert "chore(oci): update to latest rules_oci"
Reverts #968

It broke master after landing.

[redacted]rning | May 5th | Added by GitHub

✅ 1

Metrics

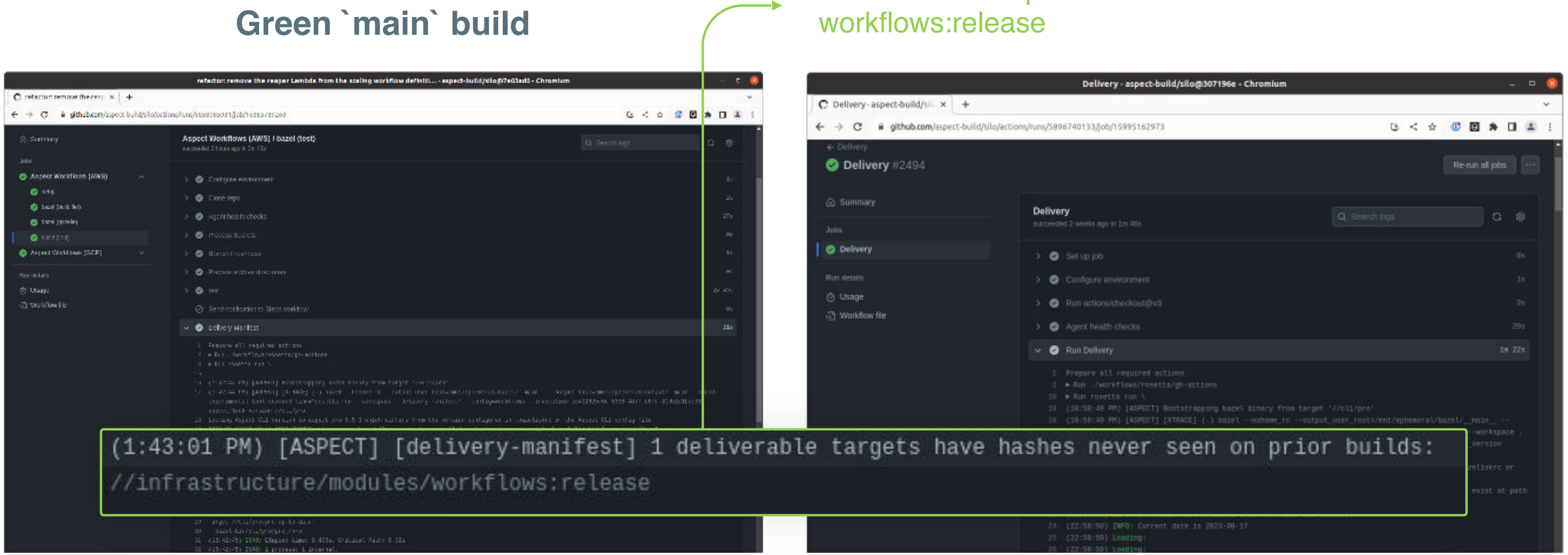


Aspect

Selective Continuous Delivery

Green `main` build

bazel run --stamp //infrastructure/modules/workflows:release



refactor: remove the reaper Lambda from the scaling workflow definiti... - aspect-build/silo@7e03ad0 - Chromium

Aspect Workflows (AWS) / bazel (test)

Delivery Manifest

```
Prepare all required actions
▶ Run ./workflows/rosetta/gh-actions
▶ Run rosetta run \
```

Delivery - aspect-build/silo@307196e - Chromium

github.com/aspect-build/silo/actions/runs/5896740133/job/15995162973

Delivery #2494

Delivery
succeeded 2 weeks ago in 1m 46s

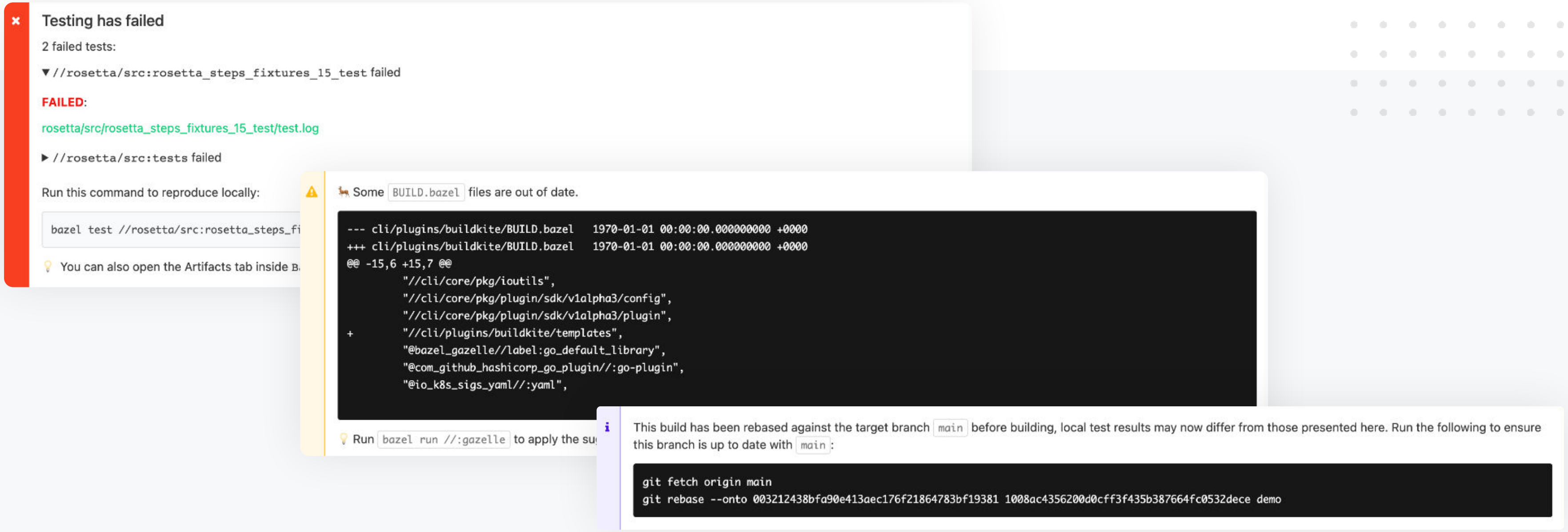
Set up job 0s
Configure environment 1s
Run actions/checkout@v3 3s
Agent health checks 29s
Run Delivery 1m 22s

```
Prepare all required actions
▶ Run ./workflows/rosetta/gh-actions
▶ Run rosetta run \
(18:58:49 PM) [ASPECT] Bootstrapping bazel binary from target '//cli/pro'
(18:58:49 PM) [ASPECT] [XTRACE] (.) bazel --nohome_rc --output_user_root=/mnt/ephemeral/bazel/__main__ --
(22:58:50) INFO: Current date is 2023-08-17
(22:58:50) Loading:
(22:58:50) Loading:
```

```
(1:43:01 PM) [ASPECT] [delivery-manifest] 1 deliverable targets have hashes never seen on prior builds:
//infrastructure/modules/workflows:release
```

Aspect

Developer Experience



Testing has failed

2 failed tests:

▼ `//rosetta/src:rosetta_steps_fixtures_15_test` failed

FAILED:

rosetta/src/rosetta_steps_fixtures_15_test/test.log

▶ `//rosetta/src:tests` failed

Run this command to reproduce locally:

```
bazel test //rosetta/src:rosetta_steps_fi
```

You can also open the Artifacts tab inside B

Some `BUILD.bazel` files are out of date.

```
--- cli/plugins/buildkite/BUILD.bazel   1970-01-01 00:00:00.000000000 +0000
+++ cli/plugins/buildkite/BUILD.bazel   1970-01-01 00:00:00.000000000 +0000
@@ -15,6 +15,7 @@
         "//cli/core/pkg/ioutils",
         "//cli/core/pkg/plugin/sdk/v1alpha3/config",
         "//cli/core/pkg/plugin/sdk/v1alpha3/plugin",
+        "//cli/plugins/buildkite/templates",
         "@bazel_gazelle//label:go_default_library",
         "@com_github_hashicorp_go_plugin//:go-plugin",
         "@io_k8s_sigs_yaml//:yaml",
```

Run `bazel run //:gazelle` to apply the su

This build has been rebased against the target branch `main` before building, local test results may now differ from those presented here. Run the following to ensure this branch is up to date with `main`:

```
git fetch origin main
git rebase --onto 003212438bfa90e413aec176f21864783bf19381 1008ac4356200d0cff3f435b387664fc0532dece demo
```

Aspect

Skip it: Aspect Workflows

aspect.build/workflows

Runs on your cloud, on your existing CI

No migrations required.

Co-managed, we carry the pager

Avoid the perverse incentive of **paying more** for **slower** builds. No migrations required.

Free one-month trial

We'll prove it's blazing fast AND pays for itself!

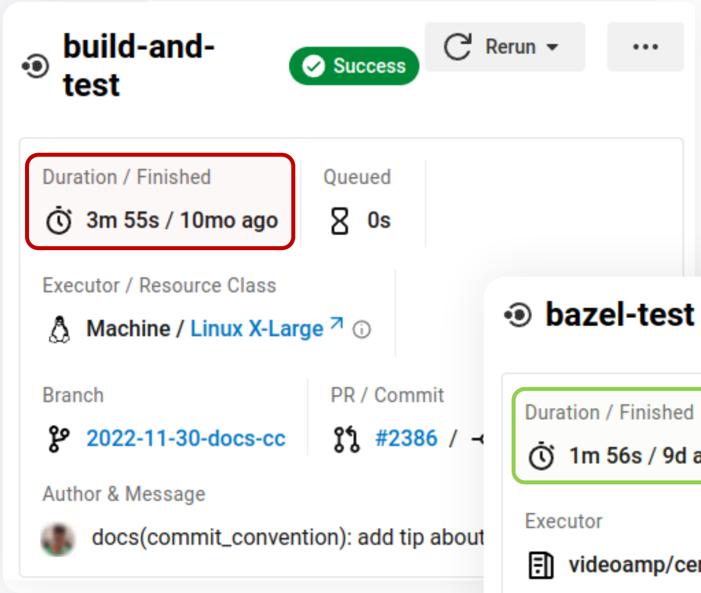
Aspect

Fast no-op

Reduced

2 minutes

of overhead



build-and-test Success Rerun

Duration / Finished
3m 55s / 10mo ago

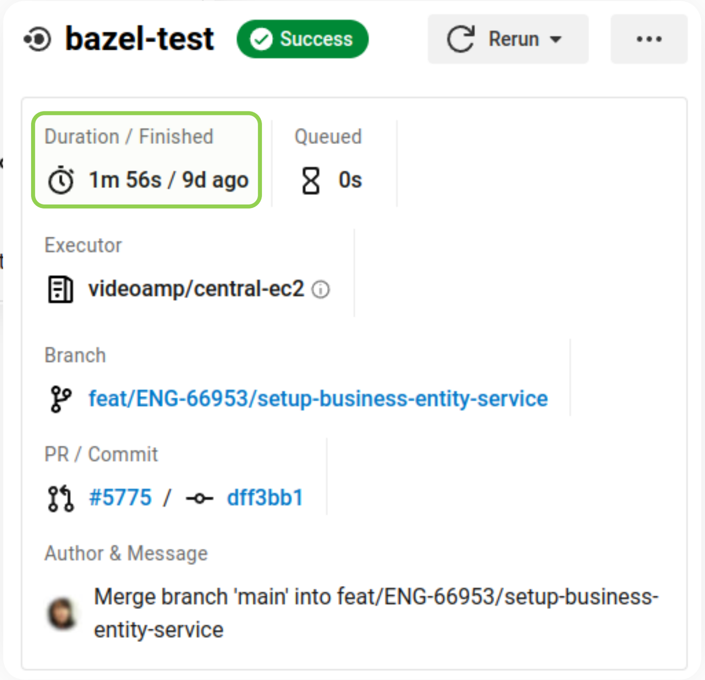
Queued
0s

Executor / Resource Class
Machine / Linux X-Large

Branch
2022-11-30-docs-cc

PR / Commit
#2386 /

Author & Message
docs(commit_convention): add tip about



bazel-test Success Rerun

Duration / Finished
1m 56s / 9d ago

Queued
0s

Executor
videoamp/central-ec2

Branch
feat/ENG-66953/setup-business-entity-service

PR / Commit
#5775 / dff3bb1

Author & Message
Merge branch 'main' into feat/ENG-66953/setup-business-entity-service

Aspect

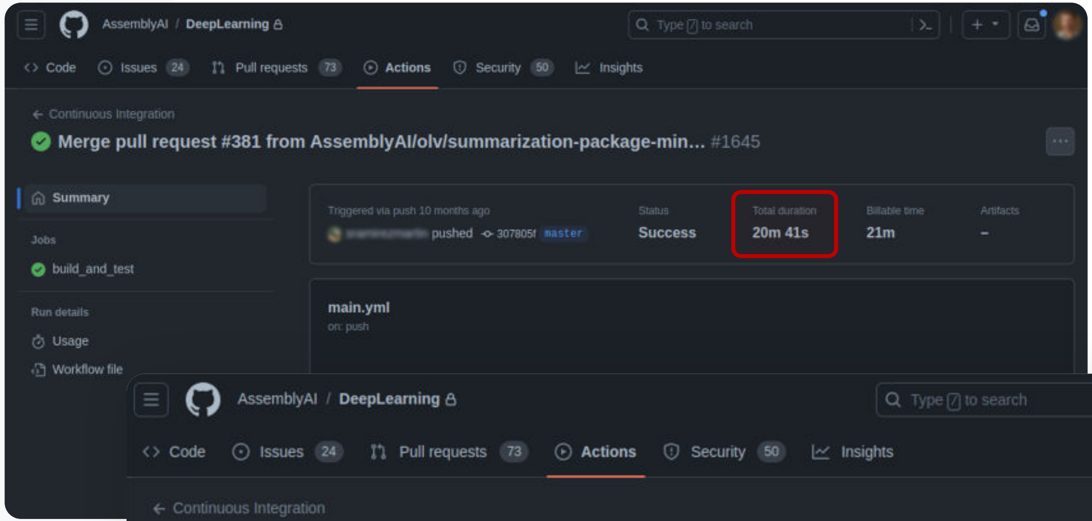
Fast 99%ile

AssemblyAI

About

2x

speedup



AssemblyAI / DeepLearning

Code | Issues 24 | Pull requests 73 | Actions | Security 50 | Insights

← Continuous Integration

Merge pull request #381 from AssemblyAI/olv/summarization-package-min… #1645

Summary

Jobs

build_and_test

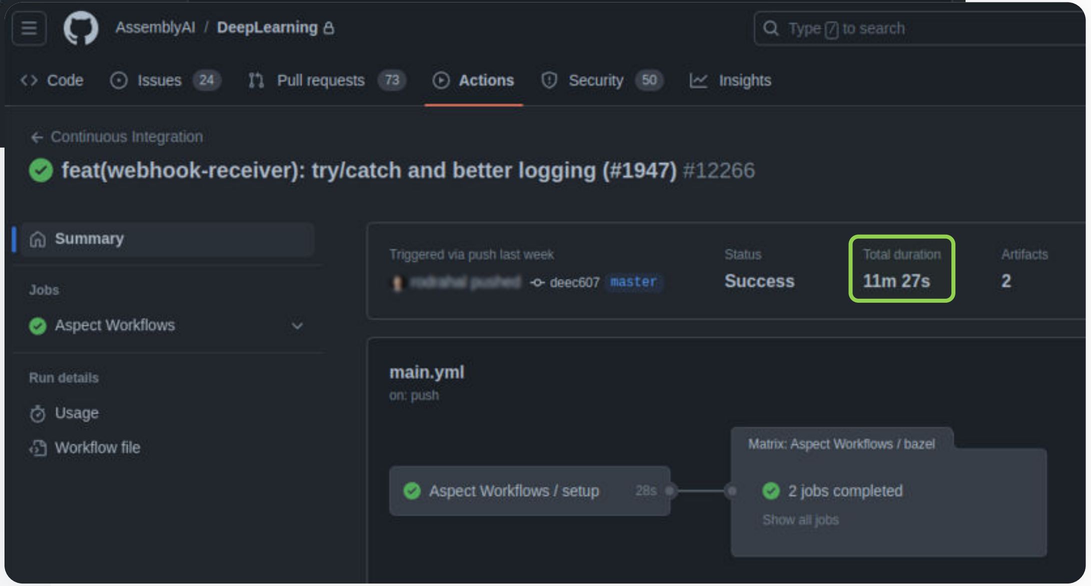
Run details

Usage

Workflow file

Triggered via push 10 months ago	Status	Total duration	Billable time	Artifacts
[redacted] pushed -o- 307805f master	Success	20m 41s	21m	–

main.yml
on: push



AssemblyAI / DeepLearning

Code | Issues 24 | Pull requests 73 | Actions | Security 50 | Insights

← Continuous Integration

feat(webhook-receiver): try/catch and better logging (#1947) #12266

Summary

Jobs

Aspect Workflows

Run details

Usage

Workflow file

Triggered via push last week	Status	Total duration	Artifacts
[redacted] -o- deec607 master	Success	11m 27s	2

main.yml
on: push

Aspect Workflows / setup 28s

Matrix: Aspect Workflows / bazel

2 jobs completed

Show all jobs

Aspect

Customer Case Study

coda

Challenge

Bazel build&test on CircleCI taking over 10min avg, often over 30min.

CI Cost too high: 2 SWE equiv.

Aspect

Solution

- Aspect Workflows trial for one month.
- Improve stability and uptime, evaluate spot instances.
- “Pretty smooth transition” for developers.
- Regular improvement through monthly releases.

Results

10x
faster no-op build
(from 11 min to 1 min)

2-3x
Speedup of typical build & test

67%
Reduced compute $
(despite higher usage)

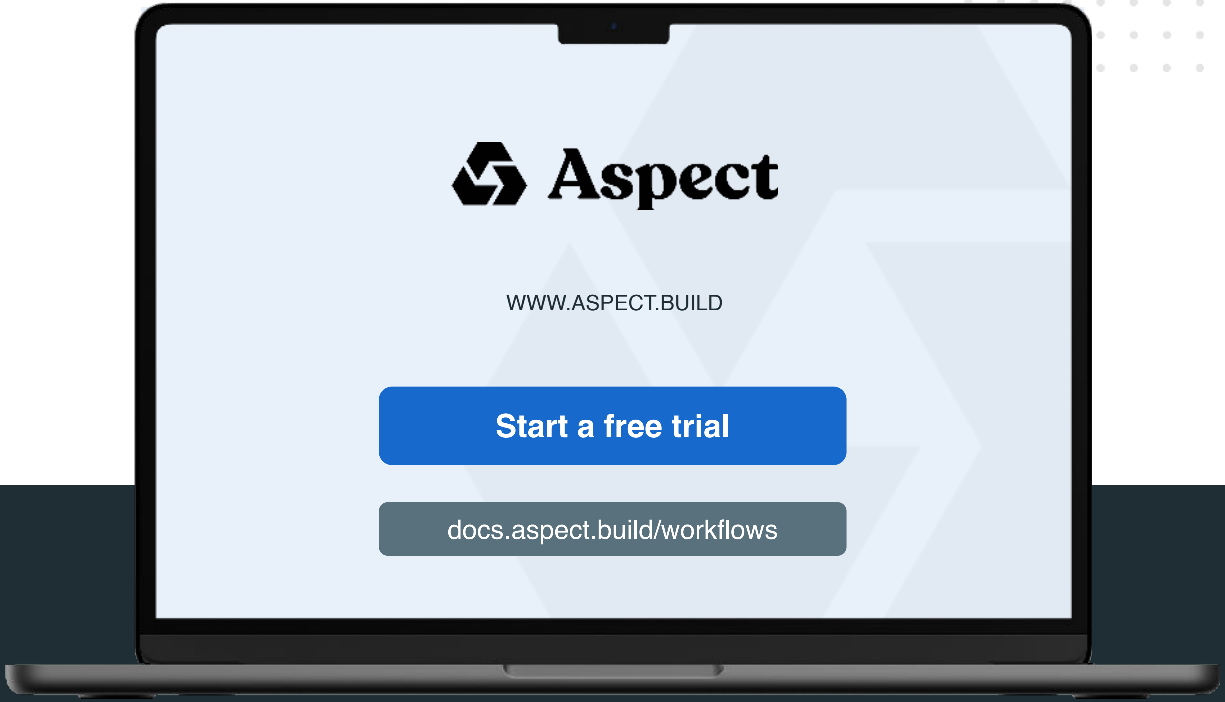
” We went from having significant limits in CI and tools to where the **limits are now just due to our code**.

Aspect

Thank you for listening!

Next Steps

- See it in action on our OSS repositories
- Book a demo



Aspect

WWW.ASPECT.BUILD

Start a free trial

docs.aspect.build/workflows

Aspect