Developer Productivity Engineering for Open Source Communities

Etienne Studer, SVP of Engineering, Develocity

Gradle

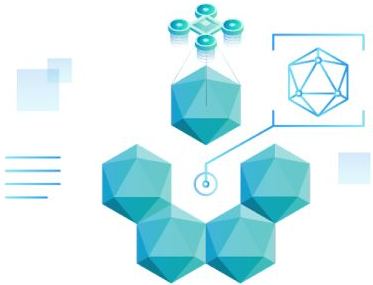
Deeply rooted in the open-source world.

- First release of the Gradle Build Tool in 2009
- More than 30 million downloads/month
- Open sourced under the Apache License 2.0
- Default build tool for Google Android and Spring Boot applications
- Leader in the build tool used by new GitHub projects

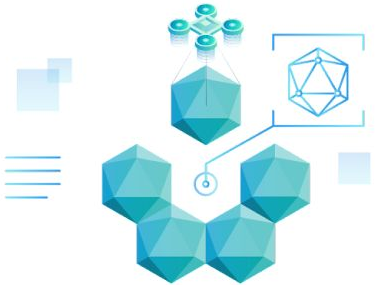
Gradle
BUILD TOOL

Community over Code 2023 – Gold Sponsor

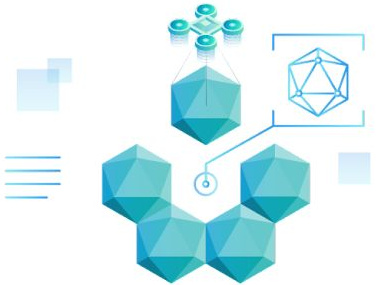
ASF – Targeted Platinum Sponsor



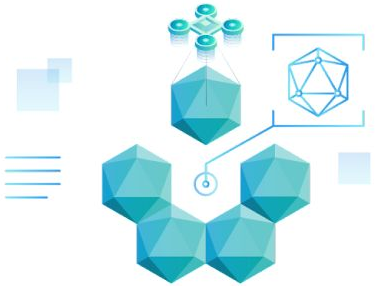
Time is precious.



Making optimal use of our time is crucial to our impact and happiness.

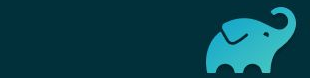


Developers are most productive and happy in the state of “flow”, creating high-value, high-quality features at a fast pace.

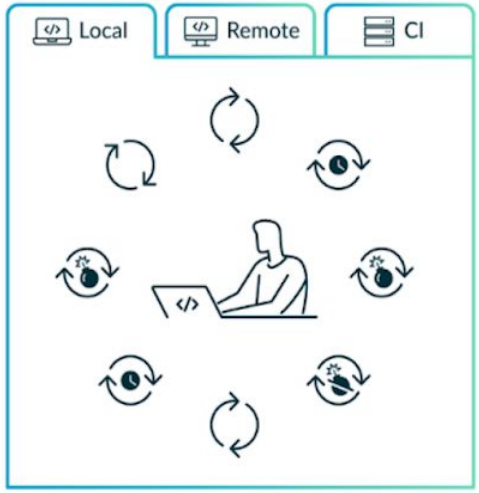


Oftentimes, developing software comes with a lot of waiting, context-switching, inefficient troubleshooting, googling, and many gut-feel decisions.

Toolchain behavior is a major contributor and inhibitor to developer productivity



High Friction in Developer Productivity



- This takes too long!
- This takes too long to fix
- This should have been observable

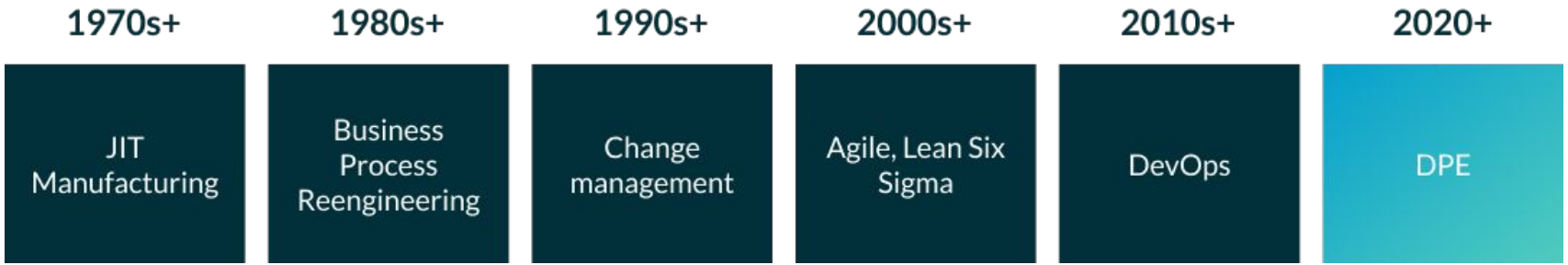
High Friction in Developer Productivity

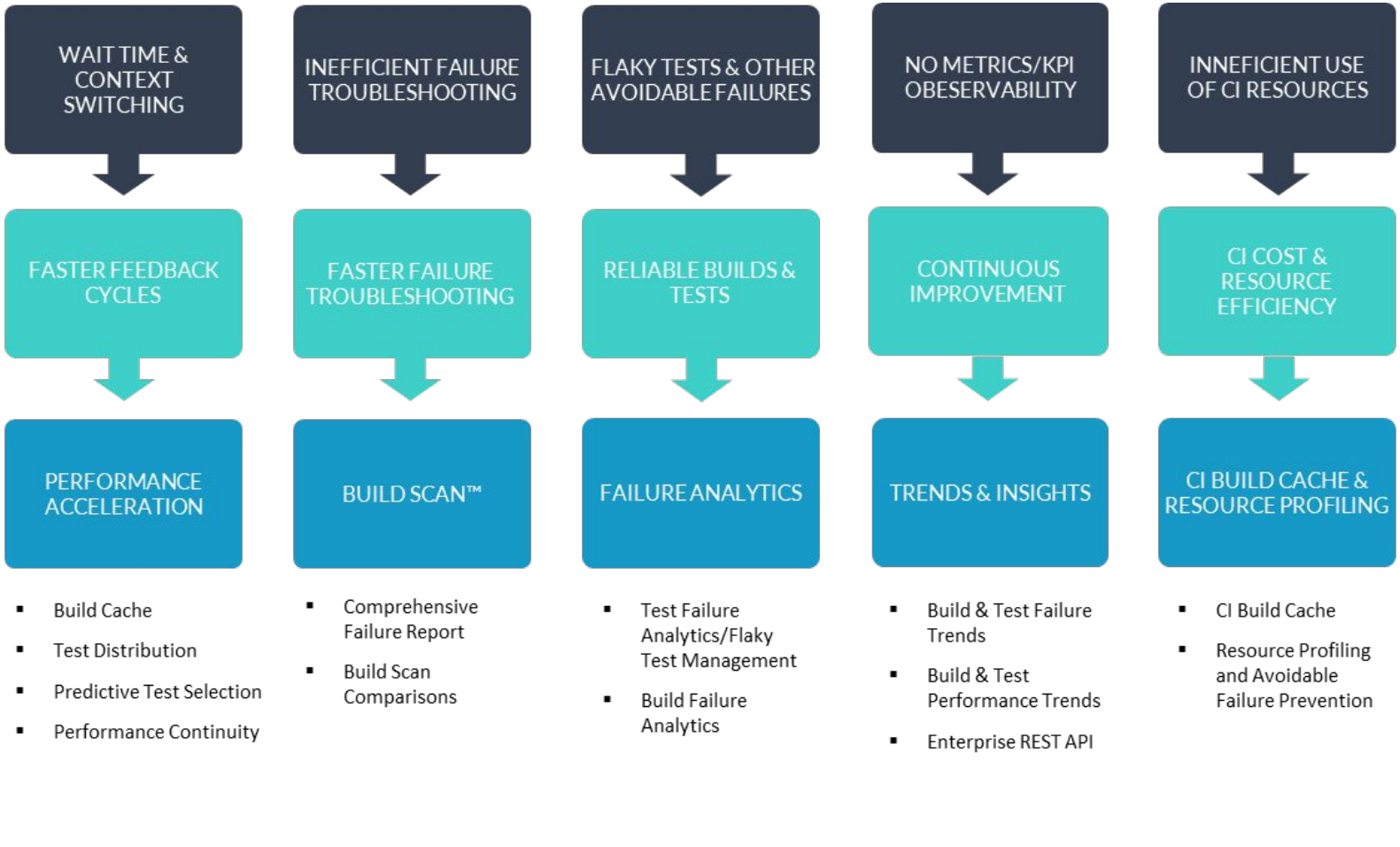
- **Waiting on builds and tests is still a problem.** Despite industry-wide investments in DevOps, developers still say the most time-consuming thing they're doing at work besides writing code is waiting on builds and tests.

GitHub Survey 2023

Developer Productivity Engineering (DPE)

DPE is a new software development practice used by leading software development organizations to maximize developer productivity and happiness.


1970s+
1980s+
1990s+
2000s+
2010s+
2020+
JIT Manufacturing
Business Process Reengineering
Change management
Agile, Lean Six Sigma
DevOps
DPE

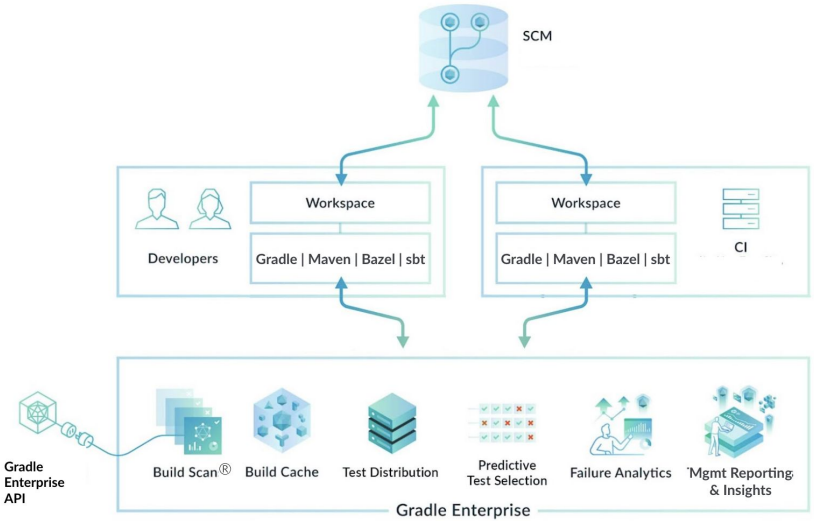

WAIT TIME & CONTEXT SWITCHING
INEFFICIENT FAILURE TROUBLESHOOTING
FLAKY TESTS & OTHER AVOIDABLE FAILURES
NO METRICS/KPI OBESERVABILITY
INNEFICIENT USE OF CI RESOURCES
FASTER FEEDBACK CYCLES
FASTER FAILURE TROUBLESHOOTING
RELIABLE BUILDS & TESTS
CONTINUOUS IMPROVEMENT
CI COST & RESOURCE EFFICIENCY
PERFORMANCE ACCELERATION
BUILD SCAN™
FAILURE ANALYTICS
TRENDS & INSIGHTS
CI BUILD CACHE & RESOURCE PROFILING
Build Cache
Test Distribution
Predictive Test Selection
Performance Continuity
Comprehensive Failure Report
Build Scan Comparisons
Test Failure Analytics/Flaky Test Management
Build Failure Analytics
Build & Test Failure Trends
Build & Test Performance Trends
Enterprise REST API
CI Build Cache
Resource Profiling and Avoidable Failure Prevention

Develocity

The Develocity data platform and acceleration engine is available for free to all Apache projects at **ge.apache.org**.

Develocity

Develocity is a platform that improves the productivity of engineering teams by providing deep insights into how their software is built locally and on CI and by highly accelerating the execution of their Maven, Gradle, Android, Bazel, sbt builds.

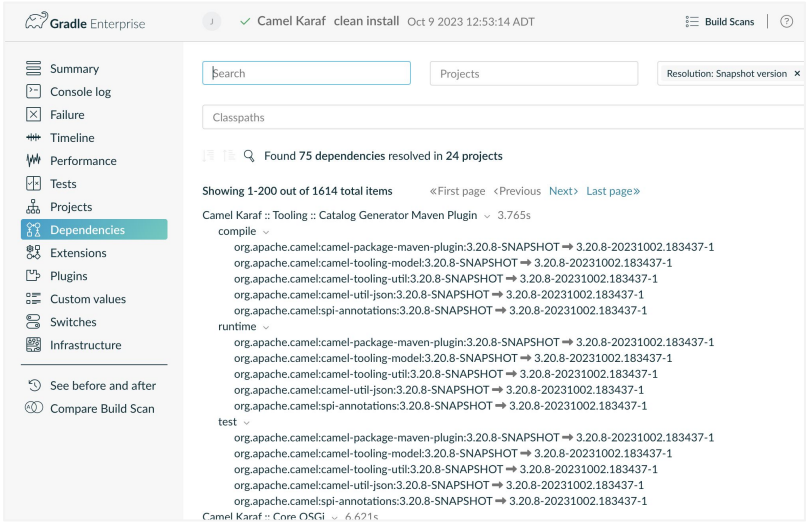


Build Insights

Build Insights via Build Scan

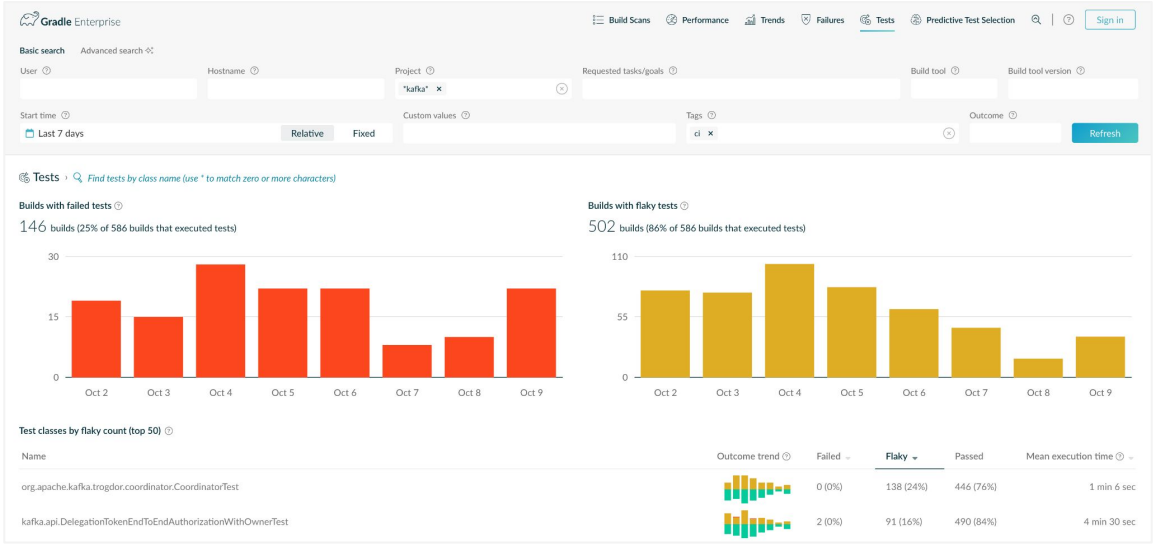
Build insights allow efficient troubleshooting and informed decisions about what to improve to make builds faster and more reliable.

- Resolved dependency graph
- Executed goals/tasks
- Executed tests
- Applied plugins
- Switches/flags
- Environment
- Log
- etc.

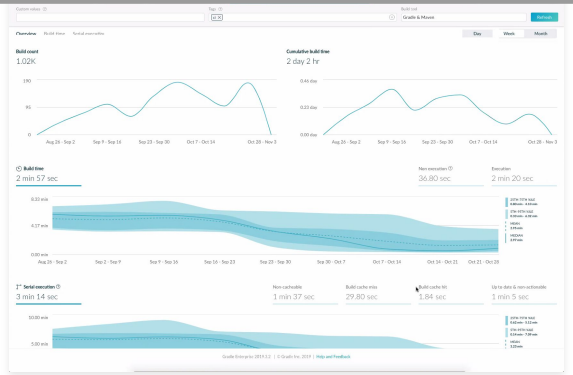


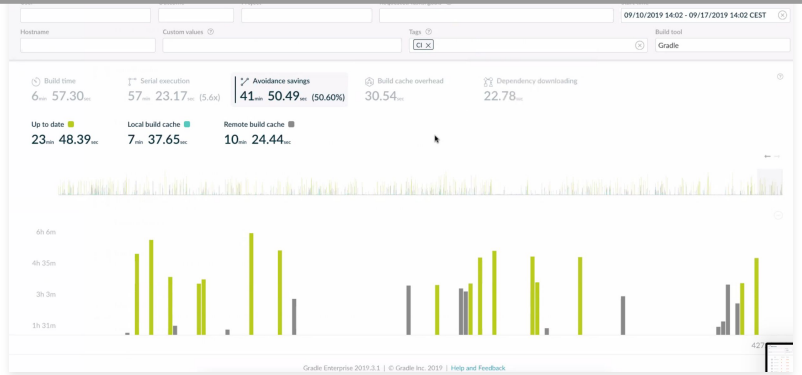
Build Insights on Flaky Tests

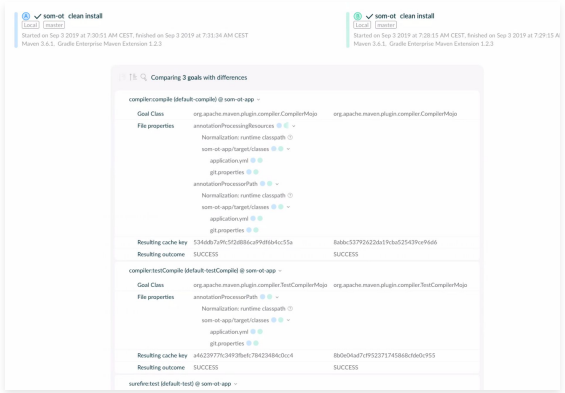
Flaky tests are present in every project and a major contributor to unreliable builds, causing wasteful build and test cycles and compute resources.

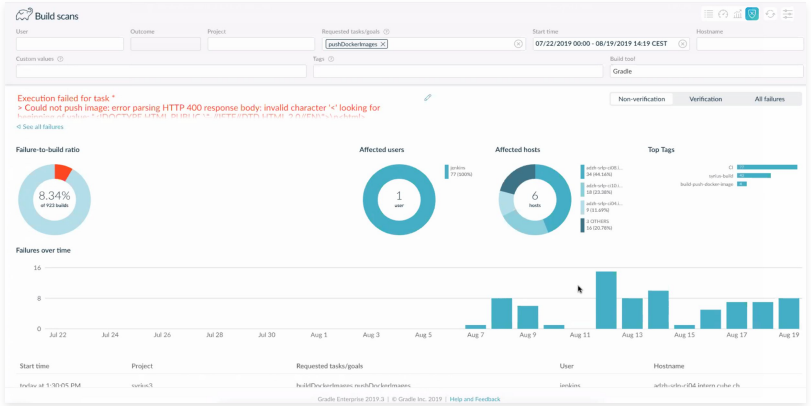


Build Insights Cross-build Analysis

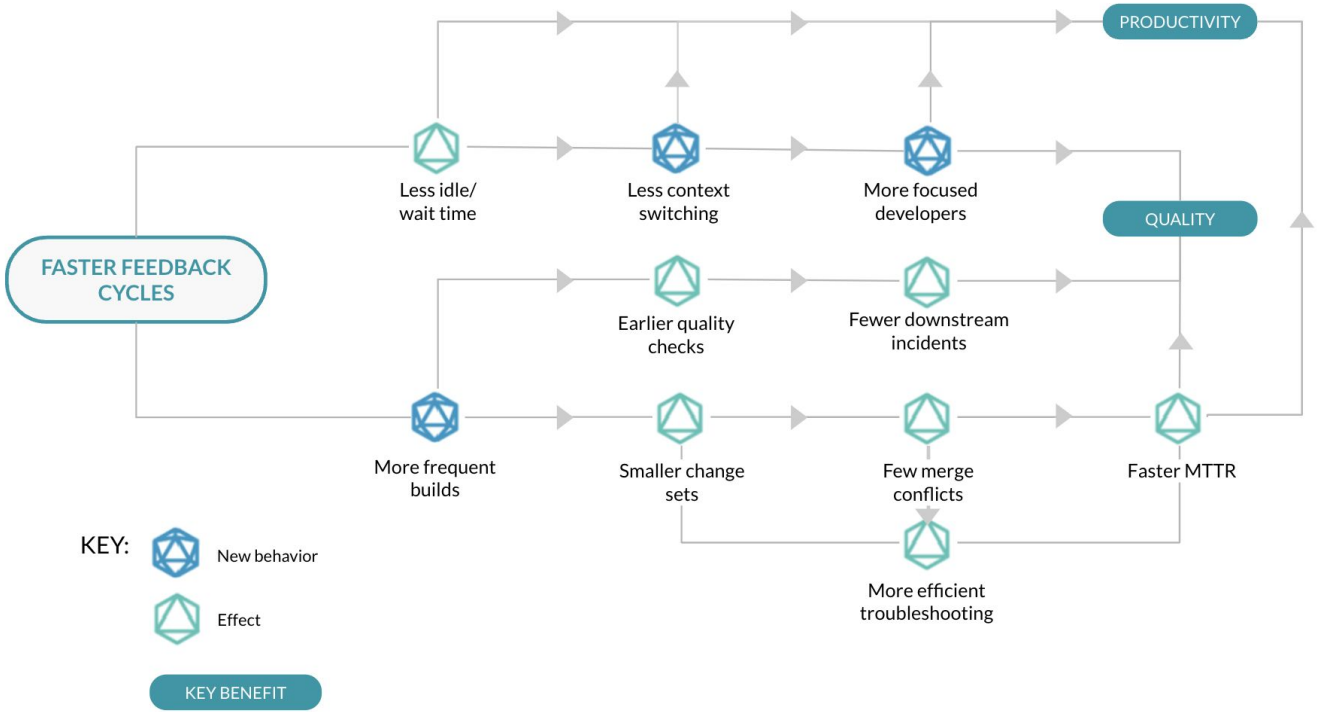








Build Acceleration


PRODUCTIVITY
Less idle/
wait time
Less context
switching
More focused
developers
QUALITY
FASTER FEEDBACK
CYCLES
Earlier quality
checks
Fewer downstream
incidents
More frequent
builds
Smaller change
sets
Few merge
conflicts
Faster MTTR
More efficient
troubleshooting
KEY:
New behavior
Effect
KEY BENEFIT

Acceleration Technologies

Build cache stores and provides artifacts produced by previous task/goal invocations to avoid work that has been done before.

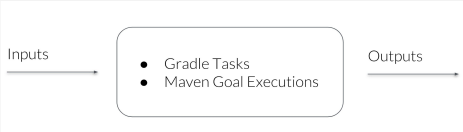
Local build cache node and geo-distributed remote build cache nodes are available for best performance.

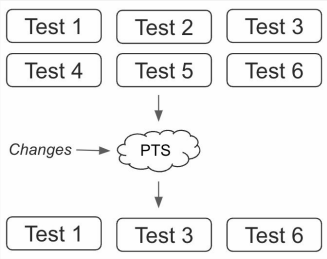
Predictive test selection runs only those tests likely to change their state for a given set of changes based on historical data and ML.

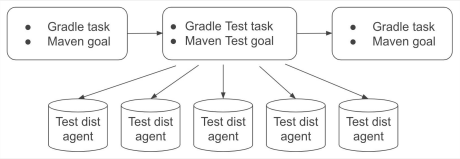
Simulator functionality is available to assess the feature's effectiveness

Test distribution takes your existing test suites and distributes them across remote agents to execute them faster.

Test agents can be auto-scaled up and down.







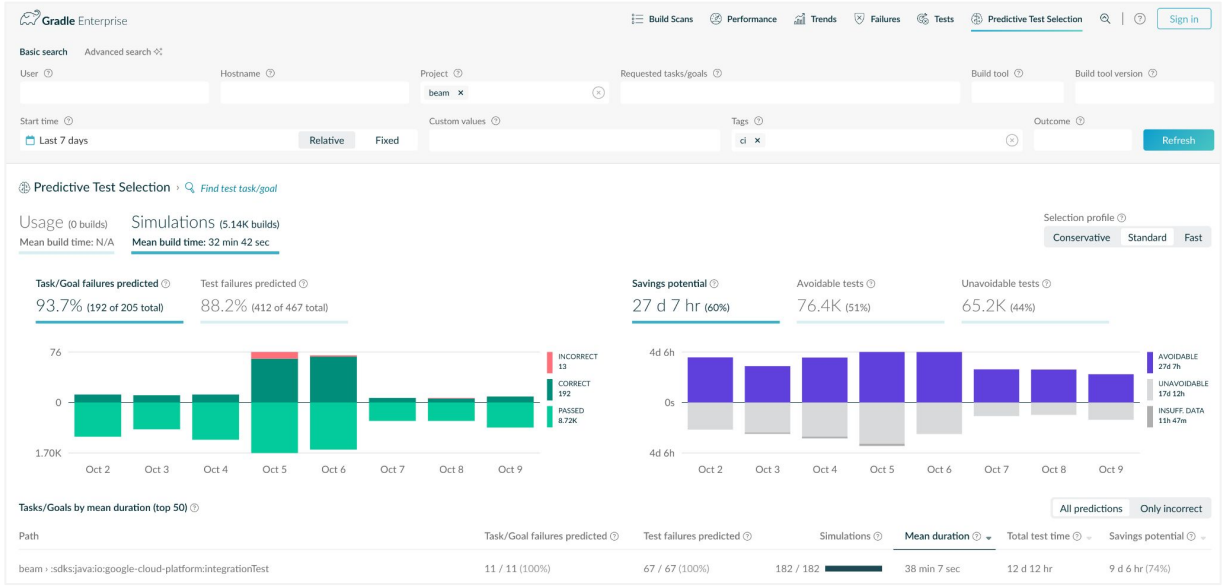
Impact of Gradle Enterprise Build Caching

In the last 28 days, the Beam project

- saved 113 days in task execution time, due to the Remote Build Cache
- ran CI builds on average in 16 min instead of 26 min per build

Impact of Gradle Enterprise Predictive Test Selection

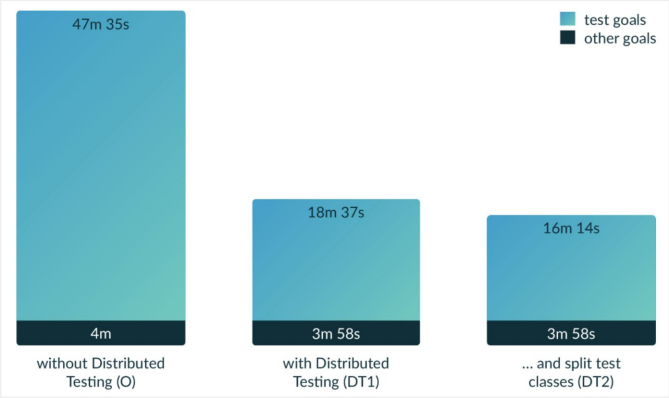
The Beam project could be saving 27 days of serial test time a week thanks to Predictive Test Selection.



Impact of Gradle Enterprise Test Distribution

The Eclipse Jetty Project can reduce their test time from 47 minutes to 16 minutes thanks to Test Distribution.

Additional optimizations can reduce their total build time from 51 minutes to 11 minutes by enabling build caching and parallel goal execution.



OSS Contributions

Developers have many OSS projects to choose from that they can contribute to:

- Popularity, relevance, activity
- Tech stack
- Impact
- Community
- etc.

Providing a first-class developer experience

Attract and retain contributors with a first-class developer experience:

- Ease and speed of building, testing, and running the project
- Efficient resolution of errors
- Ability to unblock oneself
- Effective collaboration with the committers
- etc.

Providing a first-class developer experience

Attract and retain contributors with a first-class developer experience:

- Checkout from scratch and build locally very fast
- Use build scans to self-troubleshoot in case of build problems
- Use build scans to collaborate with project maintainers on local & CI build issues
- Use build scans to identify flaky tests
- etc.

Revved up by Develocity OSS program

>25 OSS projects sponsored with a free Develocity license:

- Spring (~70% avg build time savings) – ge.spring.io
- JetBrains Kotlin (~80k builds per week) – ge.jetbrains.com
- JUnit (~55% remote cache benefit) – ge.junit.org
- Quarkus, Hibernate, Testcontainers, AndroidX, Armeria, etc.

Next steps

Embrace developer productivity:

- Visit Gradle at the conference booth
- Browse ge.apache.org
- Contact the ASF Infra team to get your project connected to Develocity (takes 5 min)
- Attend our upcoming training on how to best leverage Develocity on your project

Developer Productivity Engineering for Open Source Communities

Etienne Studer, SVP of Engineering, Develocity