

Flaky Tests & Binary Repositories

A Year in the Life of a Google DPE Engineer



Simona Bateman/Nov 3, 2022

Agenda

- Intro
- Background info
- Flaky Tests
- Migrating Binary Repositories
- Q & A

Introduction

Fitbit Developer Productivity responsibilities

- Infrastructure and tools surrounding the CI/CD workflow
- Infrastructure:
 - Jenkins
 - Binary Repositories (Artifactory/Artifact Registry)
 - Source Control (Bitbucket/Gerrit)

Our responsibilities (cont)

- Tools
 - Code Quality (Sonarqube)
 - Build Analysis (Gradle Enterprise)
 - Deployment automation/standardization
 - Flaky Test reporting
 - Gradle Plugins
 - Jenkins Libraries & Plugins

Background information

Repositories & Languages

- Code written in Java, Go, Node, Python
- Over 2000 repositories overall
- We also have a monorepo
 - Made out of Java-based microservices
 - Monolith!

Our Monorepo

- 2,000+ Gradle Projects
- 20,000+ Tests
- 40+ Gradle Plugins
- Monolith
- A build orchestrator to help build only what's necessary

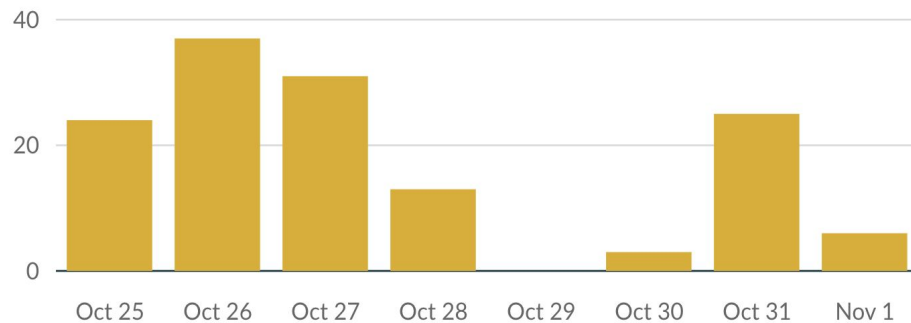
Flaky Tests

Flaky Tests Stats

- A min of 10 requests per week telling us their PRs are failing due to flaky tests
- Releases blocked weekly due to CI failing
- 200+ flaky tests in just our monorepo

Builds with flaky tests ?

139 (2% of 6.11K builds that executed tests)



Tools to identify flaky tests

- Internal built flaky test detector (repeats failed builds)

- Gradle Enterprise

- Test retry (a lot faster)

- Failure exploration

- The problem? ... People don't look at it

Name	Outcome trend ⓘ	Failed ▾	Flaky ▾	Passed
██████████ GroupEngagementThresholdServiceComponentTest		7 (13%)	20 (38%)	26 (49%)
██████████ FeedThriftServiceComponentTest		7 (13%)	17 (32%)	29 (55%)
██████████ AuthApplicationAuthorizationControllerTest		0 (0%)	15 (100%)	0 (0%)
██████████ RequestRateLimiterTest		0 (0%)	13 (76%)	4 (24%)
██████████ PublicUserServiceGrpcServerComponentTest		3 (5%)	11 (20%)	41 (75%)
██████████ ExerciseSummaryServiceComponentTest		14 (6%)	8 (3%)	227 (91%)

Flaky Tests - in Gradle Enterprise

Name	Outcome trend ?	Failed ▾	Flaky ▾	Passed
[REDACTED]GroupEngagementThresholdServiceComponentTest		7 (13%)	20 (38%)	26 (49%)
[REDACTED]FeedThriftServiceComponentTest		7 (13%)	17 (32%)	29 (55%)
[REDACTED]AuthApplicationAuthorizationControllerTest		0 (0%)	15 (100%)	0 (0%)
[REDACTED]RequestRateLimiterTest		0 (0%)	13 (76%)	4 (24%)
[REDACTED]PublicUserServiceGrpcServerComponentTest		3 (5%)	11 (20%)	41 (75%)
[REDACTED]ExerciseSummaryServiceComponentTest		14 (6%)	8 (3%)	227 (91%)

Building awareness

- Report integrating GE with our owners system
- Email + bi-weekly presentation slide

Test class	Owner	Flaky Test Count	Gradle Enterprise URL
PublicUserServiceGrpcServerComponentTest		18	Debug in Gradle Enterprise
FeedThriftServiceComponentTest		17	Debug in Gradle Enterprise
GroupEngagementThresholdServiceComponentTest		16	Debug in Gradle Enterprise
AuthApplicationAuthorizationControllerTest		15	Debug in Gradle Enterprise
RequestRateLimiterTest		10	Debug in Gradle Enterprise
UserServiceGrpcServerComponentTest		9	Debug in Gradle Enterprise
DutchAcceptanceTest		8	Debug in Gradle Enterprise
ExerciseSummaryServiceComponentTest		8	Debug in Gradle Enterprise
ExerciseSummaryServiceComponentTest		8	Debug in Gradle Enterprise
GrpcTest		8	Debug in Gradle Enterprise
ScalesAcceptanceTest		5	Debug in Gradle Enterprise
CustomerProfileAcceptanceTest		4	Debug in Gradle Enterprise
SupportToolsTabsSmartApiAccessAcceptanceTest		4	Debug in Gradle Enterprise
MigrationBatchProcessorUnitTest		4	Debug in Gradle Enterprise
DashboardAcceptanceTest		3	Debug in Gradle Enterprise

Next steps

- Priority Codes at Google (Red, Yellow)
- Code Yellow for Flaky Tests
 - All teams owning Flaky Tests work solely on fixing the tests until Code Yellow is complete
 - Director buy-in

Binary Repository Migration

Why migrate and how to choose a replacement

- Acquisition requirement (use internal or 1st party tools wherever possible)
- All our infrastructure/services already in GCP



Designing and Planning

- Things to take into account
 - Number of repositories in Artifactory (21 local, 57 remote, 13 virtual)
 - Security
 - Features required and workarounds if features don't exist
 - Make it self-service
 - The DEADLINE!

Challenges and workarounds

- Artifactory: older, feature-rich product
- Artifact Registry: new product, features are still being added
- Creating our roadmap based on Artifact Registry's roadmap
- Getting creative when roadmap is delayed

Challenge 1: Remote repositories (Maven)

- Problem: Artifact Registry currently supports only Maven Central
 - We need 18 external repositories supported
- Solution:
 - Manual mirroring & Artifactory as backup until required feature is available

Challenge 2: Artifact metadata

- Problem: We need to add metadata to artifacts (e.g. owners, whether it can be removed during regularly scheduled cleanup)
- Solutions:
 - Feature request to the Artifact Registry team
 - Custom DB + service to store the metadata

Challenge 3: Getting teams to test & migrate (Maven)

- Spreadsheets for keeping track of project/repository migration status
- Documentation & Instructions
- Emails/Announcements
- Artifactory Brownouts

Challenge 4: Resulting incidents

- Local machines not connecting to Artifact Registry
- The monolith can't be released!!!!
 - Custom publishing that doesn't follow standard protocol
 - Mystery compilation errors

Next Steps

- The show must go on!
- Docker, Python, Go, Debian
- Implementing our workarounds
- Do not underestimate legacy code

Thank You