splunk> .conf2017

Splunk Helping in Productivity

Sumit Nagal | Principal Engineer , Intuit | @sumitnagal

27-Sept-2017| Washington, DC

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

splunk> .conf2017

About Me

Work

intuit

turbotax quickbooks proconnect mint

Play

Java

splunk>

CHEF
GETCHEF.COM

dynatrace

New Relic.

AppDynamics

OverOps

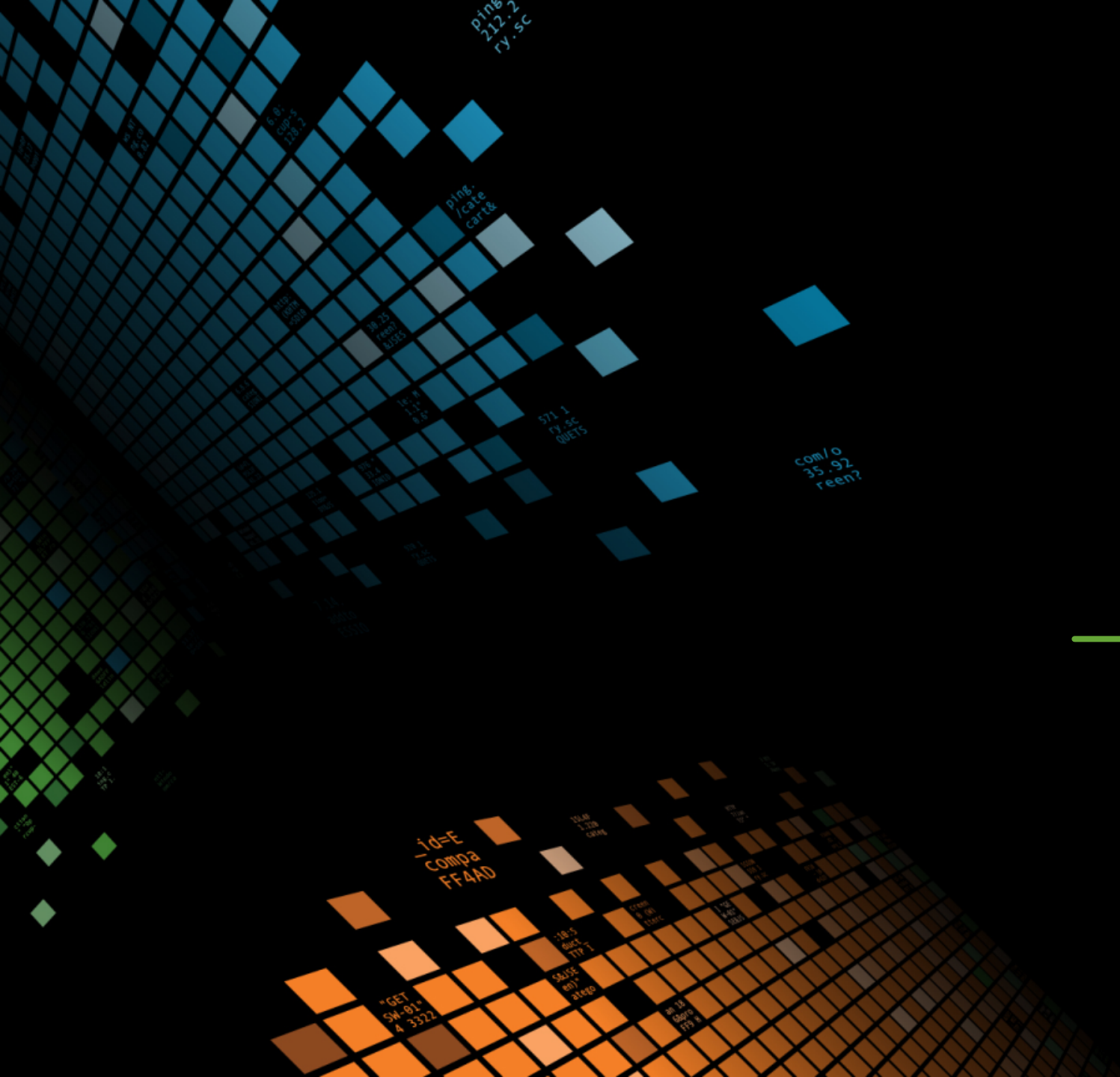
node.js

CloudBees
The Enterprise Jenkins Company

splunk> .conf2017

Agenda

- Overview
- In Quality
- In Development
- Improving Velocity
- Key Takeaways
- Q &A



Overview

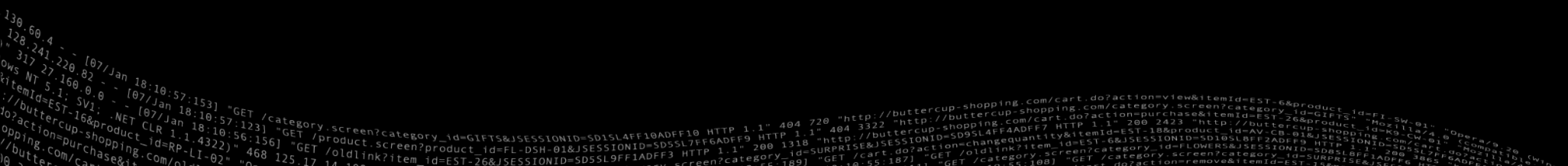
Journey with Splunk in productivity

splunk> .conf2017

Overview

Jenkins and Splunk Background

- Many Jenkins
- Many Splunk
- Thousands of build, test and deploy data per day
- Splunk only for Application Logs
- Collaboration with development Tools challenge
- Data is scattered
- Splunk is Operation Tool



splunk> .conf2017

Journey Line

How we started using Splunk for productivity

Quality

- Splunk Validator
- Jira Validator
- Overops
- Forensic Dashboard

2016-July

Quality

Jenkins App

- Splunk App
- Jenkins App
- Conf 2016
- Jenkins Conf

2016-Sept

Operation

Development

- Build Optimization
- Splunk maven plugin
- Build Dashboard
- Webinar

2016-Dec

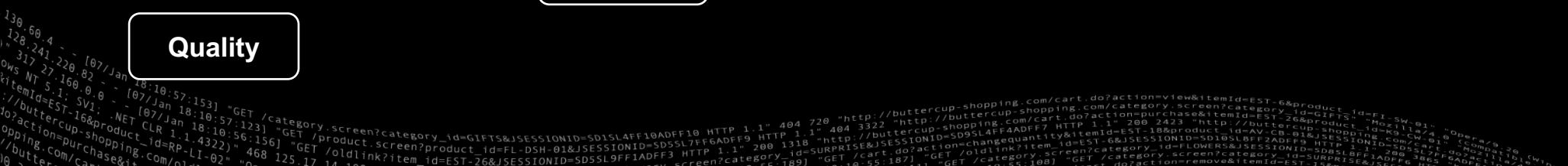
Developer

Velocity

- Visibility Jenkins Farm
- Collaboration Tools Integration
- Next Gen Dashboard
- Jenkins App integration

2017- Mar

Management



splunk> .conf2017

Quality

Quality productivity – pulling data from Splunk

splunk> .conf2017

“Scaling the Automation with Quality.”

Quality

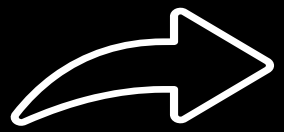
Problem	Opportunity	Solution
Test Failures	Accountability	Jira Validator
Root Cause Analysis	Traceability	Splunk Validator
Process	Transparency	Integrated with Test

splunk> .conf2017

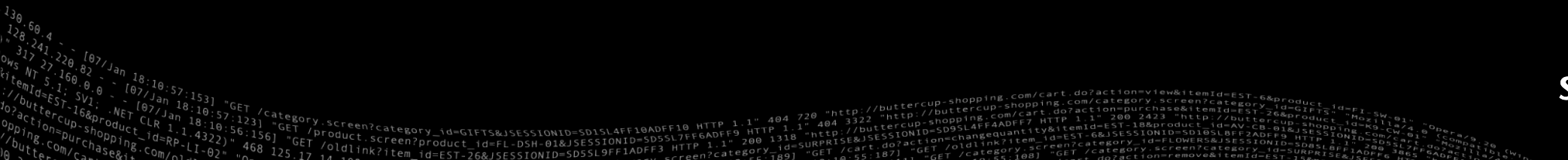
Quality – Solution

Scaling the Automation with Quality

Jira/Splunk Validator



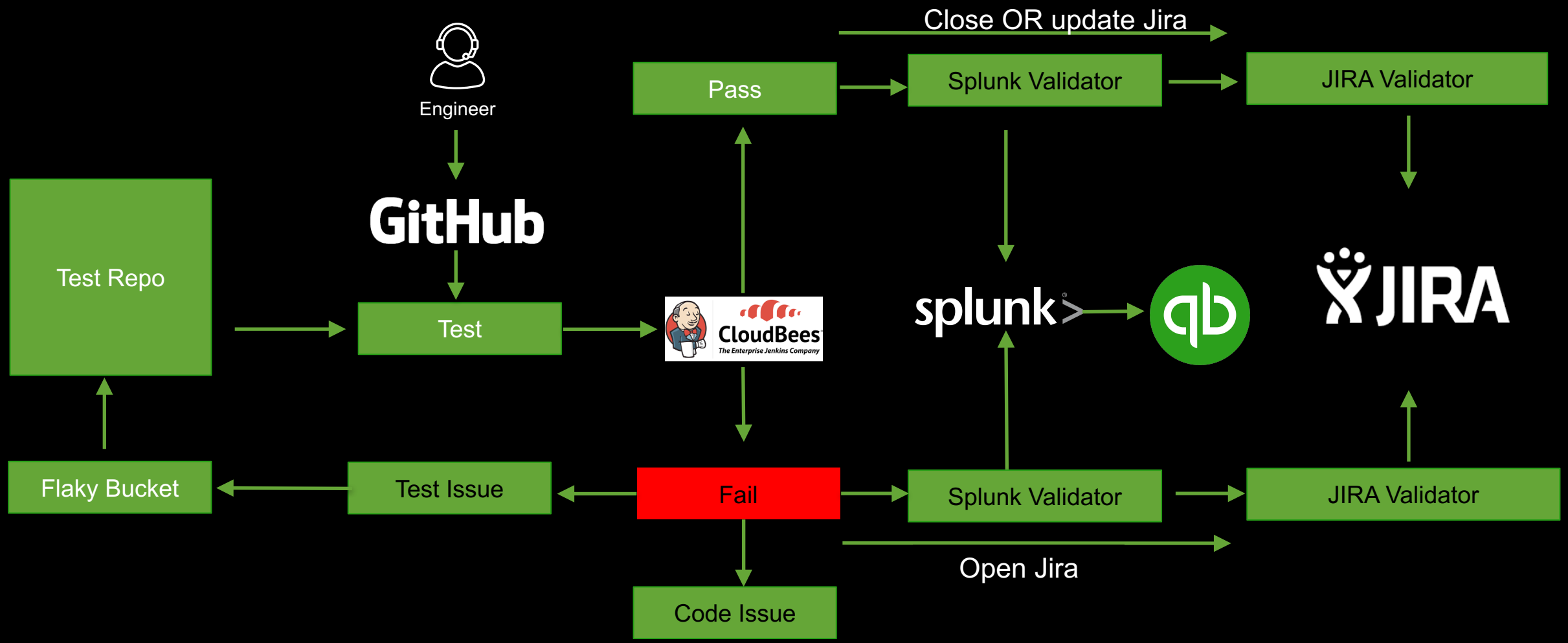
Test Integration with
Splunk / Jira Validator



splunk> .conf2017

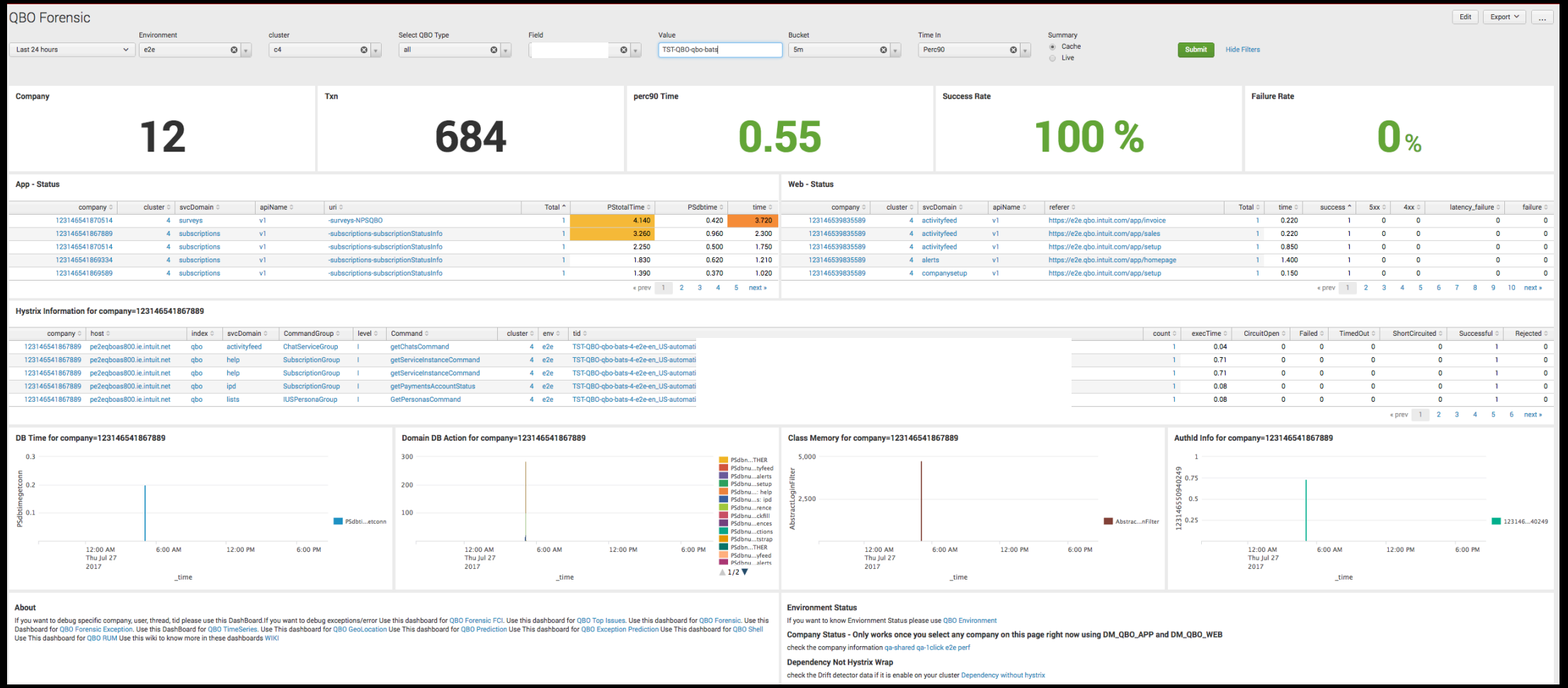
Quality - Solution

Scaling the Automation with Quality



Quality – Performance Time

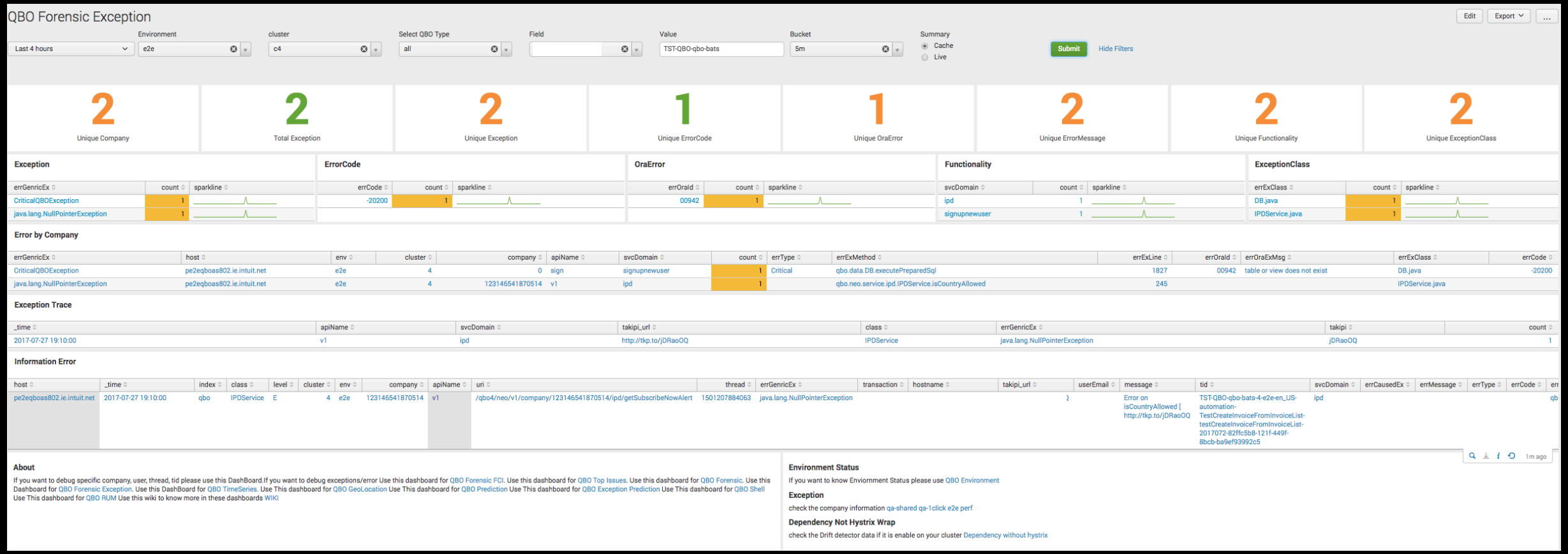
How we are identity opportunity, reporting and taking action – pulling from Splunk



splunk> .conf2017

Quality – Application Error

How we are identity opportunity, reporting and taking action – pulling from Splunk



Development

Developer productivity – pushing data in Splunk

splunk> .conf2017

“Making developer 10X Productive.”

splunk> .conf2017

Developer

Problem	Opportunity	Solution
Build Time	Visibility	Splunk Data uploader
Build Failures	Problem identification	Error Categorization
Root Cause Process	Self Healing	User contribution

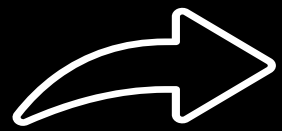
splunk> .conf2017

Developer – Solution

Making developer 10X Productive

splunk> Maven™

Splunk Data uploader



splunk>

Build Dashboard, User contribution system

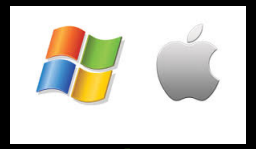
splunk> .conf2017

Developer - Solution

Making developer 10X Productive

Engineer

Local Build



Local Build Data

Splunk Data Uploader

GitHub

PR Build

PR Build Data

Splunk Data Uploader

splunk>

CI Build

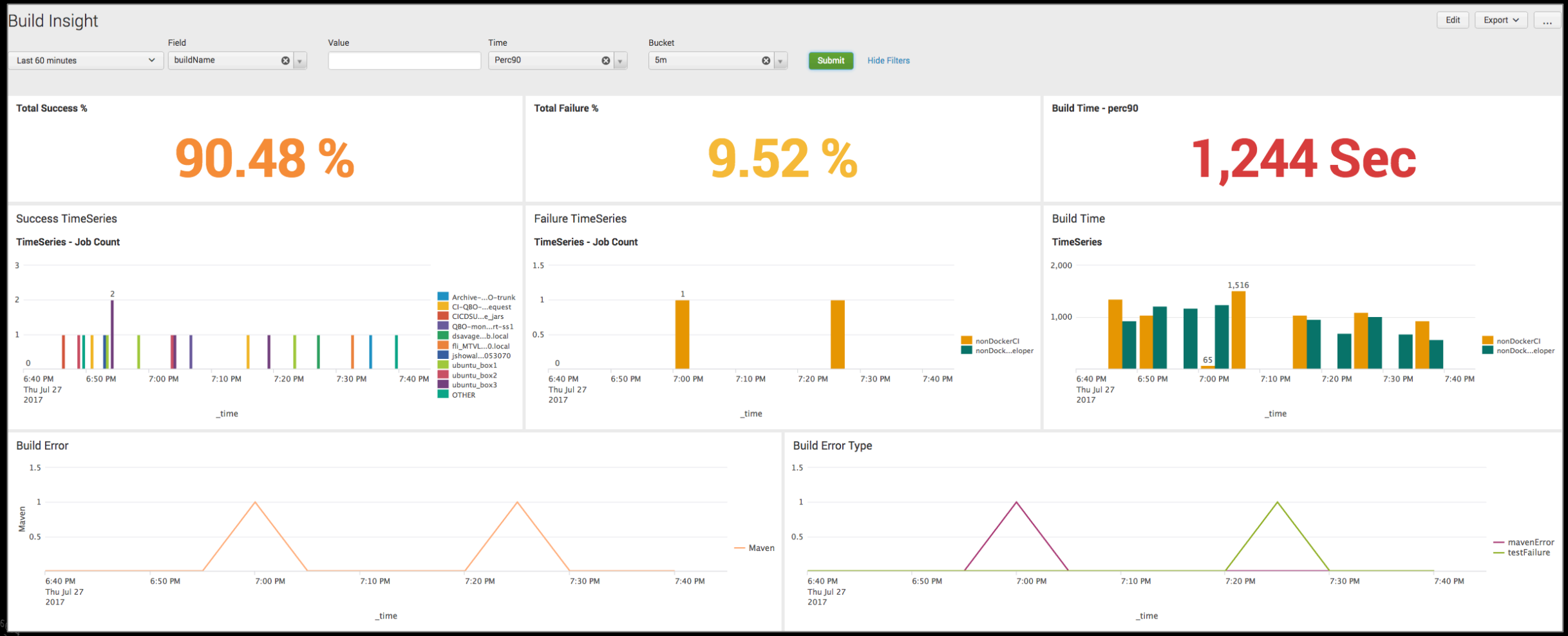
CI Build Data

Splunk Data Uploader

splunk> .conf2017

Development - Build Insight

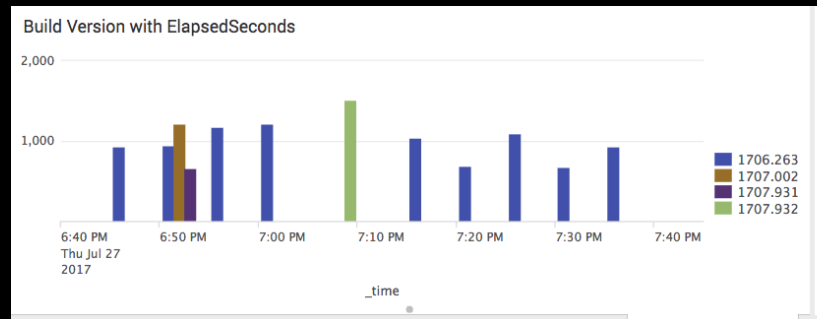
How we are getting build error across developer, PR and CI - pushing to Splunk

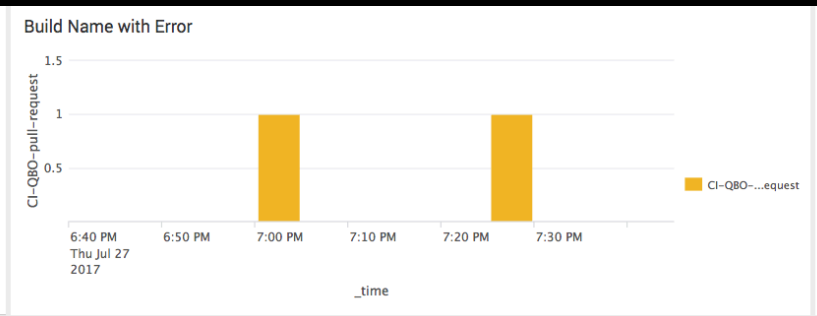


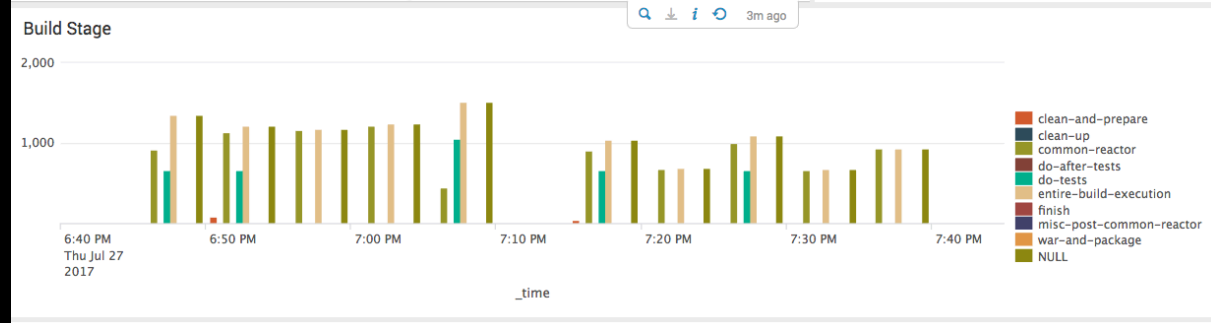
splunk> .conf2017

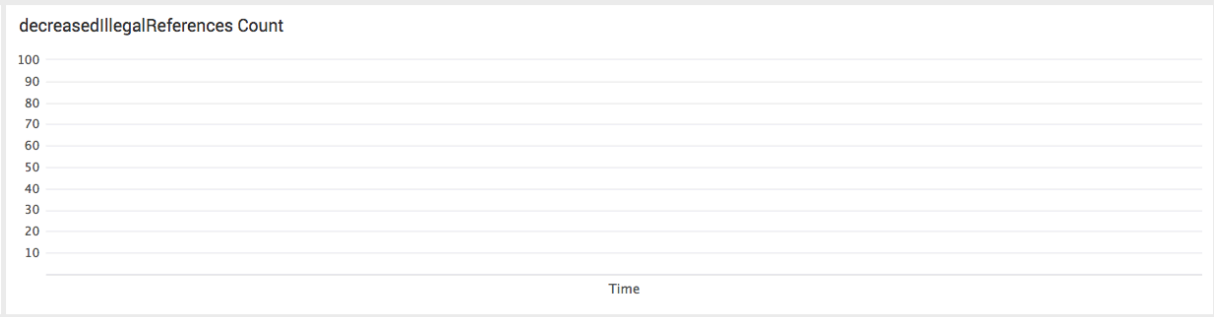
Development - Build Insight

How we are getting build error across developer, PR and CI - pushing to Splunk









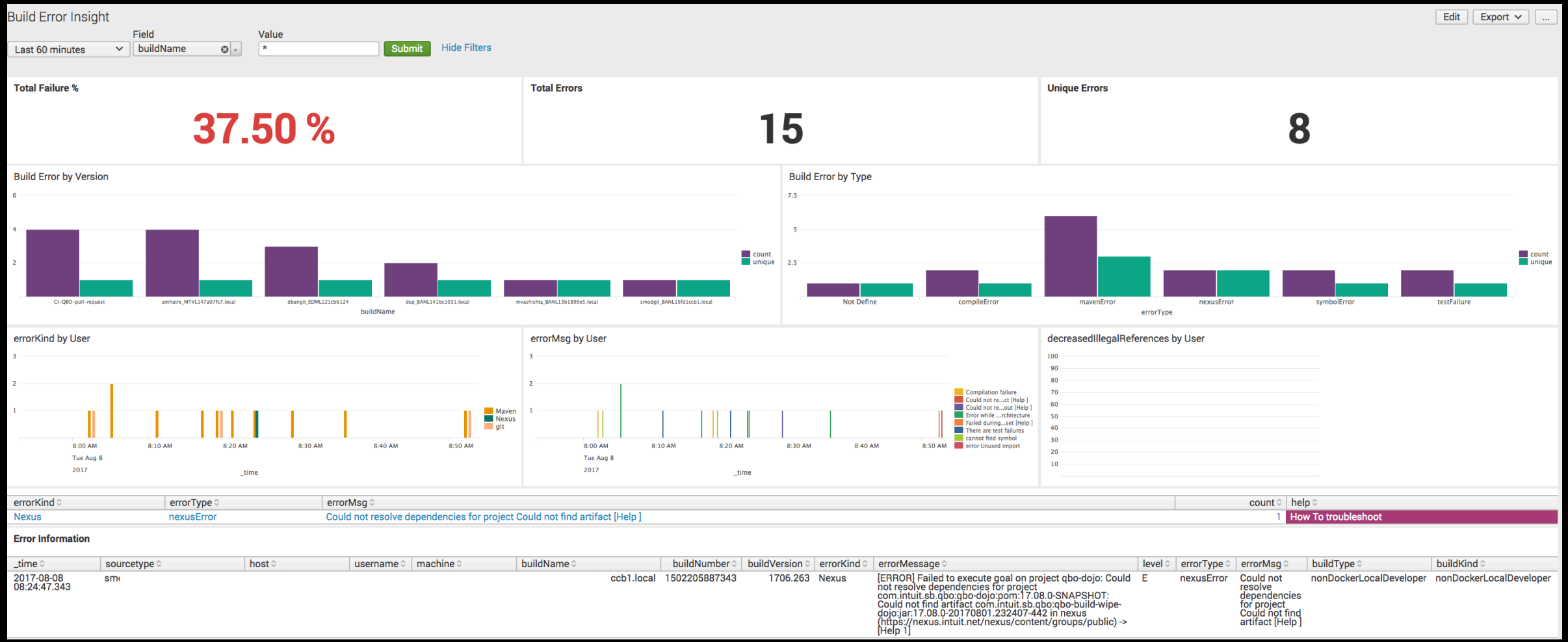
Job Information

buildSection	_time	sourcetype	host	username	elapsedSeconds	machine	buildName	buildNumber	buildVersion	errorKind	errorMessage	level	errorType	errorMsg	buildType	buildKind
	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	937	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263				Not Define		nonDockerCI	nonDockerCI
entire-build-execution	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	937	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
finish	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	0	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
do-after-tests	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	0	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
war-and-package	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	0	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
clean-up	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	1	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
do-tests	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	0	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
misc-post-common-reactor	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	0	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
common-reactor	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	931	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI
clean-and-prepare	2017-07-27 19:39:43.365	scmbuild_mtv-ubld-115	mtv-ubld-115	scmbuild	5	mtv-ubld-115	Archive-DY-QBO-trunk	6902	1706.263			I	Not Define		nonDockerCI	nonDockerCI

splunk> .conf2017

Development - Build Error Insight

How we are getting build error across developer, PR and CI - pushing to Splunk



splunk> .conf2017

Improving Velocity

Improving Velocity – Leveraging Splunk Jenkins App

splunk> .conf2017

“If You can’t Measure it, You Can’t Improve it.”

Journey Line – Jenkins App

How we started using Splunk for productivity

- Local Jenkins
- Local Splunk
- Jenkins Conf

2016-Aug

POC

- Splunk App
- Jenkins App
- Conf 2016
- 3 instances

2016-Sept

Tools Instance

- Performance Fixes
- Coverage Fix
- Dashboard improvement
- Webinar

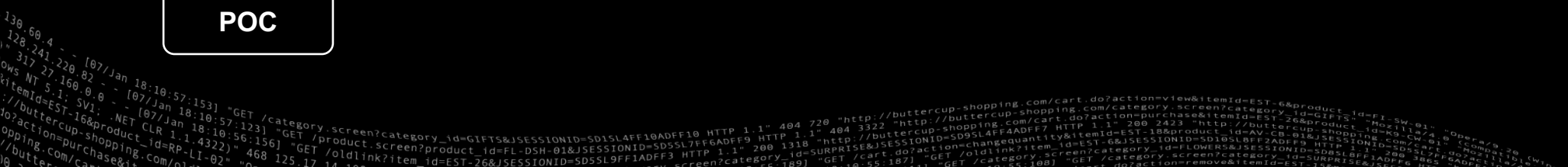
2016-Jan

Pre prod

- Production roll out
- Visibility Jenkins Farm - SBG
- Collaboration Tools Integration
- Next Gen DashBoard

2017- Mar

Production



splunk> .conf2017

Improving Velocity

Problem	Opportunity	Solution
Many Data Sources	Single Source	Splunk Jenkins App
Identify Gaps	Exposing Data	Splunk Dashboard
Right Owner	Define Ownership	Integration with Collaboration tools

splunk> .conf2017

Improving Velocity– Solution

If You can't Measure it, You Can't Improve it

GitHub

JIRA

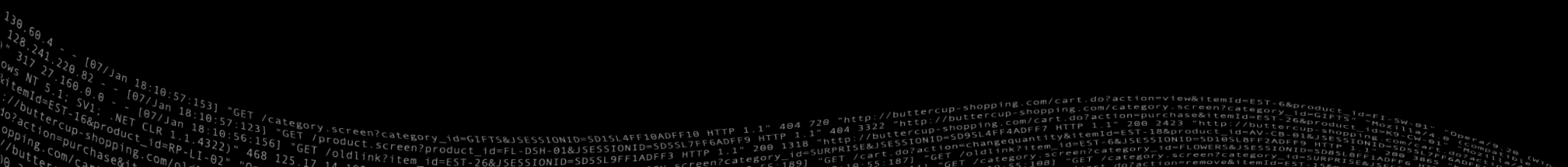
workday.

CloudBees
The Enterprise Jenkins Company

Integration with
Collaboration Tools

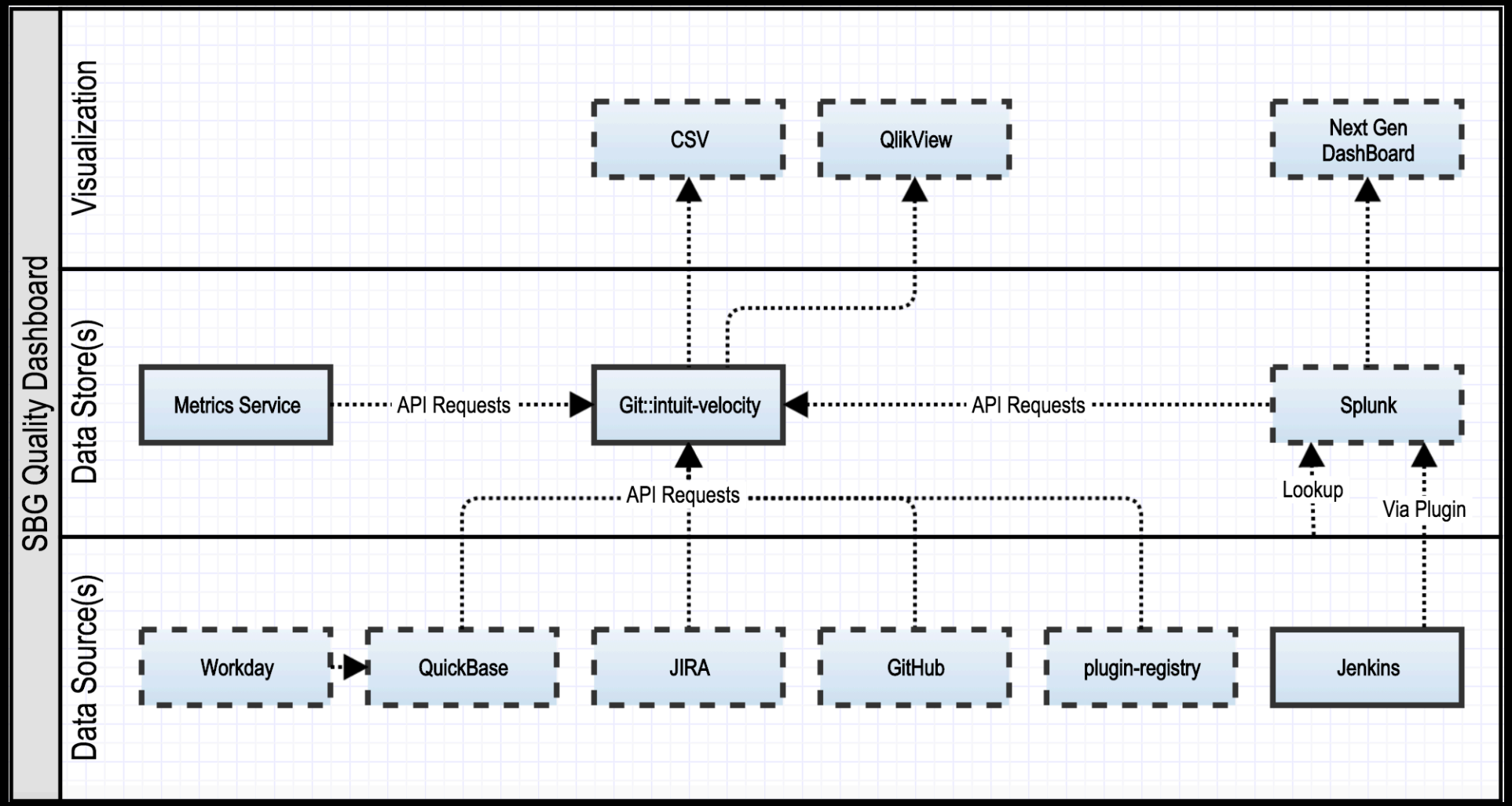
splunk>

Splunk Jenkins App with
Velocity data



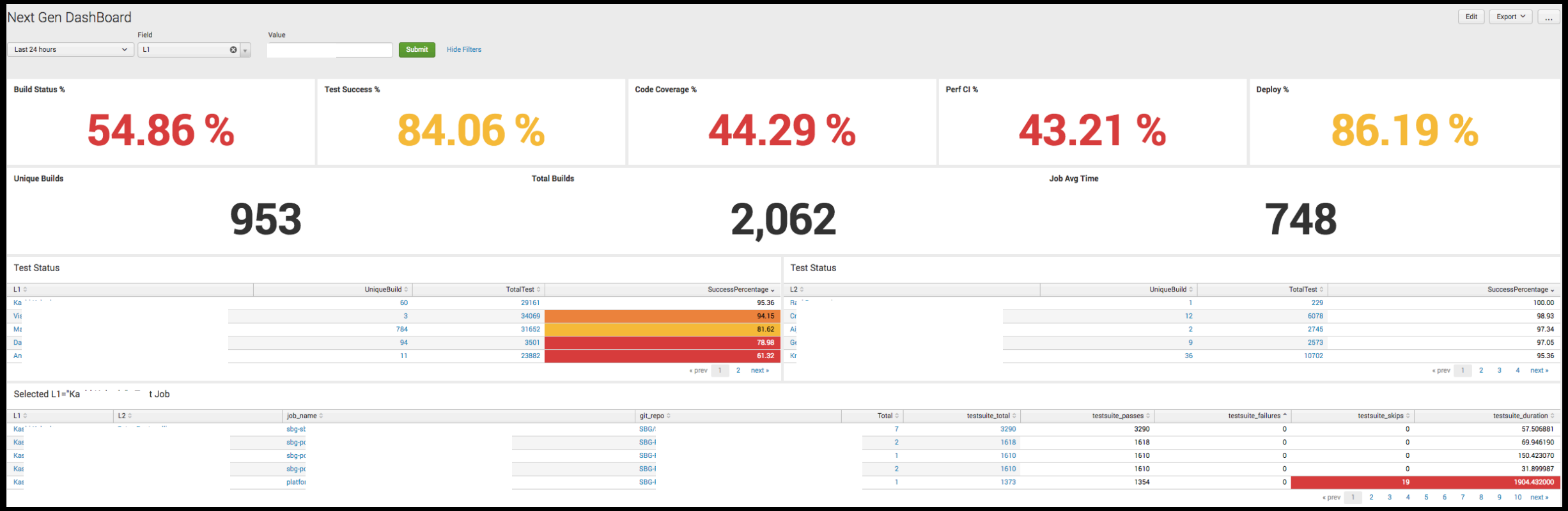
splunk> .conf2017

Improving Velocity - Design



Velocity - Overall

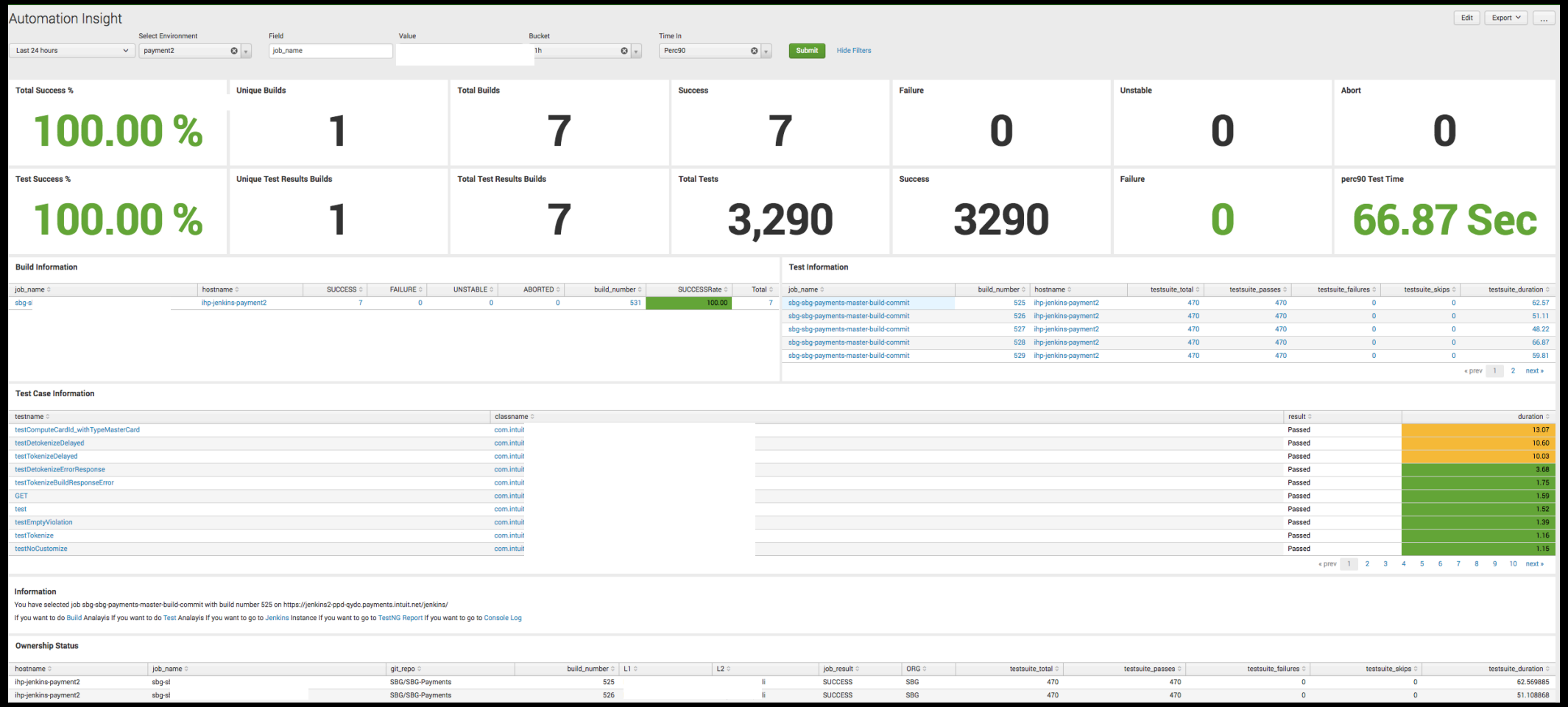
How we are identity opportunity, reporting and taking action – using Jenkins App



splunk> .conf2017

Velocity - Test

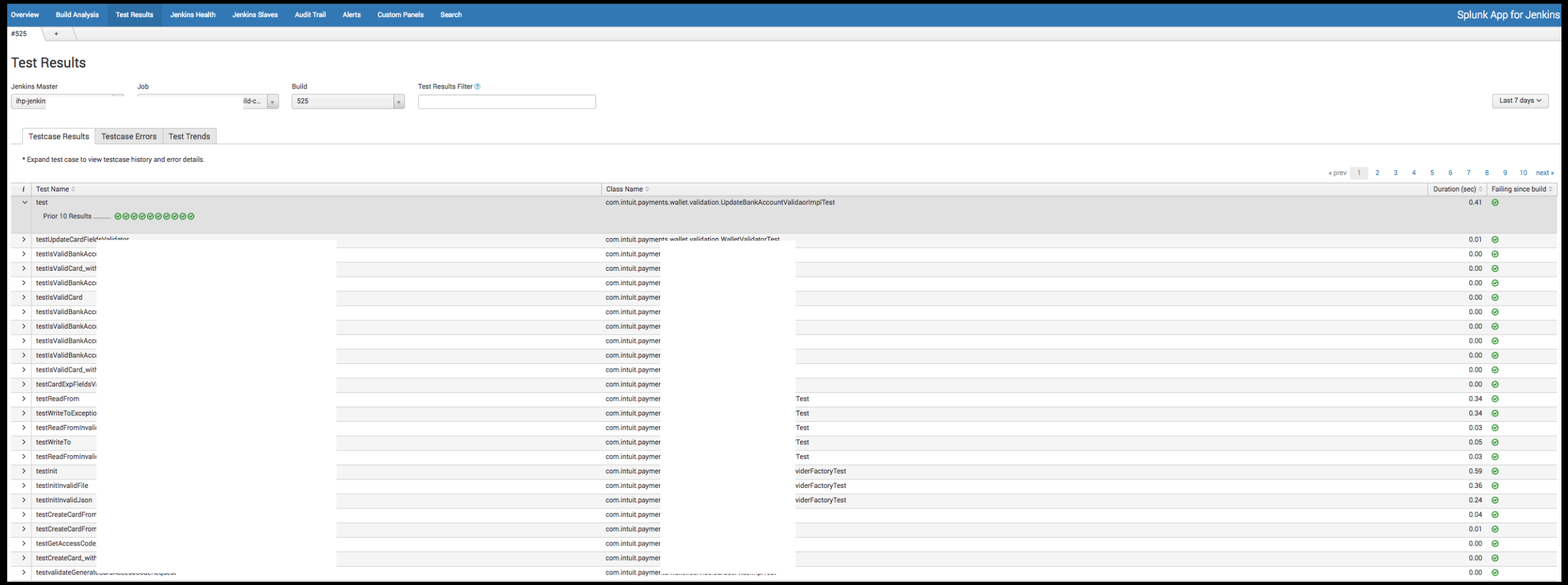
How we are identity detail information - using Jenkins App



splunk> .conf2017

Velocity – Jenkins App

How we are connecting to Jenkins App



splunk> .conf2017

Overall Improvement

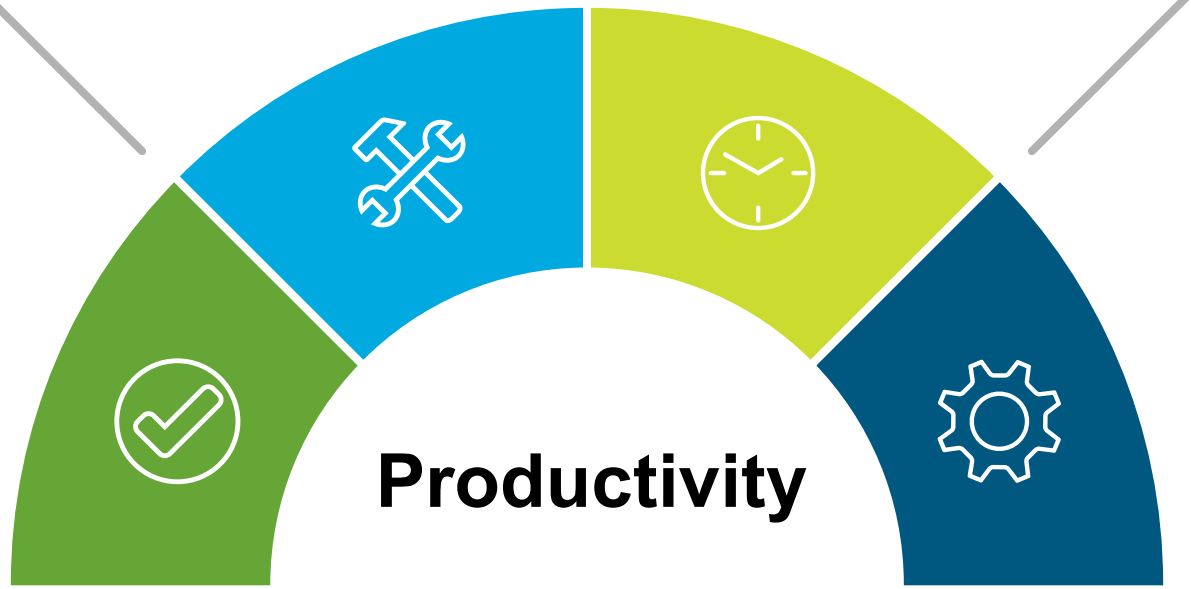
Key Takeaways

Scaling the Automation

Manual to Auto-magic
Context to Core

Measure & Improve Velocity

Visibility
Ownership /Accountability



Development

Time (from 50+ to 20+ min)
Success (80% to 98%)
Issues(Many to Few)

Productivity

Data is Key

Collaboration of Data
Integration of Tools

splunk> .conf2017

Using Splunk to Improve Splunk's Build Process

Eddie Shafaq

Splunk Infrastructure

First joined Splunk in August 2011 as a Systems Administrator. Aided in expanding engineering support in "exotic operating system" (AIX, HPUX, S390X and PowerLinux). Served as a member of release engineering to address operational and infrastructure support for products team. Currently serving an infrastructure leadership role around Core Engineering and Release Engineering services.

Bill Houston

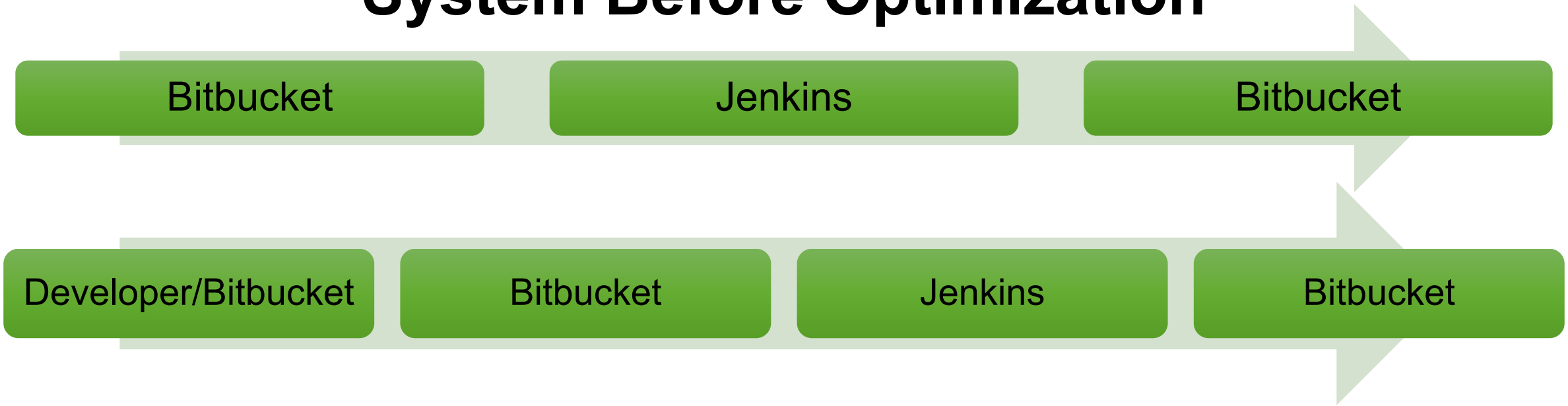
Splunk Infrastructure

Bill started his career as an analog hardware engineer designing professional recording equipment. Currently he is a senior release engineer at Splunk working on improvements to the Jenkins CI systems. Prior to Splunk he spent 16 years at Adobe working in various roles; the last four were spent using Jenkins to build and test Adobe Flash.

The Goal Set by Management

- Improve Developer Productivity
  - Our contribution: Get CI test results to developers faster

System Before Optimization

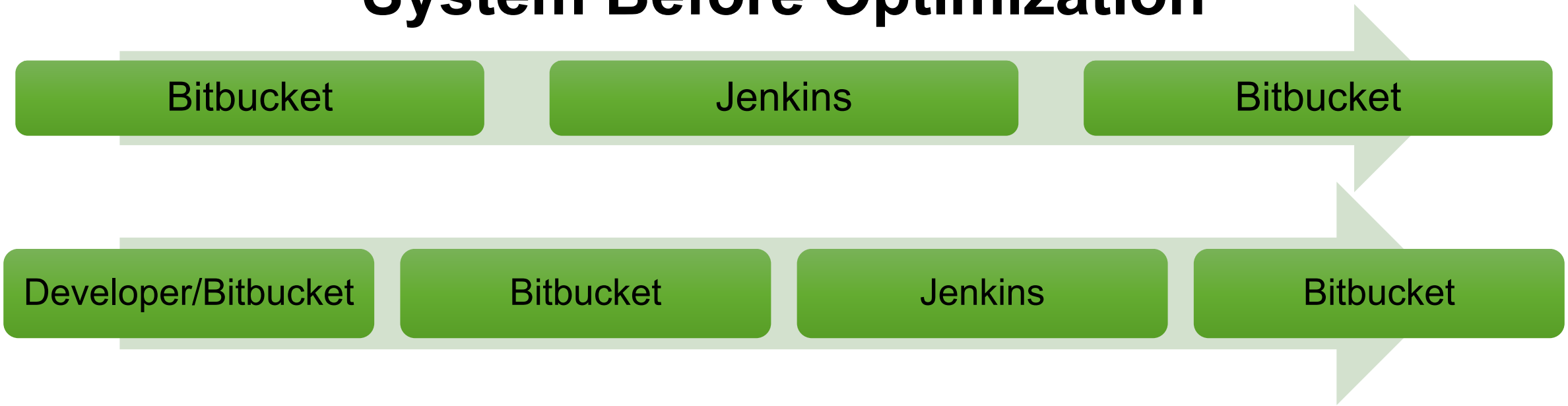


The system had **15 dedicated Linux agents** to perform continuous integration testing

Each job ran for approximately **54 minutes**, performing a build of Splunk and running a set of validation tests

That meant it could perform an approximate average of **17 jobs per hour**

System Before Optimization



The system had **15 dedicated Linux agents** to perform continuous integration testing

Each job ran for approximately **54 minutes**, performing a build of Splunk and running a set of validation tests

That meant it could perform an approximate average of **17 jobs per hour**

System Before Optimization

If more than **17 triggers** were received in a **one hour** period the excess **triggers were queued** waiting for a Linux agent to run on

Under “normal” circumstances the system operated with minimum delays, however during peak load periods when the pressure on developers was the highest…

We experienced **significant delays** resulting in frustration and phone calls as the **engineers waited** for results of the validation test jobs they were required to run before they could commit thier work

It was Obvious

We Needed To Speed Things Up!

Our Analysis

Our Analysis

Understanding the Situation

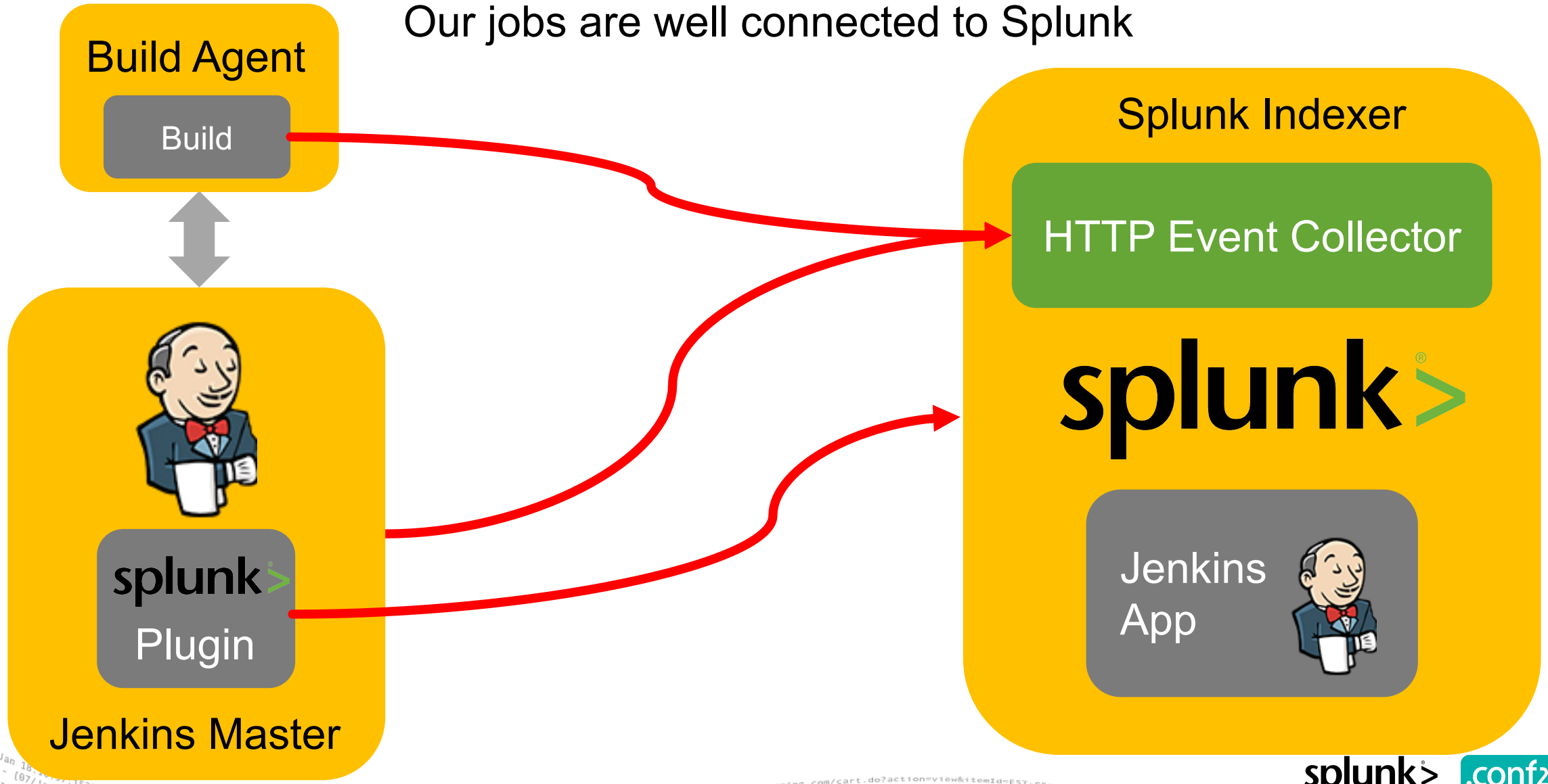
There were four factors that affected the delay developers experienced while waiting for test results

- Build time – how long it takes to build the Splunk executables
- Test time – how long it takes to perform the required set of tests
- Queue time – how long before the test actually started to run
- Notification – how long before developers know the test results

We need to quantify each of those factors and determine what we could do to mitigate their effects on the overall time

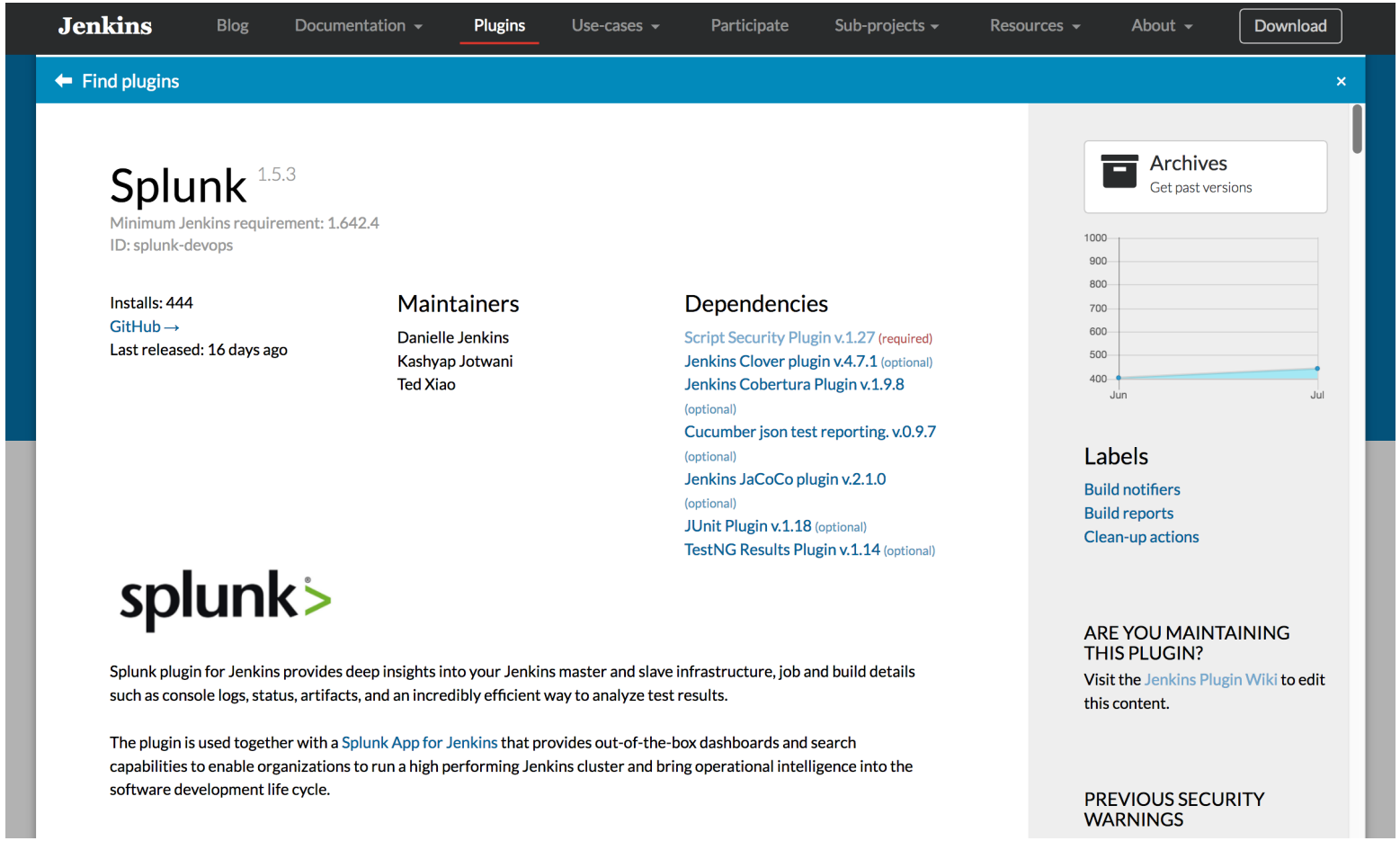
How We Collected The Data

Our jobs are well connected to Splunk



The Splunk Plugin for Jenkins

Makes it Easy to Send **Your** CI/CD Data to Splunk



https://wiki.jenkins.io/display/JENKINS/Splunk+plugin+for+Jenkins

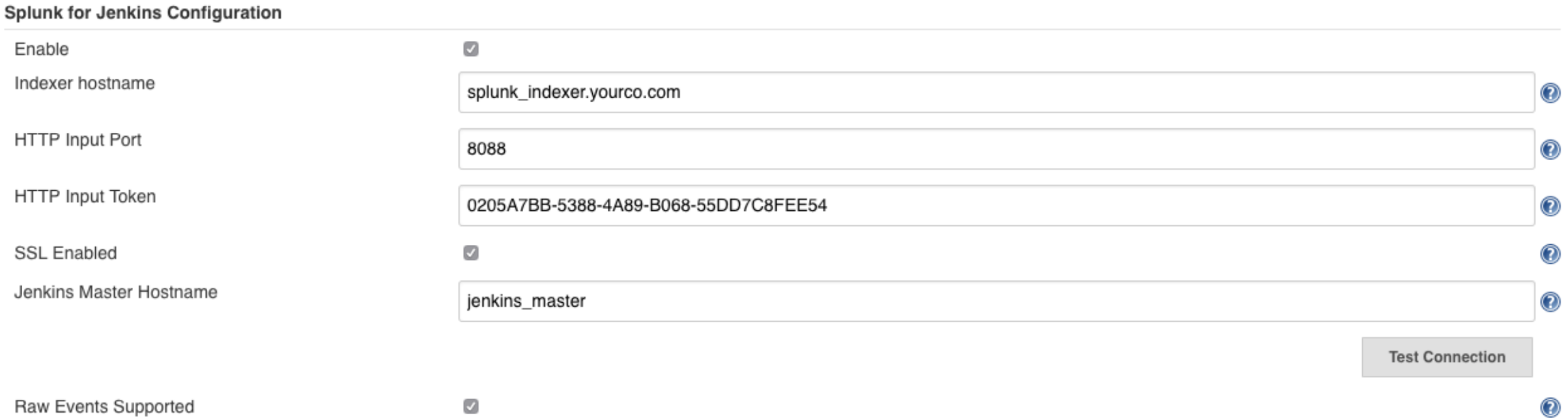
splunk> .conf2017

The Splunk Plugin for Jenkins

Easy to install

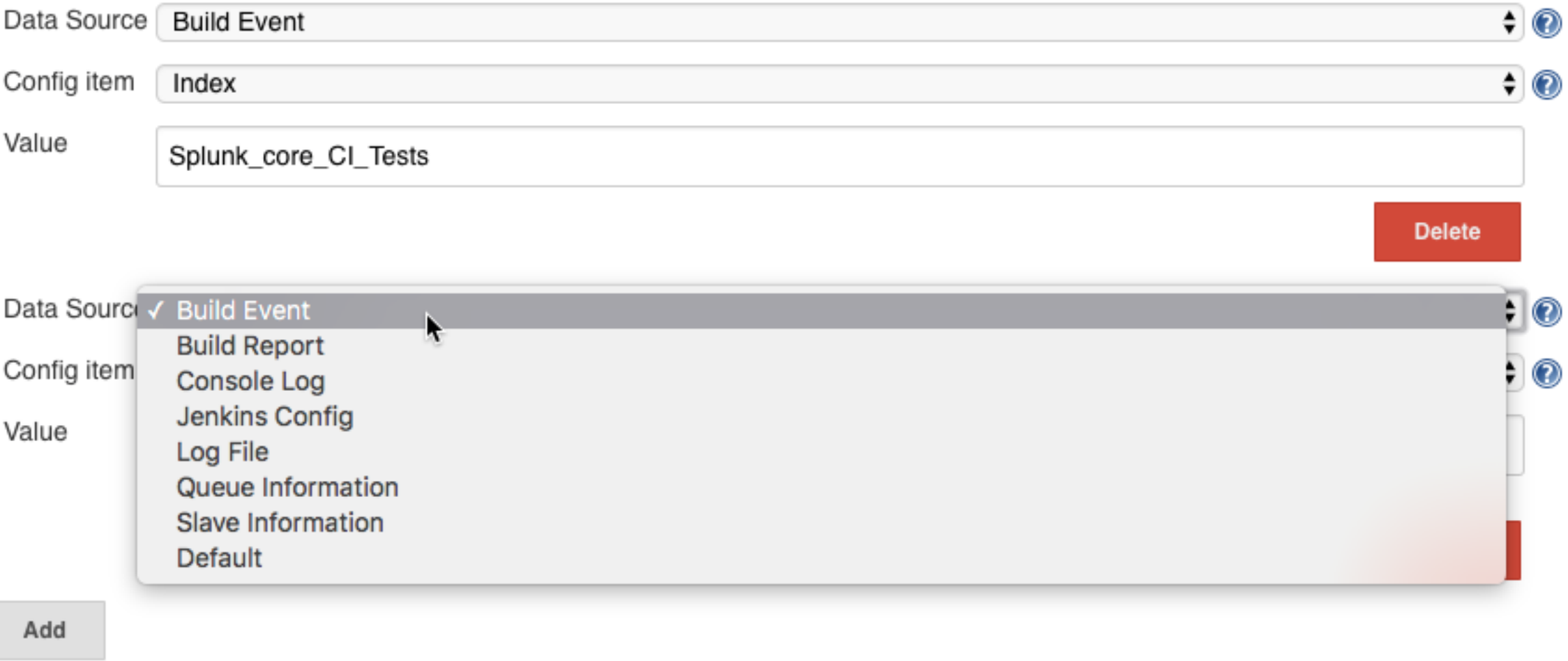
The Splunk Plugin for Jenkins

Easy to install



The Splunk Plugin for Jenkins

Easy to customize



splunk> .conf2017

The Splunk HTTP Event Collector

Simple to send custom data to your Splunk instance

In Bash

Format your data as a JSON string:

```
jsonData="{\"time\": 12345, \"index\": \"YourIndex\", \"sourcetype\": \"YourSourceType\", \"source\":
\"YourSource\", \"event\": \{\"YourFieldName\": \"SomeData\", more json formatted data goes here}}”
```

Include as much json formatted information as you need in the event section

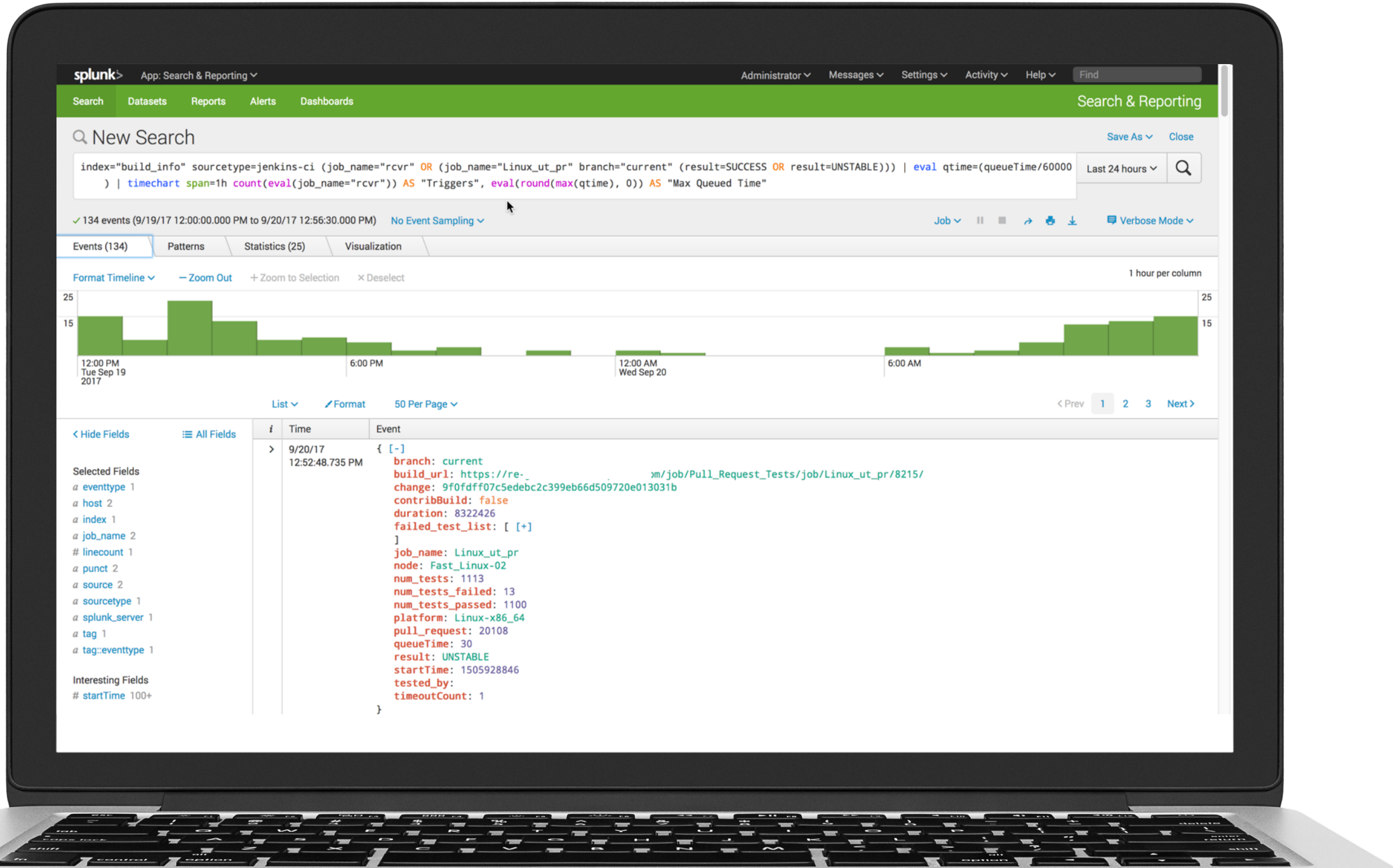
Then execute a curl call:

```
curl \
  --tlsv1.2  --header "Authorization: <Splunk_auth_token_goes_here>" \
  --header "Content-Type: application/json" \
  --request 'POST' \
  --data $jsondata \
```

https://YourSplunkInstance/services/collector/event

Its that simple…

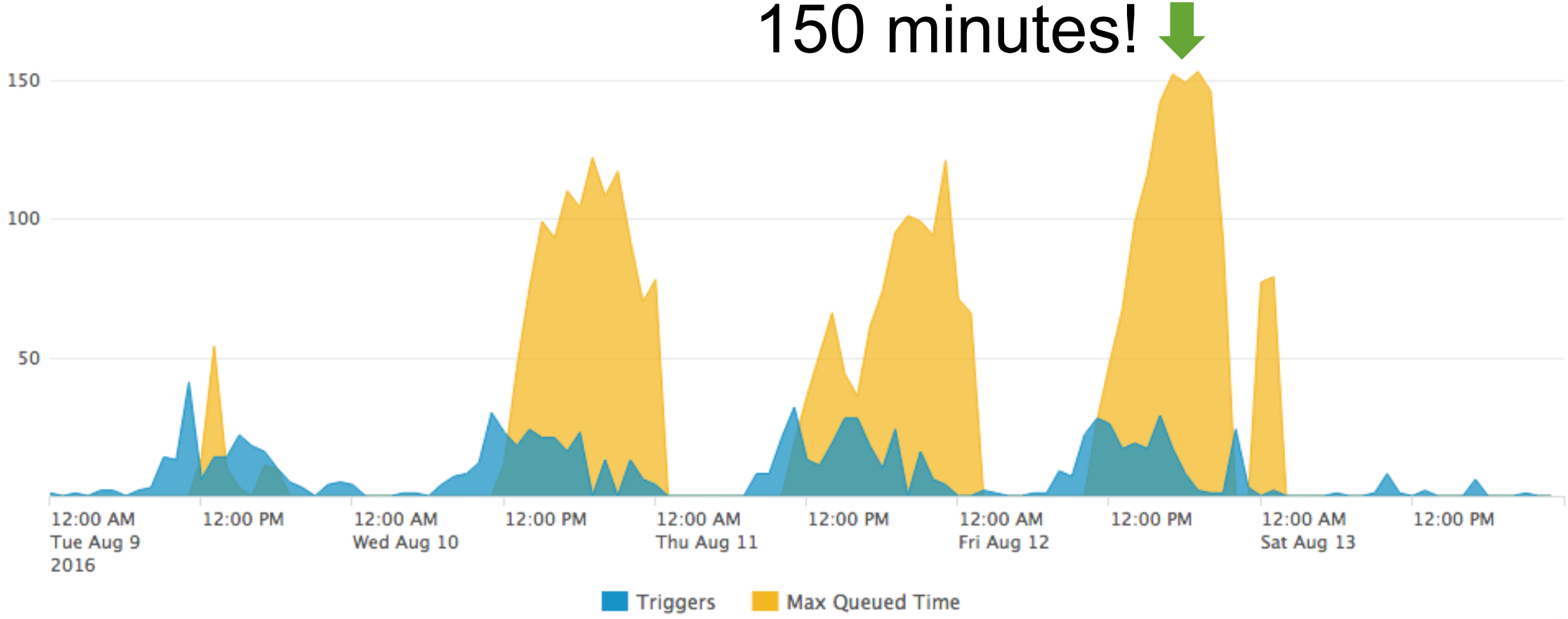
splunk> .conf2017



Queue Times Before Optimization

Peak load period had significant delays

150 minutes!



splunk> .conf2017

The Search We Use to Analyze Jobs

Of course this won't work for you, but…

```
index="jenkins_console"  host=”aJenkins.ourco.com" source="*Linux_ut_pr*" ("make -j48 || exit 0" OR "Install the project..." OR "Core build is done" OR "run the tests again" OR "Starting backend unit tests" OR "Package and publish Splunk" OR "starting Linux 64 test" OR "fetch the jenkins scripts directory" OR ("nodes run &gt;&gt;&gt;&gt; STARTING ACTION" AND "Write splunk-version.txt") OR ("STARTING COMMAND" AND "Running the contrib command") OR "Done all requested steps") | rex field=source "job/Pull_Request_Tests/job/Linux_ut_pr/(?&lt;build_number&gt;.*)/console" | eval buildStep=case(searchmatch("fetch the jenkins scripts directory"),"start", searchmatch("starting Linux 64 test"),"clone", searchmatch("Running the contrib command"),"chroot", searchmatch("Write splunk-version.txt"),"contrib", searchmatch("make -j48 || exit 0"),"setup", searchmatch("Core build is done"),"build_1", searchmatch("Install the project..."), "build_2", searchmatch("Starting backend unit tests"), "package", searchmatch("run the tests again"), "tests_1", searchmatch("Package and publish Splunk"), "tests_2", searchmatch("Done all requested steps"), "publish") | chart values(_time) by build_number, buildStep    | eval gc = round(('clone' - 'start')/60) |eval cs = round((chroot - 'clone')/60) | eval "cb" = round((contrib - chroot)/60) | eval "bs" = round((setup - contrib)/60) | eval "cub" = round((build_1 - setup)/60) | eval "cbc" = round((build_2 - build_1)/60) | eval "ts" = round((package - build_2)/60) | eval "pst" = round((tests_1 - package)/60) | eval "sst" = round(('tests_2' - 'tests_1')/60) | eval "pub" = round(('publish' - 'tests_2')/60) | search cb &lt; 5 | search sst &gt; 0 | search pst &lt; 25 |chart values(pub) as Publishing, values(sst) as "Sequential Smoke Tests", values(pst) as "Parallel Smoke Tests", values(ts) as "Test Setup", values(cbc) as "Core Build Continues", values(cub) as "Core and UI build", values(bs) as "Build Setup", values(cb) as "Contrib Build", values(cs) as "Chroot Setup", values(gc) as "Git Clone" by build_number
```

splunk> .conf2017

The Search We Use to Analyze Jobs

Collect the specific log file lines we will use in our analysis

```
index="jenkins_console"  host="_aJenkins.ourco.com
" source="*Linux_ut_pr*" ("make -j48 || exit 0" OR
"Install the project..." OR "Core build is done" OR
"run the tests again" OR "Starting backend unit
tests" OR "Package and publish Splunk" OR "starting
Linux 64 test" OR "fetch the jenkins scripts
directory" OR ("nodes run >>>> STARTING ACTION" AND
"Write splunk-version.txt") OR ("STARTING COMMAND"
AND "Running the contrib command") OR "Done all
requested steps")
```

The Search We Use to Analyze Jobs

Extract the job number

```
| rex field=source
"job/Pull_Request_Tests/job/Linux_ut_pr/(?&lt;build
_number&gt;.*)/console
```

The Search We Use to Analyze Jobs

Build a table of time stamps by job number and job step

```
| eval buildStep=case(searchmatch("fetch the jenkins scripts directory"),"start", searchmatch("starting Linux 64 test"),"clone", searchmatch("Running the contrib command"),"chroot", searchmatch("Write splunk-version.txt"),"contrib", searchmatch("make -j48 || exit 0"),"setup", searchmatch("Core build is done"),"build_1", searchmatch("Install the project..."), "build_2", searchmatch("Starting backend unit tests"), "package", searchmatch("run the tests again"), "tests_1", searchmatch("Package and publish Splunk"), "tests_2", searchmatch("Done all requested steps"), "publish")

| chart values(_time) by build_number, buildStep limit=50
```

The Search We Use to Analyze Jobs

Calculate the deltas between job steps

```
| eval gc = round(('clone' - 'start')/60) |eval cs
= round((chroot - 'clone')/60) | eval "cb" =
round((contrib - chroot)/60) | eval "bs" =
round((setup - contrib)/60) | eval "cub" =
round((build_1 - setup)/60) | eval "cbc" =
round((build_2 - build_1)/60) | eval "ts" =
round((package - build_2)/60) | eval "pst" =
round((tests_1 - package)/60) | eval "sst" =
round(('tests_2' - 'tests_1')/60) | eval "pub" =
round(('publish' - 'tests_2')/60) | search cb &lt;
5 | search sst &gt; 0 | search pst &lt; 25
```

The Search We Use to Analyze Jobs

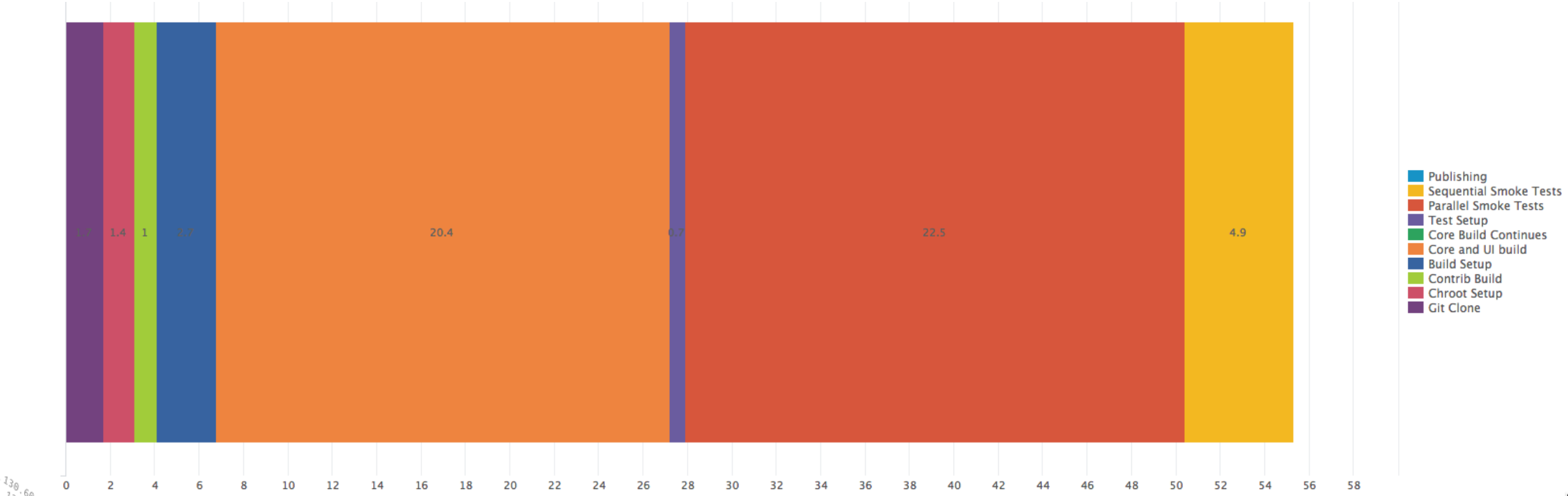
Build the final table with user friendly names for display

```
|chart values(pub) as Publishing, values(sst) as "Sequential Smoke Tests", values(pst) as "Parallel Smoke Tests", values(ts) as "Test Setup", values(cbc) as "Core Build Continues", values(cub) as "Core and UI build", values(bs) as "Build Setup", values(cb) as "Contrib Build", values(cs) as "Chroot Setup", values(gc) as "Git Clone" by build_number limit=50
```

Our Analysis

How long does each step of a job take?

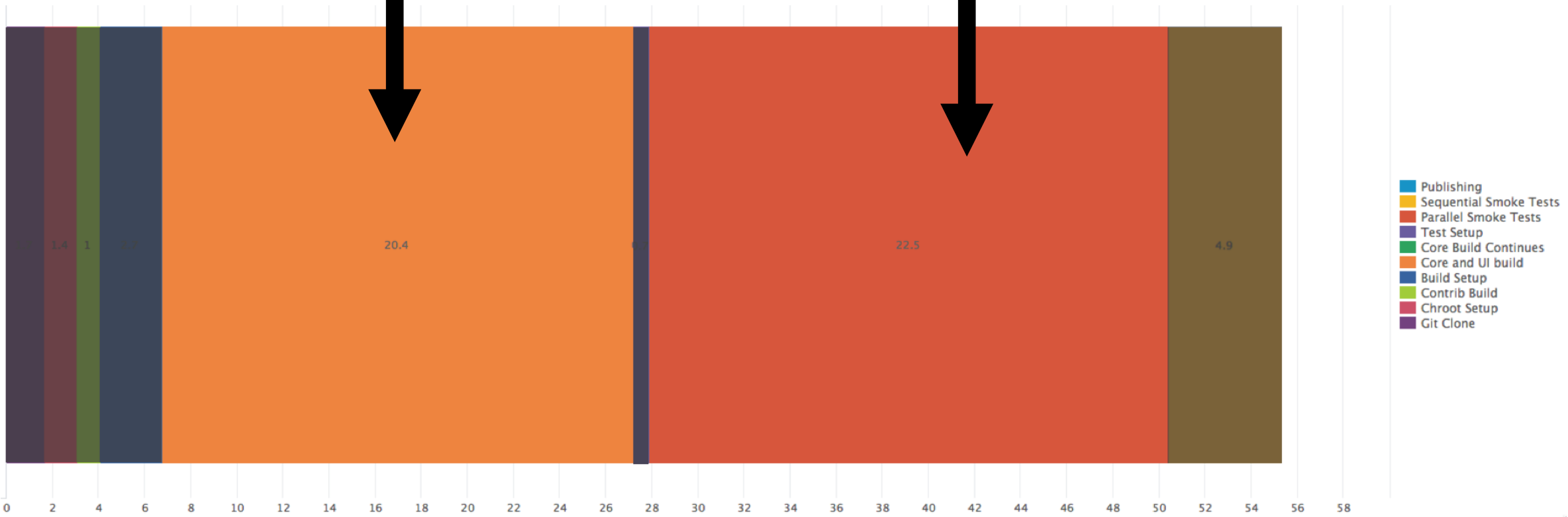
- We used the previous search to chart the time each step took



Our Analysis

How long does each step of a job take?

- We Identified the two portions of the job that took the longest
- The Splunk build (in orange) and the validation tests (in red)



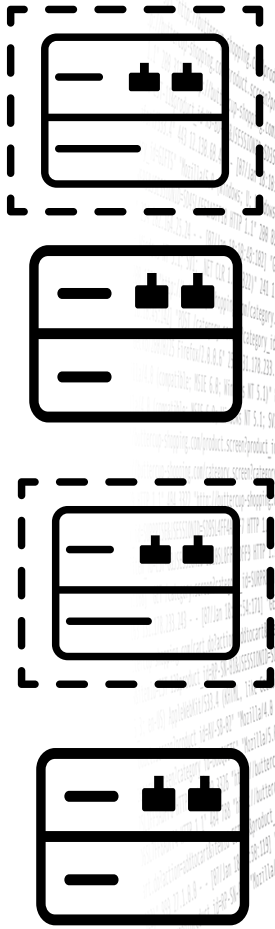
splunk> .conf2017

Speeding up the Splunk Build

splunk> .conf2017

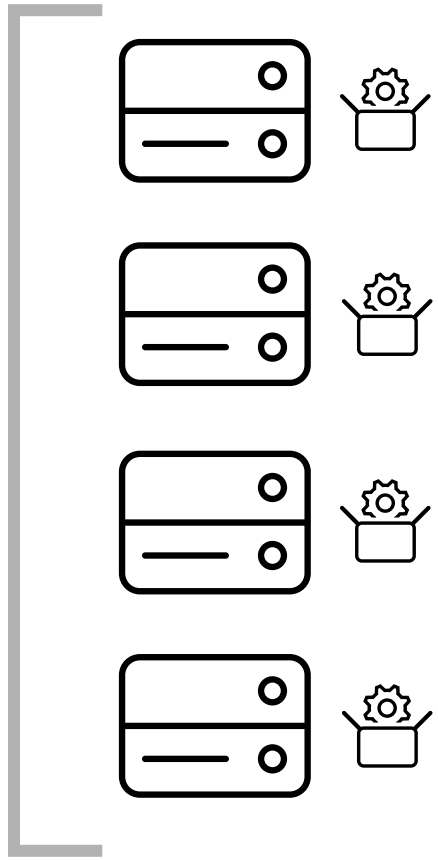
Distcc Architecture

Physical Machines and VMs:



- All DistCC Build Clients and servers use the same build toolchain and chroot
- 15 Build agents
- 12 compile nodes
- DistCC server is used ONLY for compile
- `make -j48`

12 x 20 Core VMs



DistCC VS Normal Build

Building Splunk with DistCC

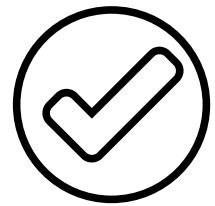
- 24 Min Build
  - `make -j 24`
  - Web UI –j 1
  - Optimal 24 core VM agent

24m →

- 19 Min Build
  - `make -j 48`
  - Web UI –j 1
  - Optimal 24 core VM agent
    - 12 DistCC hosts

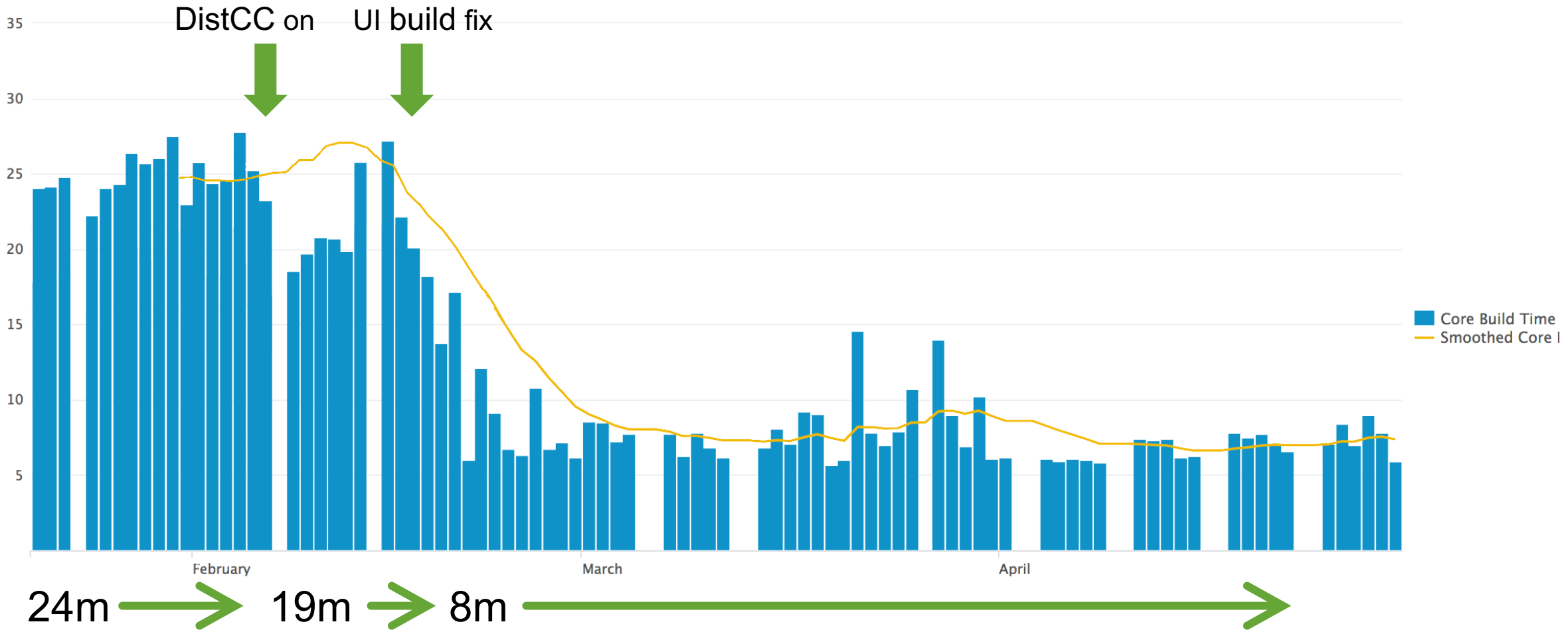
19m →

- 8 Min Build
  - **`make -j 48`**
  - **Web UI –j 6**
  - Optimal 24 core VM agent
    - 12 DistCC hosts

8m → 

Build Time Improvement - Results

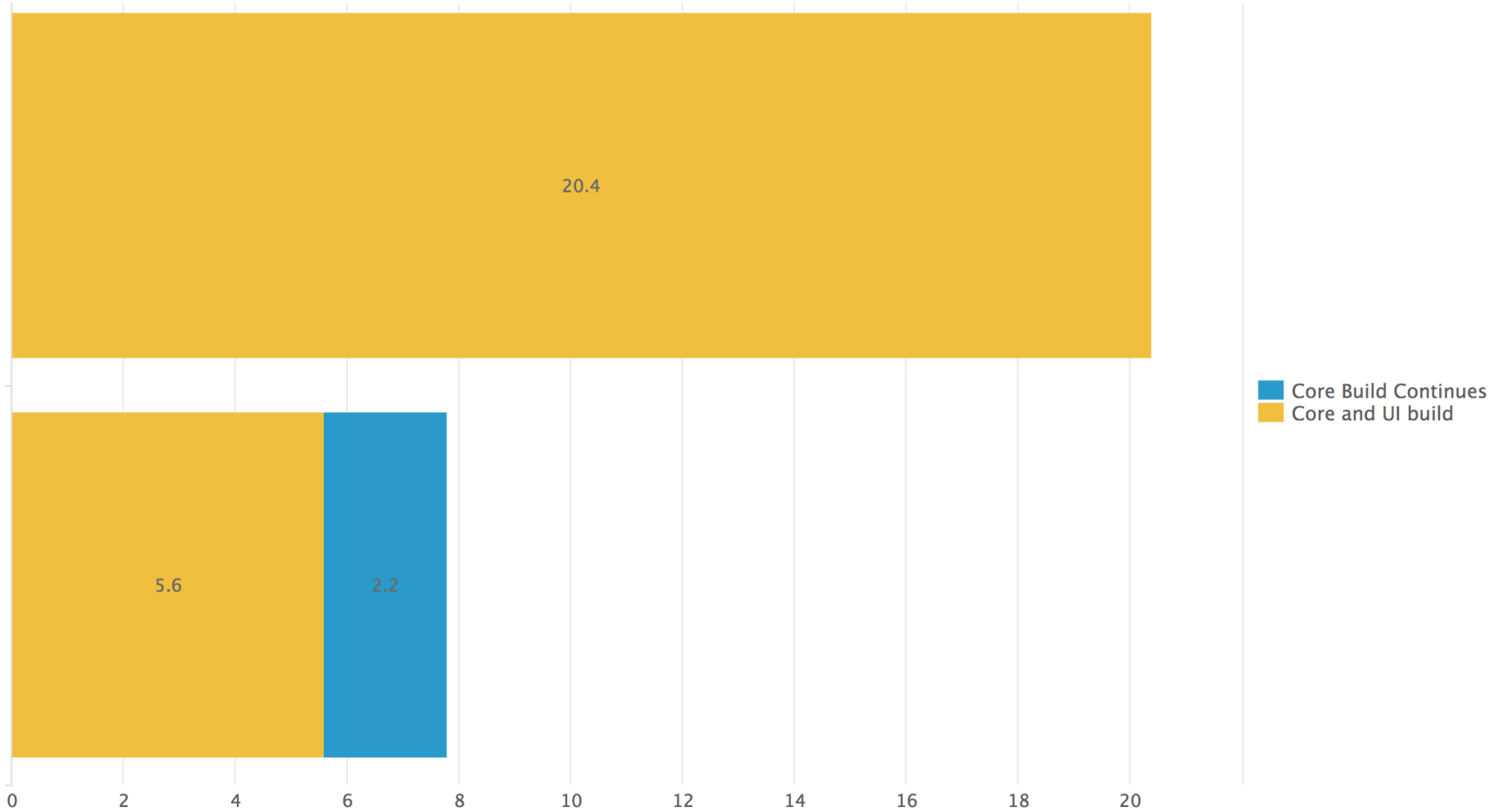
Dramatic reduction in the overall build time



splunk> .conf2017

Build Time Improvement - Results

Dramatic reduction in the core compile time



Speeding up Testing

splunk> .conf2017

Our Analysis

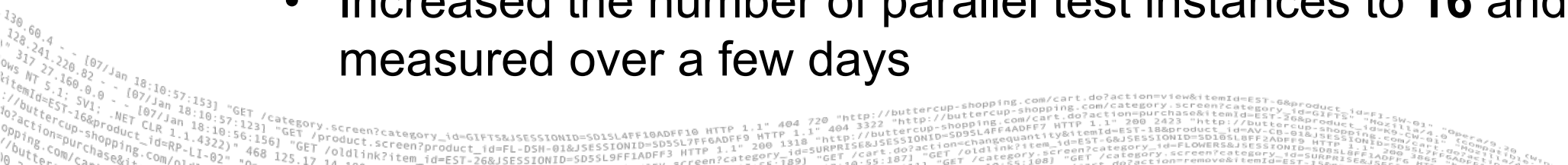
Reducing the Build Testing Time

Analysis:

- We sequentially run two different test configurations on build agents – parallel and sequential
- Parallel tests scheduled across 8 Splunk test instances
- Sequential tests ran one at a time
- We used Splunk to measure the overall timing of each test configuration as well as the individual tests

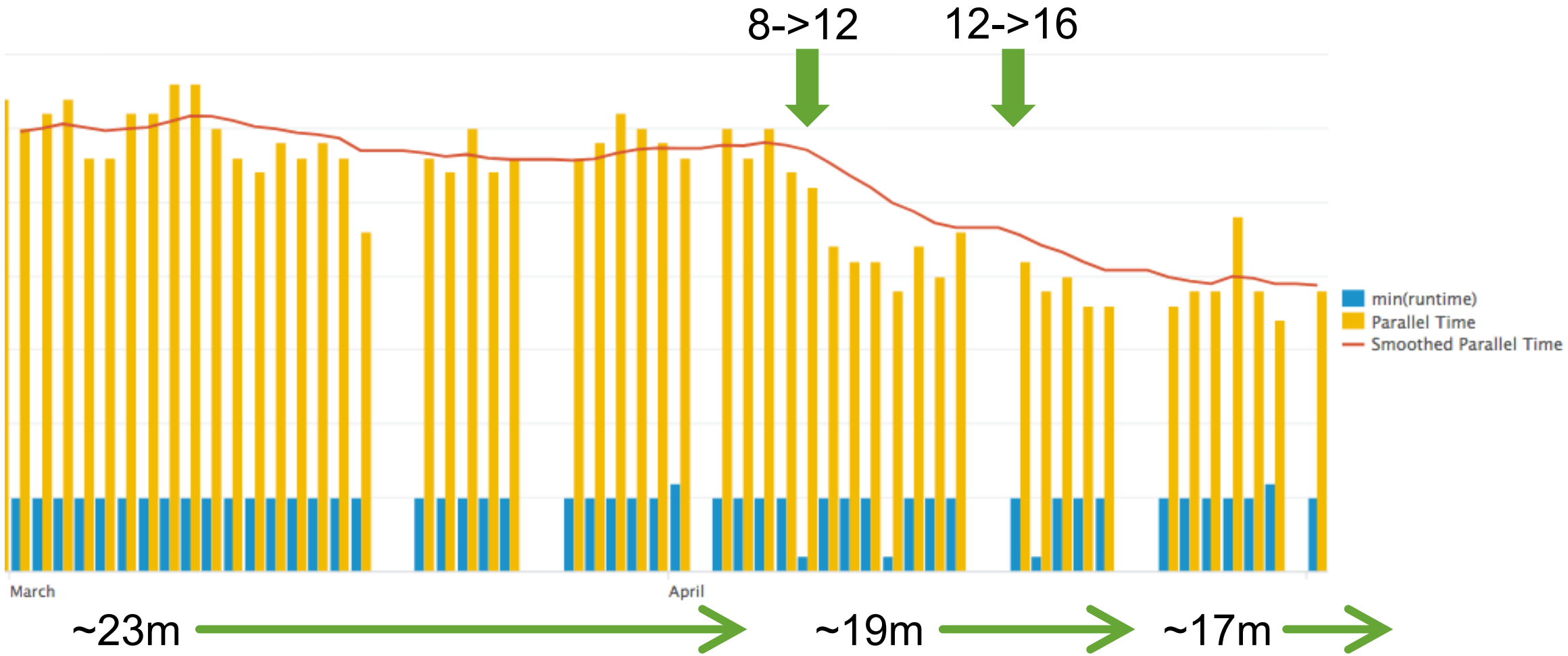
Mitigation:

- Nothing could be done about the sequential tests
- Increased the number of parallel test instances to **12** and measured over a few days
- Increased the number of parallel test instances to **16** and measured over a few days



Increasing Test Parallelization

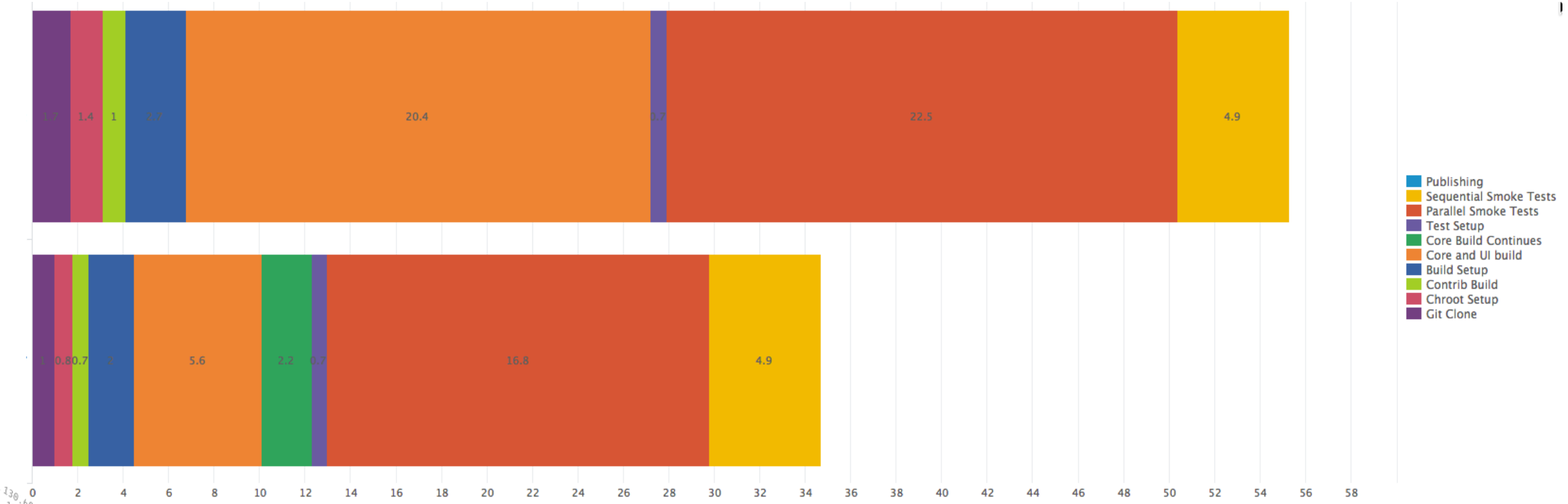
Improvement in the overall test time as parallel instances were increased



The Final Overall Result

Overall job time reduced to ~35 minutes

- Build time reduced to under 8 minutes
- Test time reduced to under 22 minutes



splunk> .conf2017

Managing the Agents

splunk> .conf2017

Our Analysis

Breaking Down The Timing

Reducing the Pull Request Queue time

Analysis:

- Developers are only waiting for results from pull request tests
- Triggers on commits to some branches and all pull requests

Mitigation:

- Use more agents for the pull request tests
- Manage the allocation of agents to tasks -> shift resources to pull requests when the queue starts to climb
- Add 5 “standby agents” with reduced capabilities that are powered up on demand -> smaller footprint on VM hosts

Our Analysis

Breaking Down The Timing

Reducing the Pull Request Queue time

Analysis:

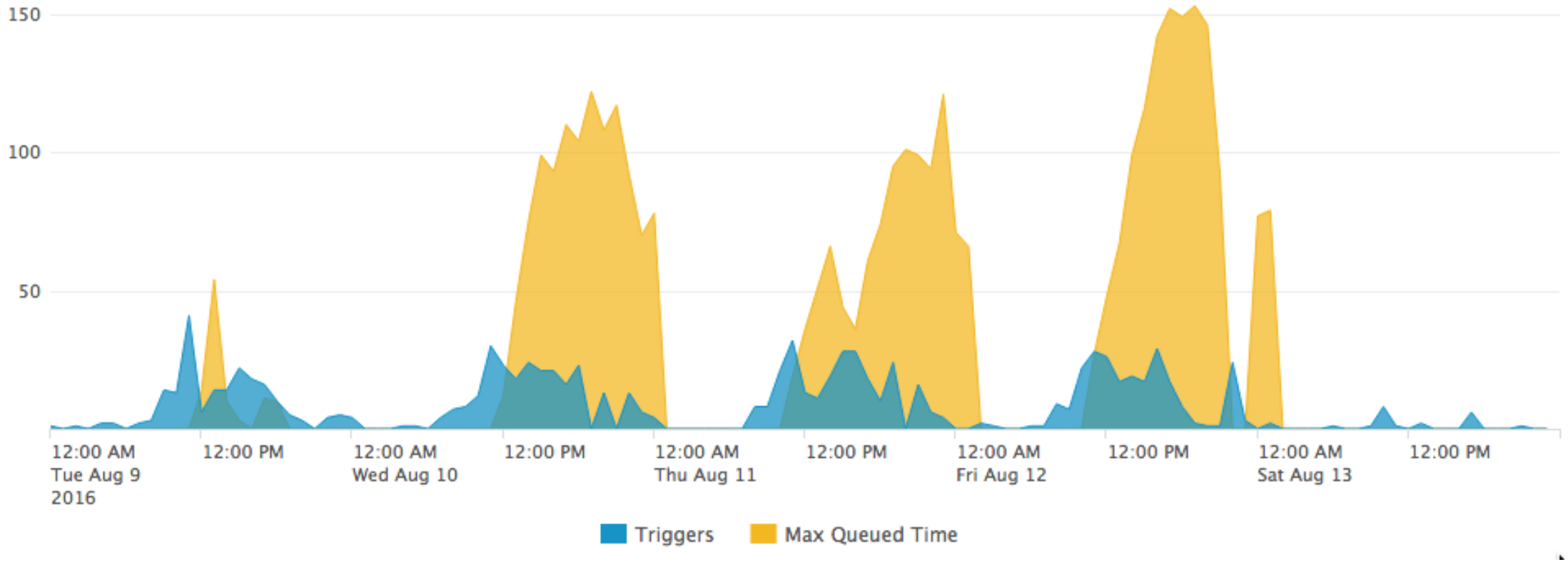
- Developers are only waiting for results from pull request tests
- Triggers on commits to some branches and all pull requests

Mitigation:

- Use more agents for the pull request tests
- Manage the allocation of agents to tasks -> shift resources to pull requests when the queue starts to climb
- Add 5 “standby agents” with reduced capabilities that are powered up on demand -> smaller footprint on VM hosts

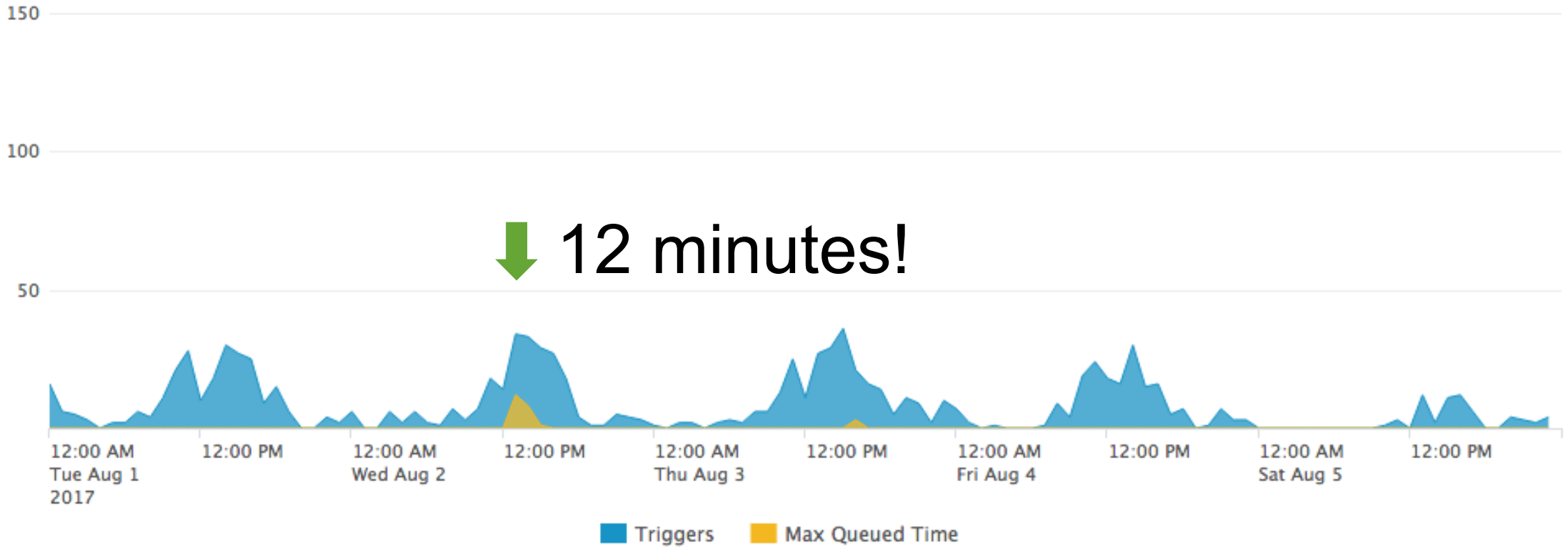
Before Optimization

Peak load period –> significant delays

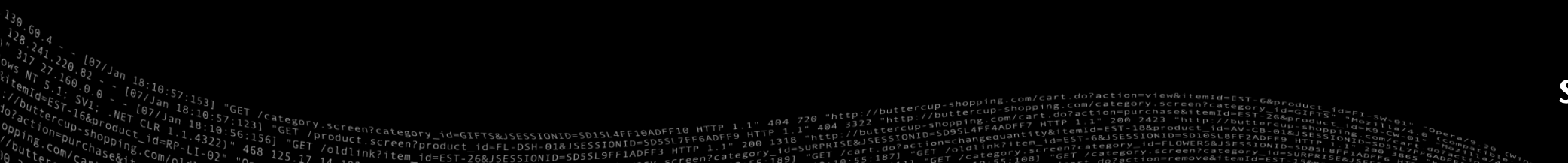


After Optimization

Recent similar trigger conditions



Improving Notifications



splunk> .conf2017

Our Analysis

Speeding up Developer Notification

Analysis:

- Test results only available as comments in pull request UI
- Developers need to go to Bitbucket frequently to check for results
- All employees are connected via Atlassian HipChat

Mitigation:

- Added personal HipChat messages
  - On receipt of trigger –> includes the current queue time
  - When job is completed –> includes test results

Overall Improvement

Key Takeaways

Parallelize Builds

Improved efficiency in builds by optimizing parallelization

DistCC in Build System

Incorporated DistCC in build system

Alerting Developers

Notified Developers about build and merge status

Splunk

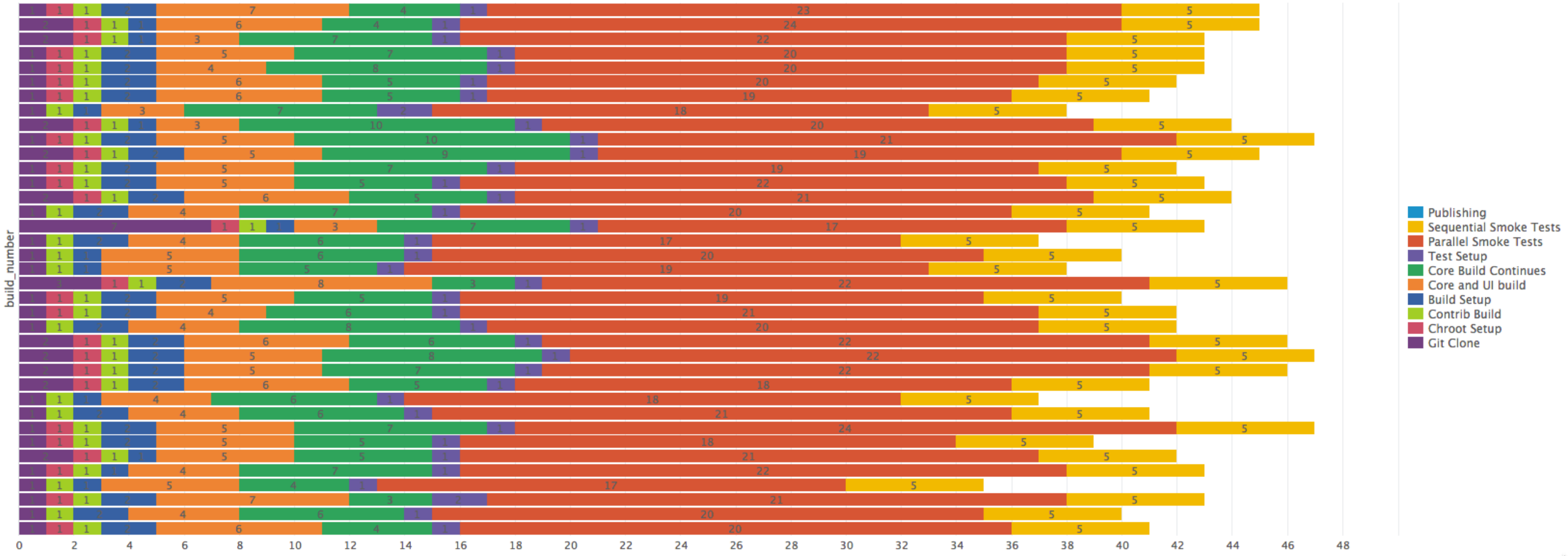
Analysis, Alerting and Reporting

Parallelize Tests

Improved efficiency in test by optimizing parallelization

The Search We Used to Analyze Jobs

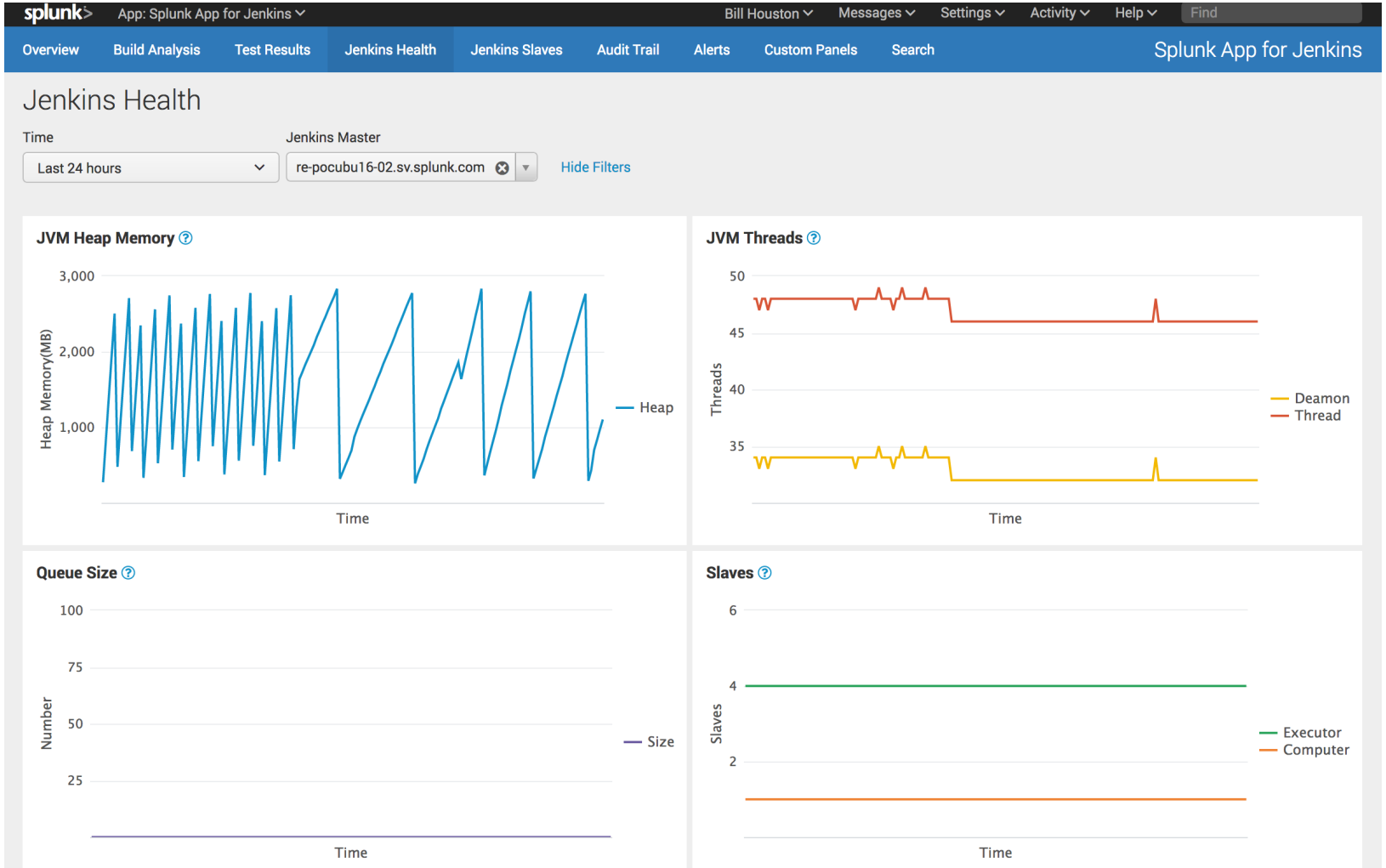
Here is what the full output of the search we presented earlier looks like



splunk> .conf2017

The Jenkins App for Splunk

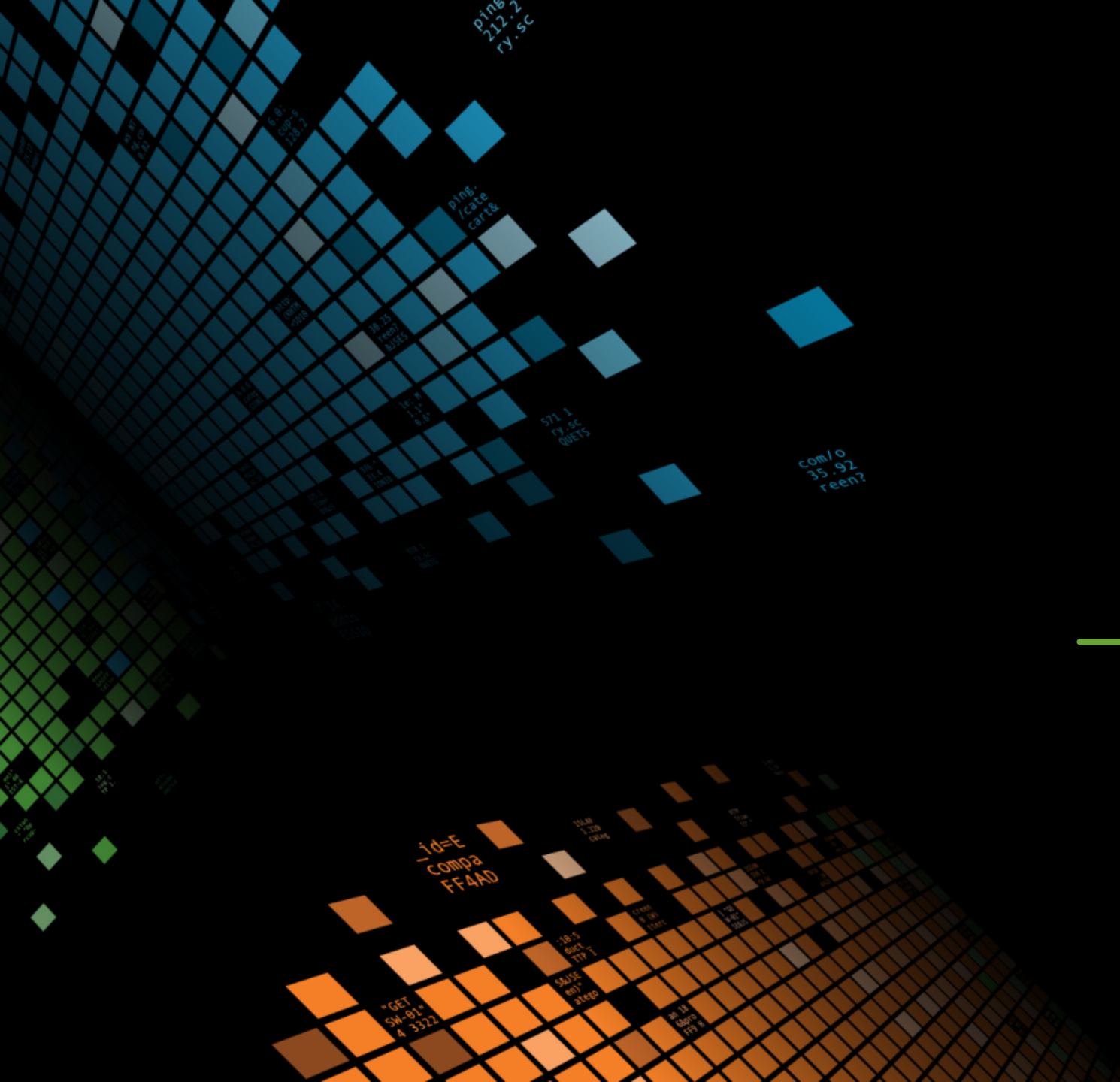
Seamlessly collect, monitor and analyze **Your** Jenkins Data



splunk> .conf2017

Thank You

Don't forget to rate this session in the .conf2017 mobile app



Q&A

Appendix

Some reference for above Deck to know more

splunk> .conf2017

Splunk – Jenkins Configuration - 1

Name	Folder name	Version	Update checking	Visible	Sharing	Status	Actions
Splunk App for Jenkins	splunk_app_jenkins	1.0.4	Yes	Yes	App | Permissions	Enabled	Launch app | Edit properties | View objects | View details on SplunkApps

JaCoCo plugin

No description available.

2.2.1

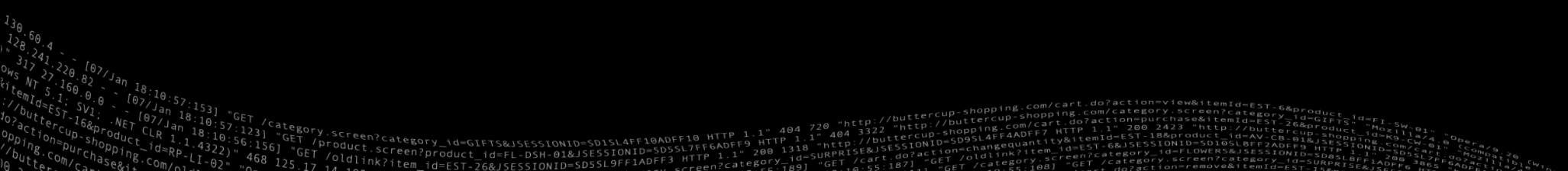
Splunk Plugin

"{color:#474444}Splunk plugin for Jenkins provides deep insights into your Jenkins master and slave infrastructure, job and build details such as console logs, status, artifacts, and an incredibly efficient way to analyze test results. The plugin is used together with a{color} {color:#474444} Splunk App for Jenkins{color} {color:#474444}that provides out-of-the-box dashboards and search capabilities to enable organizations to run a high performing Jenkins cluster and bring operational intelligence into the software development life cycle.{color}"

1.5.1

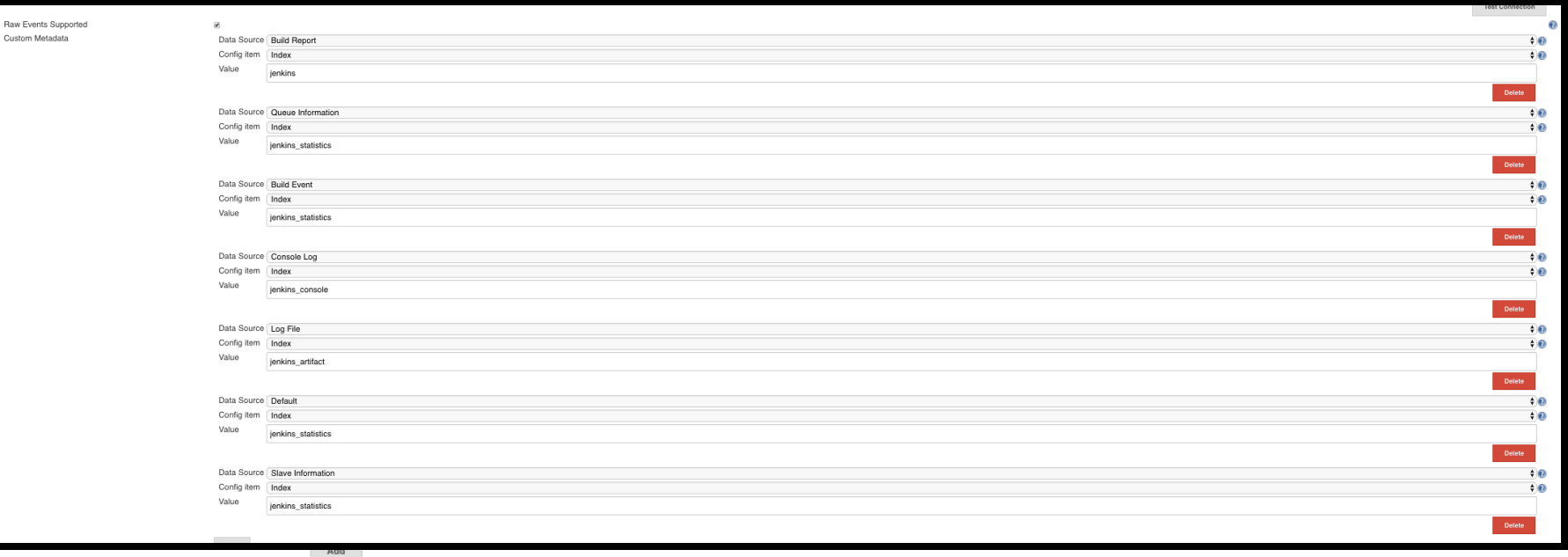
Splunk for Jenkins Configuration

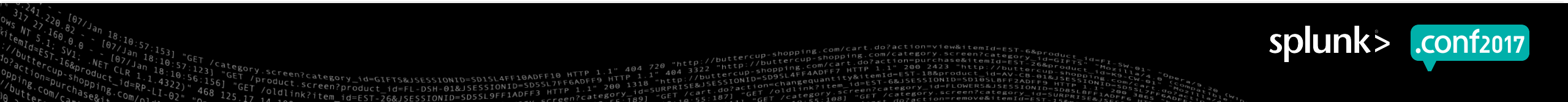
Enable	☑
Indexer hostname	splunk.
HTTP Input Port	443
HTTP Input Token	5765CFBF-9C5C-425A-BF79-BB0C17D110D0
SSL Enabled	☑
Jenkins Master Hostname	mtv-jenkins-qbo



splunk> .conf2017

Splunk – Jenkins Configuration - 2





splunk> .conf2017

Splunk/Jira Validator Flow

POM.XML

```
<dependency>
        <groupId>com.intuit.qbo.qa</groupId>
        <artifactId>SplunkValidator</artifactId>
        <version>1.0.10</version>
</dependency>
```

Test

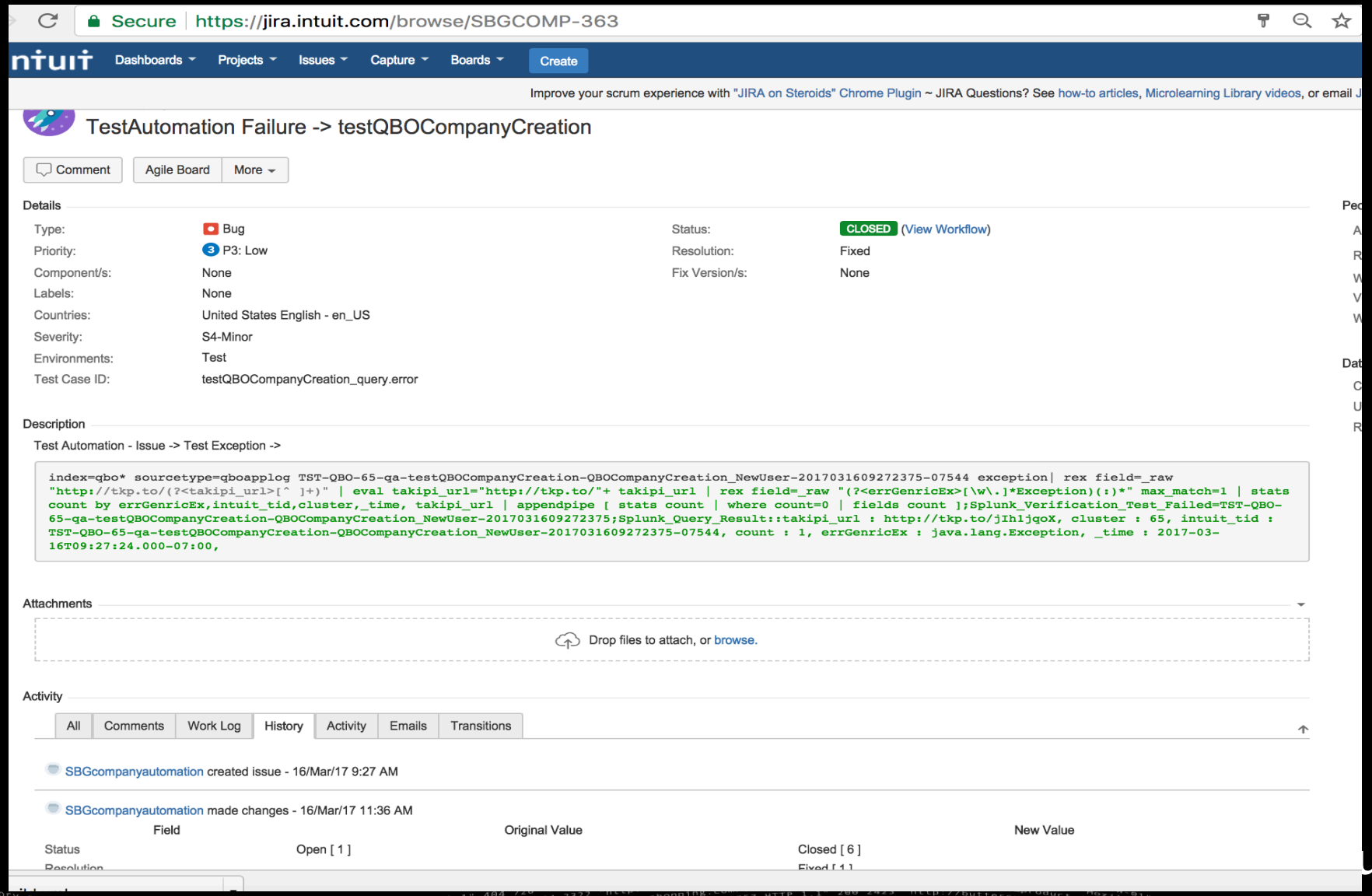
```
@Test(enabled = true)
@SplunkValidator(testFunctionality = "companyCreationforUS", enabledValidation=true, enabledJiraValidation=true, jiraProject="SBGCOMP"
public void USCompanyCreation() throws Exception {
        String i      _     = RuntimeSplunkConfig.getInstance().getTestToken();
        System.out.println("I                                    uit_tid);
```

Configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">
<suite name="Suite">
        <listeners>
                <listener class-name="com.intuit.sbg.splunk.SplunkValidatorListener" />
        </listeners>
        <test name="Test">
                <classes>
                        <class name="healthcheck.CompanyCreationTest" />
                </classes>
        </test>
</suite>
```

splunk> .conf2017

Jira defect



Splunk Configuration

Branch: master ▾ **EnvHealthCheck** / healthcheck / src / test / resources / **splunkConfig.properties** Find file | Copy path

snagal upgrade for latest splunk query 92689cd on Apr 30

2 contributors

9 lines (9 sloc) | 1.74 KB Raw | Blame | History

```
1  env=qa
2  splunk.app.name=intu_sbg_qbo_search
3  delay.splunk.operations.milis=20000
4  USCompanyCreation=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.perf;count:1;er
5  GlobalCompanyCreation=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.perf;count:
6  OBILLCompanyCreation=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.perf;count:1
7  GlobalSubscription=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.perf;count:1;e
8  GlobalItemSubscription=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.perf;count
9  OBILLGetOffersAndSubscription=splunk.query.fci;count:1;error:null;status:true && splunk.query.error;count:0 && splunk.query.per
```

Branch: master ▾ **EnvHealthCheck** / healthcheck / src / test / resources / **splunkQueries.properties** Find file | Copy path

snagal upgrade for latest splunk query 92689cd on Apr 30

2 contributors

6 lines (6 sloc) | 2.01 KB Raw | Blame | History

```
1  splunk.query.fci=index=qbo* $intuit_tid$ sourcetype=access_combined 200 signupnewuser| stats dc(intuit_tid) as count by intuit_
2  splunk.query.perf=index=qbo* sourcetype=qboapplog $intuit_tid$ PerfStat signupnewuser service| stats dc(intuit_tid) as count,ma
3  splunk.query.error=index=qbo* sourcetype=qboapplog $intuit_tid$ "*Exception"| rex field=_raw "\http://tkp.to/(?<takipi_url>[^ ]
4  splunk.query.InternalQBOException=index=qbo* sourcetype=qboapplog $intuit_tid$ InternalQBOException| rex field=_raw "\http://tk
5  splunk.query.CriticalQBOException=index=qbo* sourcetype=qboapplog $intuit_tid$ CriticalQBOException| rex field=_raw "\http://tk
6  splunk.query.RuntimeException=index=qbo* sourcetype=qboapplog $intuit_tid$ RuntimeException| rex field=_raw "\http://tkp.to/(?<
```

Splunk/Jira Validator Flow

```
Splunk_Verification_Test=QBO-65
Delaying splunk query for::10000 mili-seconds
Querying For:QUERY.FCI
SPLUNK QUERY::search index=qbo* sourcetype=qboapplog TST-QBO-65-qa-testQBACompanyCreation-QBACompanyCreation_NewUser-
20170316103616672-23419 signupexistinguser QboAuthFilter httpStatusCode=500  PerfStat| stats count
Time Range: From---15m - To --now
PATH:: /servicesNS/rest_tbarua/intu_sbg_qbo_search/search/jobs
intu_sbg_qbo_search
From Splunk:
   count:  0
Expected::
count :: 0
[Splunk Verification]
```

Successful Test

```
[Splunk Verification]
Mismatch:0, against:2
Mismatch:0, against:3
Mismatch:0, against:1
Mismatch:0, against:1
Splunk_Verification_Test_Failed=TST-QBO-qbo-bats-4-e2e-en_US-testCreateInvoiceFromInvoiceList-TestCreateInvoiceFromInvoiceList
Action=Updated bug comments -> for JiraId=https://jira.intuit.com/browse/SBGCOMP-486
```

Failed Test

```
Querying For:QUERY.ERROR
SPLUNK QUERY::search index=qbo* sourcetype=qboapplog TST-QBO-65-qa-testQBACompanyCreation-QBACompanyCreation_NewUser-
20170316103616672-23419 exception| rex field=_raw "http://tkp.to/(?<takipi_url>[^ ]+)" | eval takipi_url="http://tkp.to/"+ takipi_url
| rex field=_raw "(?<errGenricEx>[\w\.]*Exception)(:)*" max_match=1 | stats count by errGenricEx,intuit_tid,cluster,_time, takipi_url
| appendpipe [ stats count | where count=0 | fields count ]
Time Range: From---15m - To --now
PATH:: /servicesNS/rest_tbarua/intu_sbg_qbo_search/search/jobs
intu_sbg_qbo_search
From Splunk:
   count:  0
Expected::
count :: 0
[Splunk Verification]
Updated / closed jira https://jira.intuit.com/browse/SBGCOMP-364
Splunk Attributes::---------------
```

Closed Test

splunk> .conf2017

Splunk Data uploader -1

```
12:35:30 entire-build-execution: 13 minutes, 27 seconds *** >>> !!! FOCUS ON SPEEDING THIS UP !!! *** <<<
12:35:30 Executing /home/scmbuild/.jenkins/workspace/QBO-Resiliency/Coverage-Daily-Build/build/maven/bin/mvn -quiet -s
/home/scmbuild/.m2/settings.xml -f /home/scmbuild/.jenkins/workspace/QBO-Resiliency/Coverage-Daily-
Build/qbo/build/upload_splunk_data/pom.xml install -DBUILD_KIND=nonDockerCI -DJOB_NAME=Coverage-Daily-Build -DBUILD_NUMBER=45
```

Uploading result After Build

i	Time	Event
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 buildStatus=pass elapsedSeconds=807 host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=807 buildSection=entire-build-execution host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=0 buildSection=finish host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=0 buildSection=do-after-tests host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=0 buildSection=war-and-package host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=0 buildSection=clean-up host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=0 buildSection=do-tests host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=1 buildSection=misc-post-common-reactor host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=782 buildSection=common-reactor host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114
>	7/28/17 12:35:22.691 PM	2017-07-28T12:35:22.691-0700 buildKind=nonDockerCI buildName=Coverage-Daily-Build buildNumber=45 buildVersion=1707.045 level=I elapsedSeconds=24 buildSection=clean-and-prepare host = mtv-ubld-114 source = splunk_data.txt sourcetype = scmbuild_mtv-ubld-114

Result in Splunk

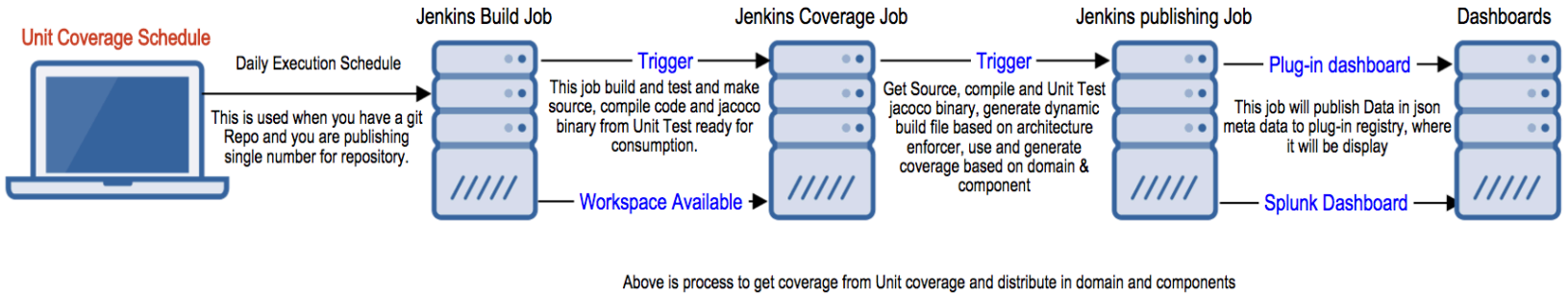
splunk> .conf2017

Splunk Data uploader - 2

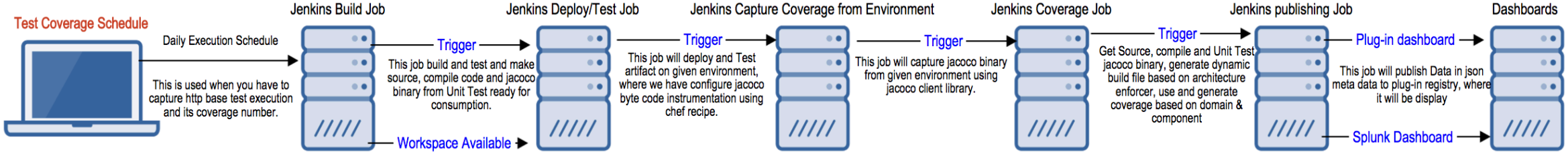
```
<parent>
  <groupId>com.intuit.sb.qbo</groupId>
  <artifactId>qbo-build-modules-list</artifactId>
  <version>17.08.0-SNAPSHOT</version>
</parent>

<artifactId>qbo-build-upload-splunk-data</artifactId>
<packaging>jar</packaging>
<name>QBO Splunk Data Upload Executor</name>

<build>
  <plugins>
    <plugin>
      <groupId>${project.groupId}</groupId>
      <artifactId>qbo-splunk-data-uploader-maven-plugin</artifactId>
      <version>${custom.maven.plugins.version}</version>
      <executions>
        <execution>
          <id>upload</id>
          <phase>install</phase>
          <goals>
            <goal>upload</goal>
          </goals>
          <configuration>
            <rootSourceDir>${src.home}</rootSourceDir>
            <host>${env.SPLUNK_DATA_UPLOADER_HOST}</host>
            <port>${env.SPLUNK_DATA_UPLOADER_PORT}</port>
          </configuration>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
```

Improving Velocity References

Coverage Process



Above is process to get coverage from Unit coverage and distribute in domain and components



Above is process to get coverage from QA, Test and V4 coverage and distribute in domain and components

splunk> .conf2017