

ADD0

ALL DAY DEVOPS

NOVEMBER 12, 2020

German Rodriguez @ H-E-B

The One (Pipeline)



Tell me about yourself

- **Platform Engineering Manager @ H-E-B**
- **20+ years of experience in Software Engineering, Infrastructure and Architecture**
- **Pin collector, football fan, other eclectic hobbies.**



Intro



- **H-E-B – Supermarket Chain**
- **San Antonio, TX**
- **350+ stores, South Texas & Mexico**
- **1000+ Engineers**
- **200+ apps/services**

Initial State – 5 years ago

- **UI-based jobs**
 - Plugin sprawl
 - Jenkins sprawl
 - Difficult to support
 - Global SSH target configuration required
- **Incipient Groovy pipelines**
 - Solving the same problems over and over
 - Lots of support issues
- **Pipeline reconditioning was a primary service**



Common Job Steps

- **Compile & Build (Gradle, npm)**
- **Package (RPM)**
- **Publish (Nexus)**
- **Deploy (SSH)**






The One Pipeline – Iteration 1

- **Groovy pipeline template**
- **Complementary to existing build jobs**
- **A set of files to copy in your project**
- **Instructions to create a pipeline job**
- **Java and npm only**



Iteration 1 – Code Samples

Name	Last commit	Last update
..		
 HEB_WEBAPP.spec	Adding httpd files	4 years ago
 flow.groovy	"Cleanup rpm directory"	4 years ago
 rpmbuild.gradle	Adding httpd files	4 years ago

```

113
114 task copySpecFile(type: Copy, dependsOn: [configRpmName, createRPMToplevelDirs]) {
115     inputs.property "release", appRelease
116     from "."
117     include "${specFile}"
118     into "${rpmTopDir}/SPECS"
119     filter(ReplaceTokens, tokens: [packageRelease: appRelease, packageName: project.rpmName, packageVersion: appVersion,
120 ]
121
122 task buildRPM(type: Exec, dependsOn: [copySources, copySpecFile]) {
123     workingDir "${rpmTopDir}/SPECS"
124     executable "rpmbuild"
125     args "--define", "_topdir ${projectDir}/${rpmTopDir}", "-ba", specFile
126 }
127
128 uploadRpmPackage {
129     doFirst {
130         configRpmName
131     }
132     repositories.mavenDeployer {
133         configuration = configurations.rpmPackage
134         repository(url: "http://...heeb.com/nexus/content/repositories/releases") {

```

```

3 node('linux') {
4
5     //Project variables
6     def credentials = '7da95001-0064-44e3-9cc9-2b40588595ba'
7     def gitBranch = branchName
8     def gitRepo = .....git"
9     def gradleHome = "${tool 'gradle 2.8'}"
10    def project = projectName //passed in from jenkins
11
12    String[] devServers = [.....heeb.com']
13    String[] certServers = [.....heeb.com']
14    String[] prodServers = [.....heeb.com']
15    //End Project variables
16
17    stage concurrency:1, name: 'Prepare ' + project + ' RPM'
18
19    git changelog: false, poll: false, credentialsId: credentials, url: gitRepo, branch: gitBranch
20
21    //build/libs/*.war should be default location
22    step([$class: 'CopyArtifact', filter: 'build/libs/*.war', projectName: project, selector: [$class: 'WorkspaceSelector']])
23
24    withEnv(["JAVA_HOME=${tool 'jdk 1.8.0_65'}", "GRADLE_HOME={tool 'gradle 2.8'}", "TOMCAT_BUILD=7.0.65"]) {
25        if (buildDev == 'true' || buildAll == 'true') {
26            sh "${gradleHome}/bin/gradle -b rpmbuild.gradle -d buildRPM uploadRpmPackage -Pdeploy=dev";
27        }

```

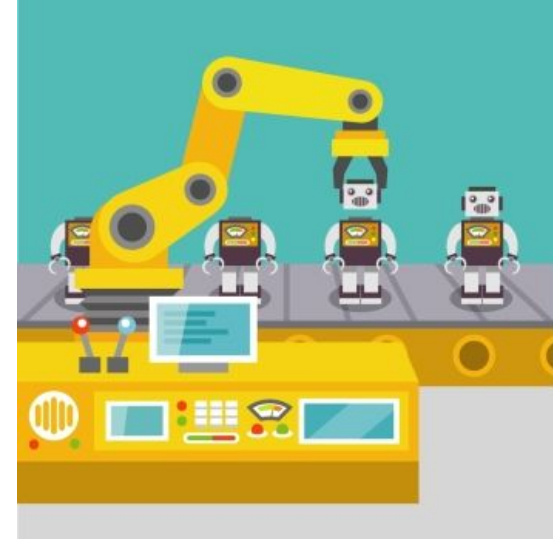
Iteration 1- Learnings

- Popular, widespread use
- Jenkins consolidation
- Faster onboarding, time to initial deployment from 1-2 days to <30 min
- Could not distribute updates or best practices, teams added small changes, drift over time
- Support issues (less) reconditioning abandoned pipelines
- Complex initial setup for the associated services (git deploy keys, Nexus credentials, SSH keys)



The One Pipeline – Iteration 2

- **Opinionated (very) central pipeline generator**
- **Run a (meta) pipeline to create your project pipeline**
- **One way to run, all access tokens pre-configured**
- **K8s targets only**
- **Very strict, no room for deviation**






Iteration 2 – Code Samples

```

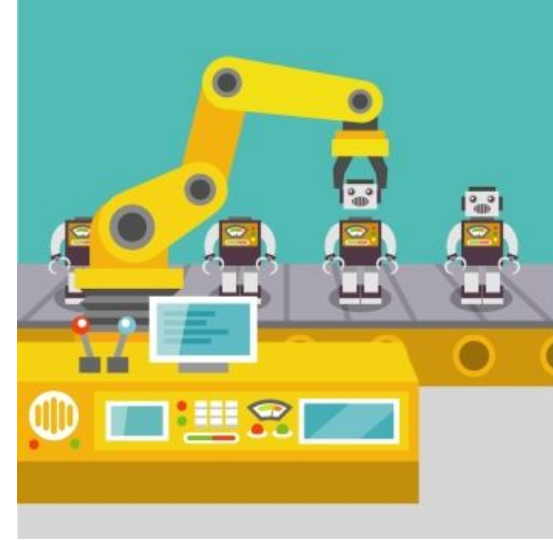
31 stage('Validating environment') {
32     if (!lookup(appTag, '^([a-z][a-z0-9]*)$')) {
33         currentBuild.result = 'FAILURE'
34         error("Bad tag name: " + appTag)
35     }
36
37     if (!lookup(appProfile, '^([a-z][a-z0-9]*)$')) {
38         currentBuild.result = 'FAILURE'
39         error("Bad profile name: " + appProfile)
40     }
41
42     def branchExists = sh (script:"git ls-remote ${gitRepo} | egrep -q refs/heads/${gitBranch}\$", returnStatus: true)
43     if (branchExists != 0) {
44         currentBuild.result = 'FAILURE'
45         error("Bad git repo/branch: " + gitRepo + " => " + gitBranch)
46     }
47
48     //sh "git config daemon.uploadarch true"
49     sh "git archive --remote=${gitRepo} refs/heads/${gitBranch} build.gradle | tar -xf -"
50
51     if (!fileExists('build.gradle')) {
52         currentBuild.result = 'FAILURE'
53         error("Could not find build.gradle on the root of : " + gitRepo + " => " + gitBranch)
54     }
55
56     appName = sh (script: /grep -s -m 1 baseProjectName build.gradle | sed 's#^[^"]*"([^\"]*)".*#\1#'
57     /, returnStdout: true).trim()
58     echo "App Name: " + appName
59     currentBuild.displayName = jenkinsFolder + "/" + appName

```

Name	Last commit	Last update
 Jenkinsfile	Adding gitlab plugin configuration	3 years ago
 Jenkinsfile_template	Fixing spring profile	3 years ago
 job_template.xml	Adding gitlab plugin configuration	3 years ago

Iteration 2 - Learnings

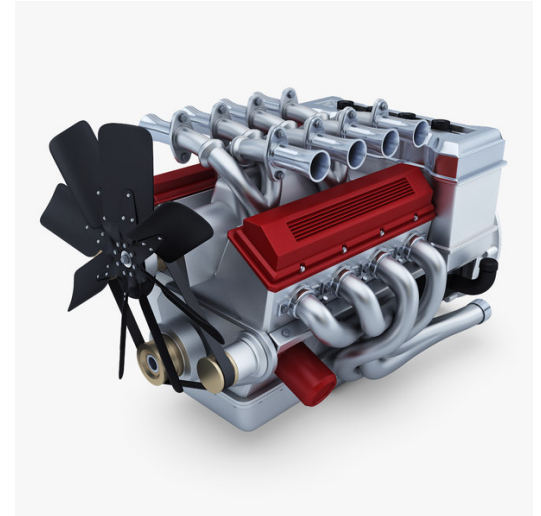
- **Low acceptance (more like zero acceptance)**
- **Abandoned**
- **Some good learnings on container image lifecycle, K8's deployments, templates**





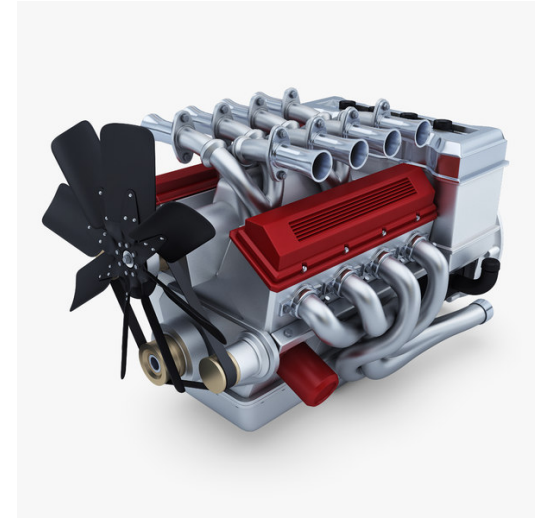
The One Pipeline – Iteration 3 (Current)

- **Shared Jenkins pipeline script, inner sourced**
- **Controlled by parameters and a YAML file in the repository**
- **Multiple languages, packaging and deployment options supported**
 - **Java, Go, Angular, Node, Scala**
 - **Gradle, Maven, npm**
 - **RPM, Docker**
 - **Lighthouse, other testing suites**
 - **Github, Gitlab**
 - **VM, Docker, K8's**
 - **Nexus, Harbor, Artifactory**



The One Pipeline – Iteration 3 (Current)

- **Support for deployments on multiple platforms and environments**
- **Ability to optionally tag git repos back**
- **Versioned, stable & beta tags**
- **All associated tools already integrated using per-tool service accounts (not good)**





Iteration 3 – Job Setup

Pipeline

Definition: Pipeline script from SCM

SCM: Git

Repositories

Repository URL: git@...

Credentials: uni-deployer (uni-deployer gitlab ssh key)

Advanced... Add Repository

Branches to build

Branch Specifier (blank for 'any'): refs/tags/v1.0.0

Add Branch

Repository browser: (Auto)

Additional Behaviours: Add

Script Path: Jenkinsfile

Lightweight checkout:



Iteration 3 – Job Parameters

This project is parameterized

String Parameter

Name

Default Value

Description

[Plain text] [Preview](#)

Trim the string

String Parameter

Name

Default Value

Description

[Plain text] [Preview](#)

Trim the string

String Parameter

Name

Default Value

Description

[Plain text] [Preview](#)

Trim the string

String Parameter

Name

Default Value

Description

[Plain text] [Preview](#)

Trim the string



Iteration 3 – Code Organization

Name	Last commit	Last update
📁 docker	Moving master to cloud	6 months ago
📁 kubernetes	Moving master to cloud	6 months ago
📁 spec	Certification	5 months ago
🔗 .gitignore	Moving master to cloud	6 months ago
📄 Jenkinsfile	Revert "Changing java 8 tool name"	1 week ago
📄 JenkinsfileIOS	Update for removing printing of curl statment in console.	4 months ago
📄 Makefile	Certification	5 months ago
📄 MakefileK8s	Moving master to cloud	6 months ago
📄 README.md	Feature autodeploy cert	1 month ago
📄 prepare.mk	Moving master to cloud	6 months ago
📄 properties.mk.mu	Moving master to cloud	6 months ago
📄 uni-doc.yml	uni-doc for Adding hint of how to use lighthouse	3 months ago



Iteration 3 – Code Sample

```

112 def extractProperties(){
113     def extractPropertiesStageDuration = currentBuild.duration
114     try {
115         stage('Extract properties') {
116             cleanWs()
117
118             gitType = GitRepo.substring(GitRepo.indexOf('@') + 1, GitRepo.indexOf('.'))
119             // Get Nexus Credentials
120             (nexusUser, nexusToken) = credentials.nexusCredentials()
121
122             if (gitType == 'github' && params.GitRepoCredential == null) {
123                 git branch: GitBranch, url: GitRepo, credentialsId: githubCreds
124                 isGitLab = false
125             } else {
126                 git branch: GitBranch, url: GitRepo, credentialsId: sharedFunctions.getGitCredential(params, gitLabCredID)
127             }
128
129             gitLabNotification(isGitLab, "build", "pending")
130             gitLabNotification(isGitLab, "build", "running")
131
132             def deployFileName = 'deploy.yaml'
133             try {
134                 if (DeployFile) {
135                     deployFileName = DeployFile
136                 }
137             }
138             catch (MissingPropertyException e) {
139                 println 'No deploy file specified, using deploy.yaml'
140             }
141
142             properties = readYaml(file: deployFileName)
143
144             properties.put("statistics.buildNumber", env.BUILD_NUMBER)
145             properties.put("statistics.jenkinsScheduled", currentBuild.timeInMillis)
146             properties.put("statistics.jenkinsStarted", currentBuild.startTimeInMillis)

```

Iteration 3 – Repo Configuration Options

```

1 application:
2   name:           # Mandatory, global or general application name
3   serverGroup:   # Server group assigned to the application
4 build:
5   type: angular # Type of build to execute, it can be java_war (default), java_war_ee, java_jar, angular, go, middleware,
6   tool: cli      # Tool that should be used to build, choose from gradle (default), maven, go, angular-cli, sbt (scala)
7   script:       # Build script to execute with the build tool, common choices are: build.gradle (default), pom.xml, main.
8   target:       # Build tool target to execute, the default changes depending on the build type, java_war = bootWar, java
9   javaVersion:  # The Java version used to build a Java project. Valid vaules are 8 and 11. If no value is defined 8 wi
10  toolVersion:  # If project is build with gradle this is the tool version to use. If left empty the verision will be 2.
11  artifactName: # Used in Go project as the name of the source file that will be built when packaging the artifact. Defau
12  workingDirectory: some-new-directory-ui # New - Valid for npm and scala builds only, change the directory w
13  options: -d -b ${buildNumber} ${environment} # NEW Optional, options for shell build script. If left undefined
14  nodeVersion: node 8.15 # Specify the node version to use. Node version must also exist in Jenkins under jenkins.heb.com
15  tasks: # Optional, Task for shell build script, for gradle and maven builds, if left undefined default w
16    - clean
17    - npm install cypress # For angular, angularJS and npm builds extra tasks can be added if needed for execution before
18    - npm install --save-dev start-server-and-test
19    - npm run ci
20  extraBuilds: # Like the build property extraBuilds allows for additinal shell scripts to be run for builds. 0
21    - name: build1 # NEW
22      script: build.gradle # Build script to execute with the build tool, common choices are: build.gradle (default), pom.xml
23      target: bootWar # Build tool target to execute, the default changes depending on the build type, java_war = bootW
24      options: -i -b # Optional, options for shell build script. If left undefined default for gradle will be -d -b,
25      tasks: # Optional, task for shell build script, applies to gradle and maven builds only. If left undefi
26        - clean
27    - name: build2
28      script: build.gradle
29      target: bootWar
30      options: -i -b
31      tasks:
32        - clean

```

```

67 environment:
68 - dev: # Each environment element defines a deployment environment
69   kube: heb.com # Only used for docker packages, ignored otherwise, defaults to the lowest non-prod kubernetes environment
70   namespace: # Only used for docker packages, ignored otherwise, Mandatory, defines the namespace where the package will be deployed
71   replicas: 1 # Only used for docker packages, ignored otherwise, number of replicas to configure in kubernetes, defaults to '1'
72   instances: # Only used for RPM packages, ignored otherwise, the list of servers (instances) where the package will be deployed
73     - heb.com # Fully qualified server name
74   keyFile: non-prod.key # SSH Key or Kubernetes bearer token used by the pipeline to authenticate to the target environment
75   target: # Build tool target to execute, the default changes depending on the build type, java_war = bootWar, java_jar = bootJar,
76   tagGit: # True or False value, when true the properties.service.version and build number will be tagged in the git command whe
77   deployment: dashboard_agent # Used when deploying application, if ssh then keyFile needs to be defined with a valid key also defined.
78   tasks: # Optional, Task for shell build script, for gradle and maven builds, if left undefined default will be -clean, if value
79     - npm install cypress # For angular, angularJS and npm builds extra tasks can be added if needed for execution before actual build command.
80     - npm install --save-dev start-server-and-test #lighthouse can be run as task sh "lighthouse --output html --output-path ./report.html"
81     - npm run ci
82   isDev: yes # Optional - Use isDev, isCert and isProd to differentiate in Nexus between by folder name different environments. The
83   runUnitTests: false # Optional, Used in Java Gradle and Maven builds to run unit tests before building the project when set to true by envir
84 - cert:
85   namespace:
86   kube: heb.com
87   replicas: 2
88   instances:
89     - heb.com
90     - heb.com
91   keyFile: non-prod.key
92   deployment: dashboard_api
93   deploymentApi:
94     slackWebhook: # Slack slackWebhook to notify of deployment failure
95     email: _heb.com # Email to notify of deployment failure
96     retries: 3 # Amount of times to retry
97   tasks: # Optional, Task for shell build script, for gradle and maven builds, if left undefined default will be -clean, if value
98     - npm install cypress # For angular, angularJS and npm builds extra tasks can be added if needed for execution before actual build command
99     - npm install --save-dev start-server-and-test
100     - npm run ci
101   runUnitTests: false # Optional, Used in Java Gradle and Maven builds to run unit tests before building the project when set to true by envir
102 - prod:
103   namespace:
104   kube: heb.com
105   replicas: 2
106   instances:
107     - heb.com
108     - .heb.com
109   keyFile: prod.key

```

Iteration 3 – Typical Configuration

```

1  application:
2     name: ██████████ # Mandatory, global or general application name
3     serverGroup: ██████████ # Server group assigned to the application
4  build:
5     type: go # Type of build to execute, it can be java_war (default), java_jar, angular, go, middleware
6     tool: go # Tool that should be used to build, choose from gradle (default), maven, go, angular-cli
7     script: main.go # Build script to execute with the build tool, common choices are: build.gradle (default), pom.xml, main.go, some build tools do not
8     target: main.go # build tool target, should produce the deploy artifact, WAR or JAR for java apps, TGZ for static content, RPM for make builds
9  package:
10     type: rpm # How to package the app, choices are: rpm (default), docker, combined
11     architecture: x86_64
12  service:
13     name: heb-bootstrap # Mandatory, the name of the service to build and package
14     version: 1.0.0 # Mandatory, version assigned to the build
15     buildDir: ██████████ # Directory where the build artifact will be located after a successful build, defaults to 'build/libs'
16     tagGit: no # Should the pipeline tag the git repo with the version plus the build number, defaults to 'yes'
17  registry:
18     - name: nexus2
19       host: ██████████
20       type: nexus2
21       repository: releases
22  environment:
23     - name: cert
24       instances:
25         - ██████████ heb.com
26         - ██████████ heb.com
27       repository: heb-app-releases
28       deployment: ██████████ agent
29  release:
30     - prod:
31       branch: master

```



Iteration 3 – Stage Run

- Up
- Status
- Changes
- Build with Parameters
- Delete Pipeline
- Configure
- Move
- Support
- Full Stage View
- Job Config History
- Rename
- Schedule Build
- Pipeline Syntax
- Set Next Build Number

Pipeline

Full project name: [...](#)

[Recent Changes](#)

Stage View

Average stage times:
(Average full run time: ~5min 34s)

	Extract properties	Build Artifact	Prepare Packaging	Package artifact	Publish artifacts	Tag Git Repo	Clean Workspace
	29s	1min 9s	5s	1min 22s	4s	28s	697ms
#267 Oct 06 12:49 3 commits	34s	58s	4s	1min 15s	2s	26s	627ms
#264 Oct 06 11:55 1 commit	29s	1min 18s	5s	1min 16s	2s	25s	662ms
#263 Oct 06 10:34 No Changes	38s	1min 6s	5s	1min 17s	2s	31s	650ms
#262 Oct 05 10:39 18 commits	51s	1min 34s	11s	1min 16s	4s	51s	731ms
#261 Oct 02 10:39 1 commit	50s	1min 34s	11s	1min 16s	4s	51s	731ms

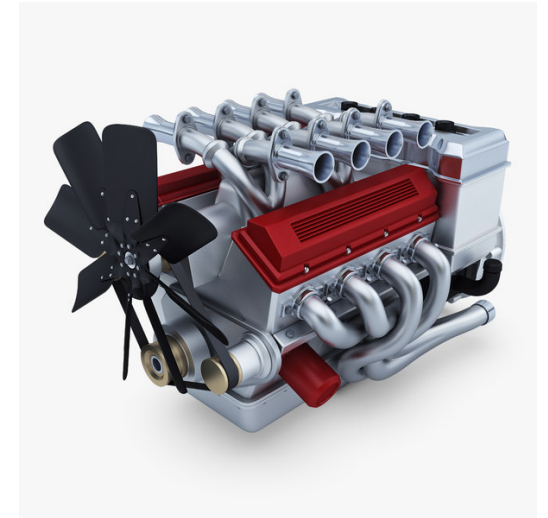
Build History [trend](#)

find

- #267 Oct 6, 2020 12:49 PM TEST - Fixing Vault
- #264 Oct 6, 2020 11:55 AM TEST - Fixing Vault
- #263 Oct 6, 2020 10:34 AM TEST - Debugging Vault
- #262 Oct 5, 2020 10:39 AM PROD ready

Iteration 3 – Expansion

- **Security Scans**
- **Slack Integration**
- **Multi-step builds**
- **Integration with our PaaS offer**
 - **Self-service pipeline setup, including creation of accounts and tokens on associated tools**
 - **Different deployment models (concurrency options)**
 - **Rollbacks**
 - **Secret injection**
 - **Instrumentation (Audit logs, metrics)**



Iteration 3 - Instrumentation

Today Builds
Since 12 Oct 19:00 CDT

154

Today Builds

This Weeks Builds
Since 11 Oct 19:00 CDT

304

This Weeks Builds

Builds Last Week
Since 04 Oct 19:00 CDT until 11 Oct 19:00 CDT

848

Last Weeks Builds

Yesterday Builds
Since 11 Oct 19:00 CDT until 12 Oct 19:00 CDT

150

Yesterday Builds

Number of Builds Over Last Month Compared to a ...
Since 30 days ago, compared with 1 week earlier

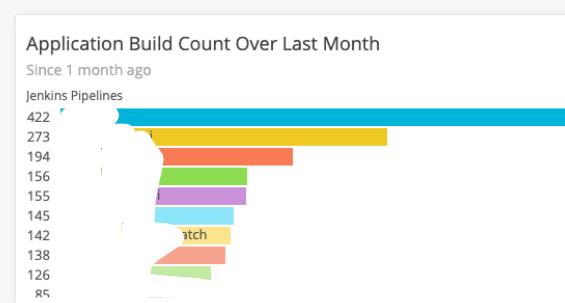
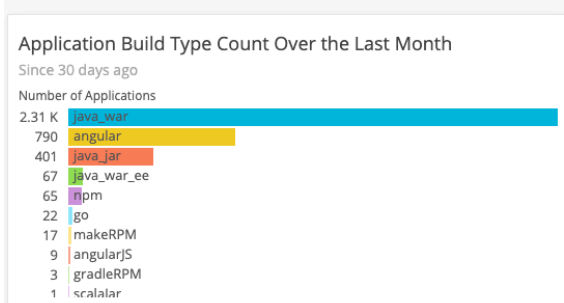
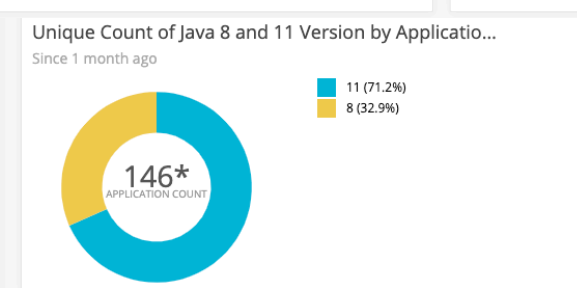
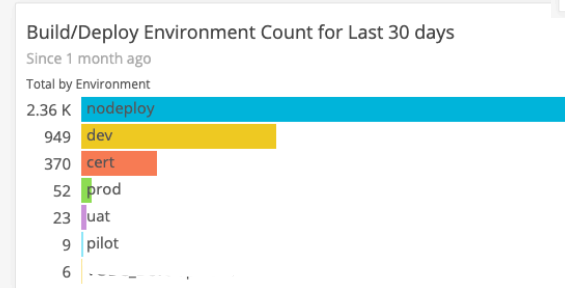
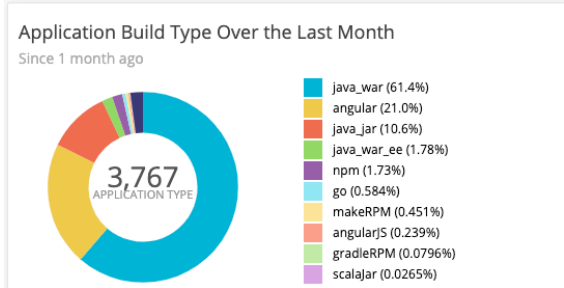
3.76 K

↓1.70%

This Months Builds

Build/Deploy Environment for Last 30 days
Since 1 month ago

3,767 TOTAL # OF ENVIRONMENTS



Unique Count of Java Version by Application for this...
Since 1 month ago

BUILD JAVA VERSION	APPLICATION COUNT
11	104
8	48

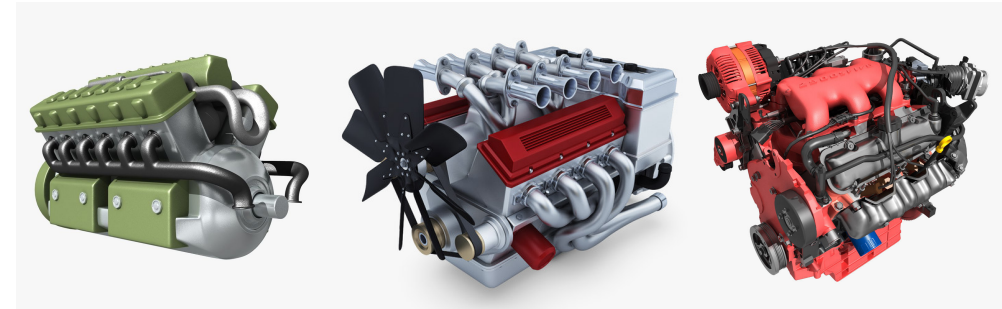
Iteration 3 - Learnings

- High acceptance, low drift
- With self-service new pipelines are ready in <5 min
- Updates are much easier to integrate
- Great collaboration, contributions by multiple teams
 - IOS, Python, R
- Flexible, still embeds our best practices
- We hit the “Method too large error”
<https://support.cloudbees.com/hc/en-us/articles/360039361371-Method-Code-Too-Large-Error>
- Jenkins node and memory management is challenging
- Pipeline starts to look like Gitlab CI or Circle CI declarative models, still allows for a quick setup, consistent best practices, updates, specific checks, support



Iteration 4?

- Refactor, simplify our code base
- Port engine to other CI tools (Gitlab CI?, GitHub Actions?), take advantage of native features
- Looking into the Waypoint model as a potential common platform



Key takeaways

- **Shared pipelines accelerate time to deployment, reduce technical debt, provide valuable insight into the engineering process**
- **Flexibility and transparency increases the chances of success**
- **Repeated best practices and innovation can be accomplished while abstracting common concerns**
- **Unknown unknowns decrease when the places where change needs to be effected and the cognitive load on teams is reduced**

Q&A

Reach me:



<https://twitter.com/callmegar>



<https://www.linkedin.com/in/callmegar/>

THANK YOU TO OUR SPONSORS

