DevOPS: CI/CD Platform at Fresenius

October 15, 2019

Mani and Mohammed

FRESENIUS
MEDICAL CARE

Introduction

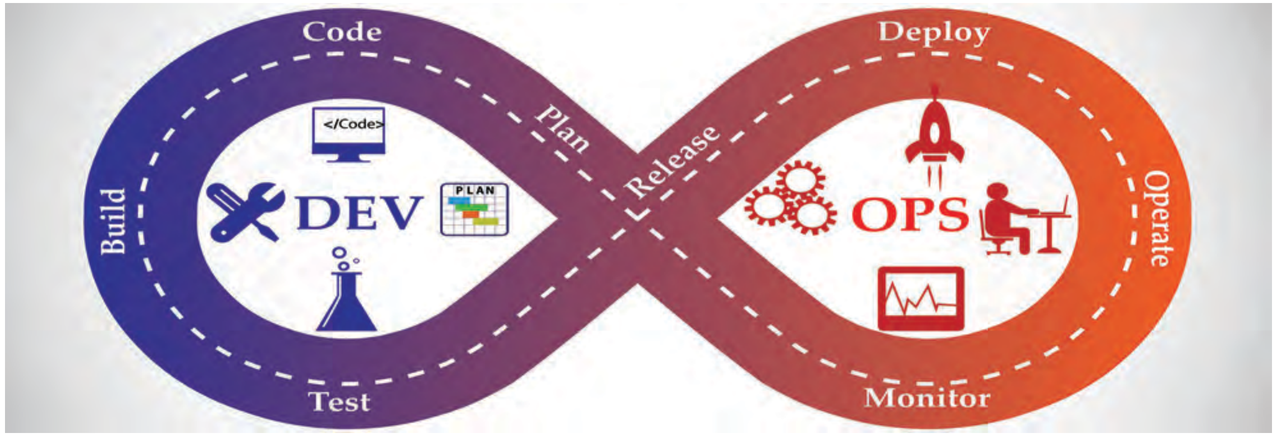
- We are all innovators at Fresenius
- We are all talking about DevOps, Are we doing something about it?
- The middleware team is bridging the gap between Developers and Infrastructure teams and many other groups.
- In the last couple of months, Lots of automation & innovations happens behind the scene.
- Let’s tell you how we are building pipeline; this will show you one of the many things that this team do for the company.

1462

Foundation of the Hirsch Pharmacy in Frankfurt am Main. The Fresenius family assumes ownership in the 18th century.

2019

By acquiring NxStage Medical, Inc., Fresenius Medical Care expands its product portfolio with an innovative technology for home dialysis offering patients greater flexibility and more individual treatment.



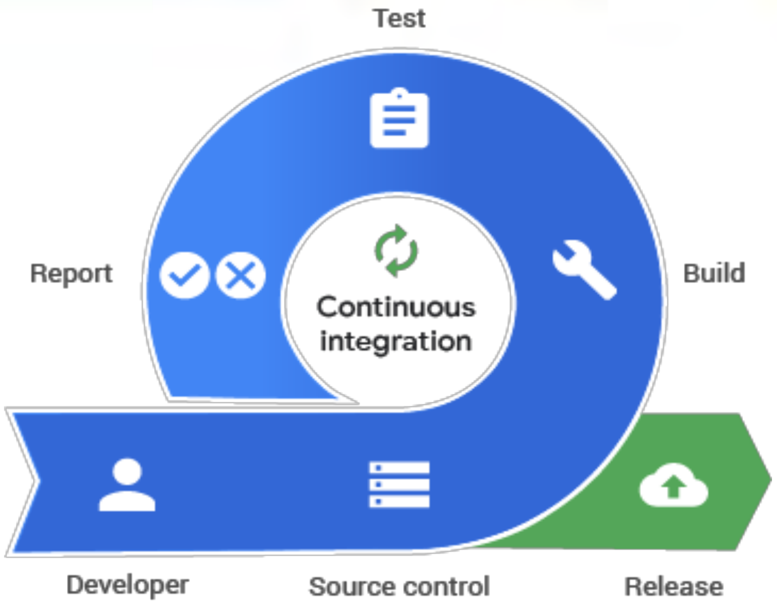
What is CI / CD

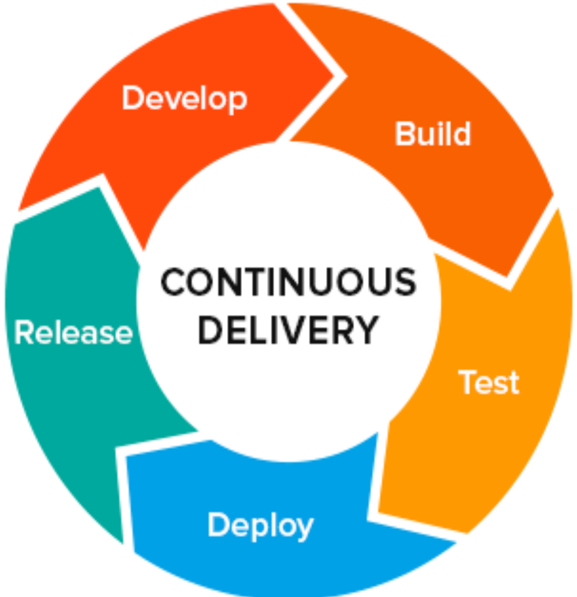
- CI/CD pipelines are designed for businesses that want to improve applications frequently and require a reliable delivery process.

- Continuous Integration (CI) is a set of practices that drive teams to implement small changes and check in code to the version control system.

- The technical Goal of CI is to establish a consistent and automated way to build package and test applications.

- Continuous Delivery (CD) picks up where CI ends. CD automates the delivery of applications to selected infrastructure environments.





Jenkins

- We use Jenkins as our primary tool for CI/CD Process.
- Jenkins is an open source automation Server which enables DevOps team the ability to build, test and deploy apps to multiple environments.
- Jenkins is the most popular CI tool.
- Jenkins at Fresenius manages the Dev and Ops side of Devops from Source Code Management to delivering code to Production.
- Fresenius has 3 different flavors of Jenkins setup and managed by Middleware

Where & How are we using Jenkins now?

- Process Automation
- CI/CD – Pipeline Build
- End to End Pipeline for Non-Containerized & Container applications

Process Automation for Non-Production systems

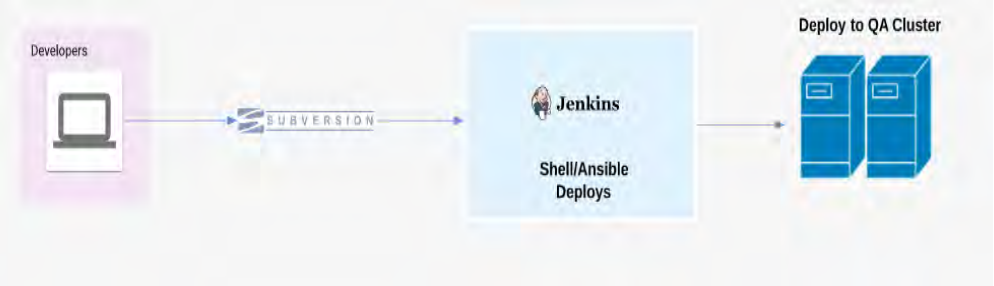
Process Automation Non-Prod Architecture

- ❑ Process Automation Jenkins allows Development teams to deploy to QA using blanket CRQ Approval process and is used across Fresenius by all development teams.

- ❑ Gives Development the ability to execute Operational activities to QA without any delay when a defect is raised by QA.

- ❑ Process Automation Jenkins has 500 Operational Jobs in Non-Prod and are currently in use by Dev team.

- ❑ On an Average 50 Jobs are being executed daily



[redacted]
and GTO teams

Process Automation Prod

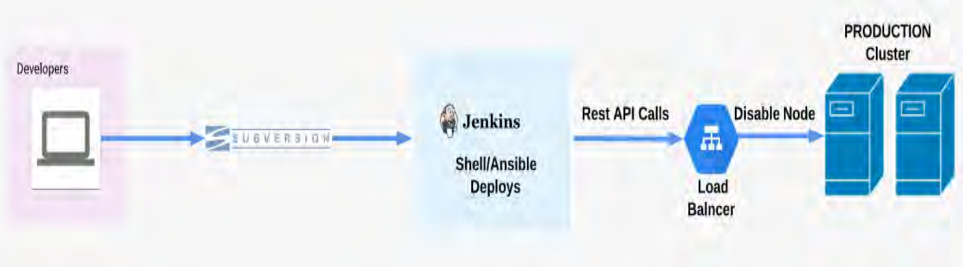
- Process Automation Prod Jenkins allows Development teams to request Prod deployments off hours to avoid disruptions of clinic operations

- Some of the Production jobs are deployed as Zero downtime using a Load balancer automation process which allows node to be taken out of a pool before deployment and added back after deployment

- Off hours Prod deployments are being executed by Operations support team using Jenkins interface provided by Middleware as they are staffed 24/7

- It is used across Fresenius by all development teams

- Process Automation Jenkins has 200 Operational Jobs(Restart and Deployment) in Prod

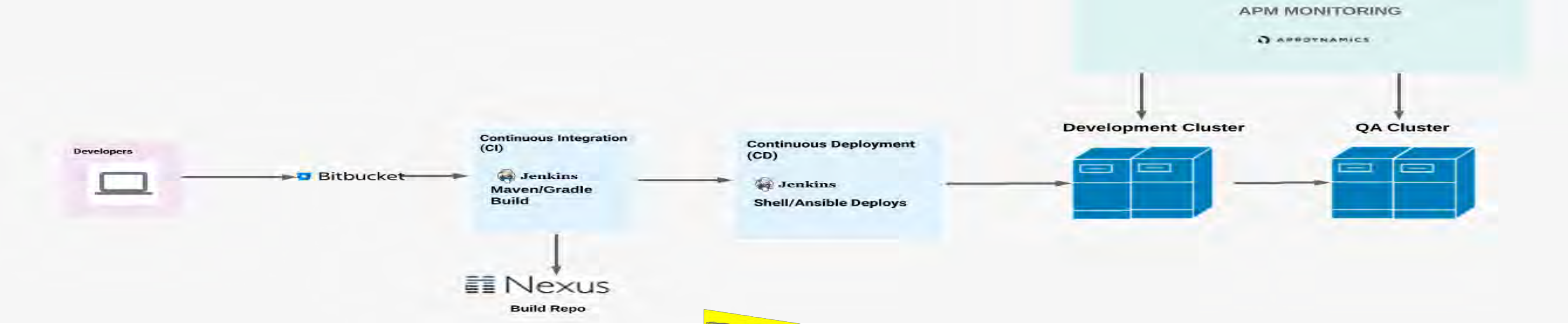


[redacted]

and GTO teams

CI/CD Architecture

- CI/CD Jenkins is the initiative of Implementing Continuous Integration and Continuous Deployment at Fresenius allowing Code to be compiled/build and deployed using Jenkins .
- It allows Middleware to build and compile code using various integrated build automation tools .
- It also allows to stage the Fully build artifacts to staging repositories and deploy to Dev and QA without any Manual Intervention



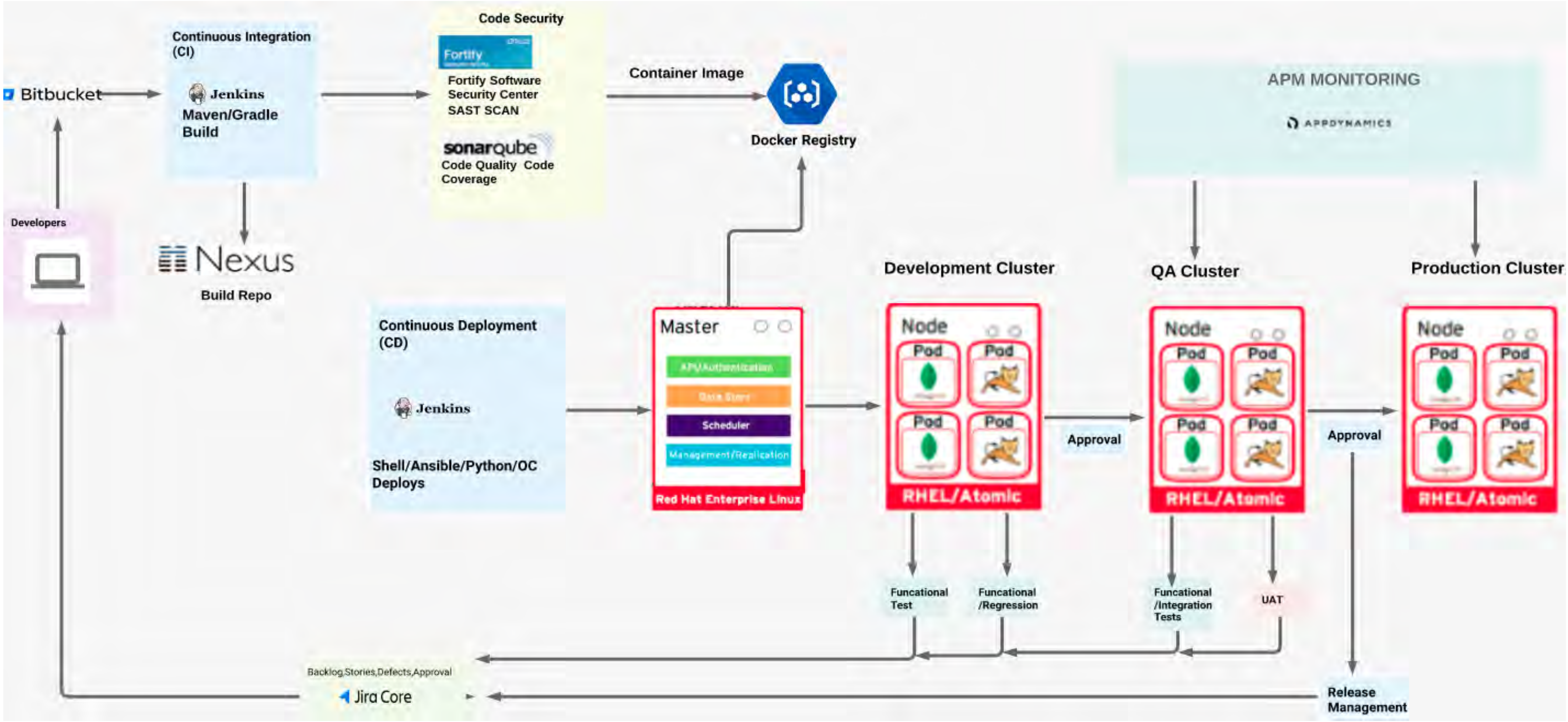
[illegible]
Portal and Integrations teams

End to end Pipeline

- Pipeline Implementation of Continuous Integration and Continuous Deployment at its full capacity allowing Code to be build, test , scan (Security Vulnerabilities, Code quality) and deploy
- Automated pipelines remove manual errors, provide standardized development feedback loops and enable fast product iterations.
- End to End pipeline helps you automate steps in your software delivery process, such as initiating code builds, running automated tests, and deploying to a staging or production environment
- Currently We are using this for Ambulenz & ePA for Clinical applications; Health cloud request services for Integrations team

salesforce

End to End CI/CD Pipeline Architecture for Containers



Bitbucket

Continuous Integration (CI)

Jenkins

Maven/Gradle Build

Code Security

Fortify

Fortify Software Security Center SAST SCAN

sonarqube

Code Quality Code Coverage

Container Image

Docker Registry

APM MONITORING

APPDYNAMICS

Developers

Nexus

Build Repo

Continuous Deployment (CD)

Jenkins

Shell/Ansible/Python/OC Deploys

Master

API/Authentication

Data Store

Scheduler

Management/Replication

Red Hat Enterprise Linux

Development Cluster

QA Cluster

Production Cluster

Node

Pod

Pod

Pod

Pod

RHEL/Atomic

Approval

Approval

Funcational Test

Funcational /Regression

Funcational /Integration Tests

UAT

Backlog,Stories,Defects,Approval

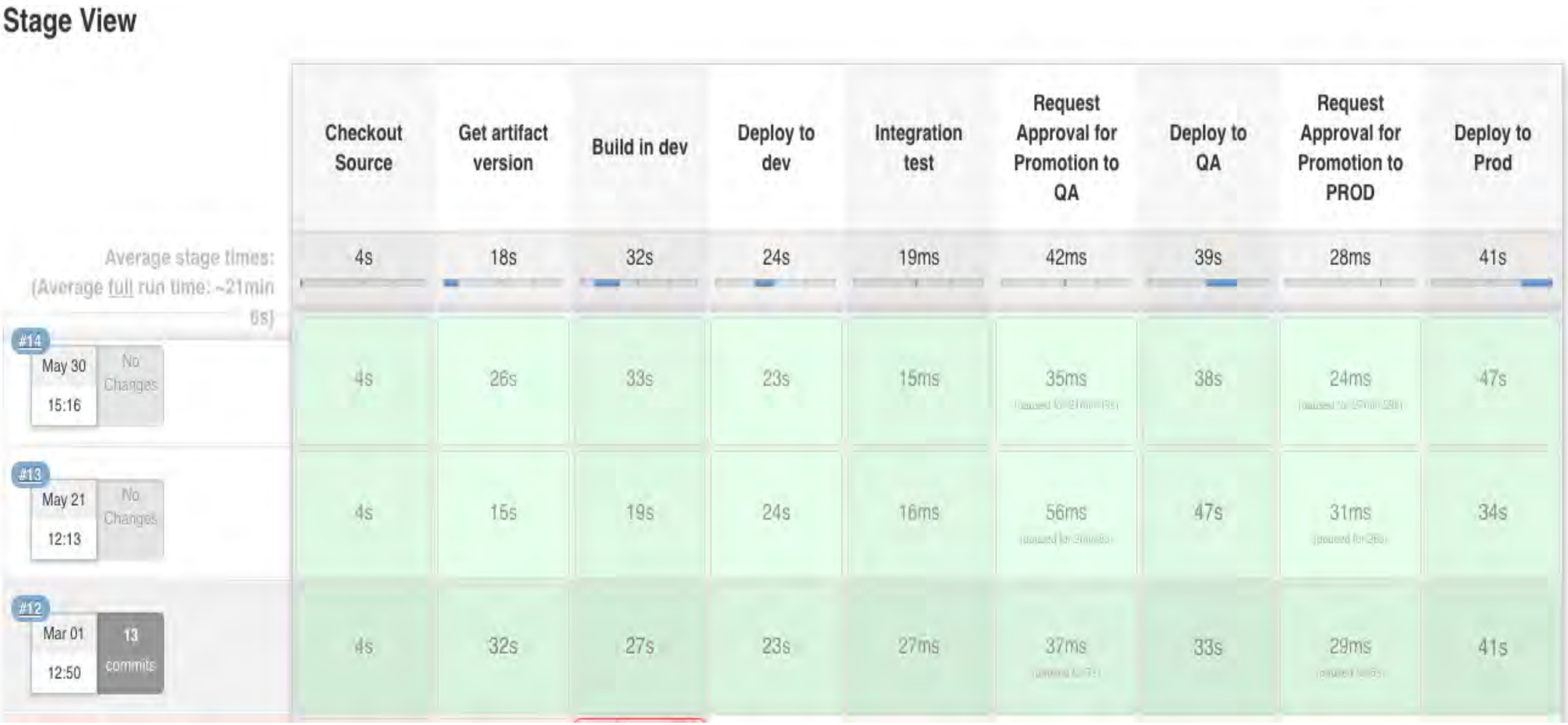
Jira Core

Release Management

Pipeline Steps and Fresenius IT team handshake

- Check source code from version control system:
  - Bitbucket – Clinicals and Integrations team
  - Gitlab – UI/UX
  - SVN – All teams
- Build / Compile using build automation tool
  - Build Archives: war, ear, jar and Deploy
  - Maven: Clinicals, BI, Integrations and Portal teams
  - Gradle: Portal for Doctors Corner
  - ANT: MW uses to deploy to WebLogic
- Stage fully build artifact to staging repository
  - Nexus antifactory: Clinicals team
  - SVN : Integrations team
- Security & Code Scanning:
  - SonarQube: Code Quality Scanning & Code coverage
  - Fortify: As a Static application security Testing(SAST)
- Deploy
  - Bash Shell: All teams
  - Ansible: Clinicals & BI team

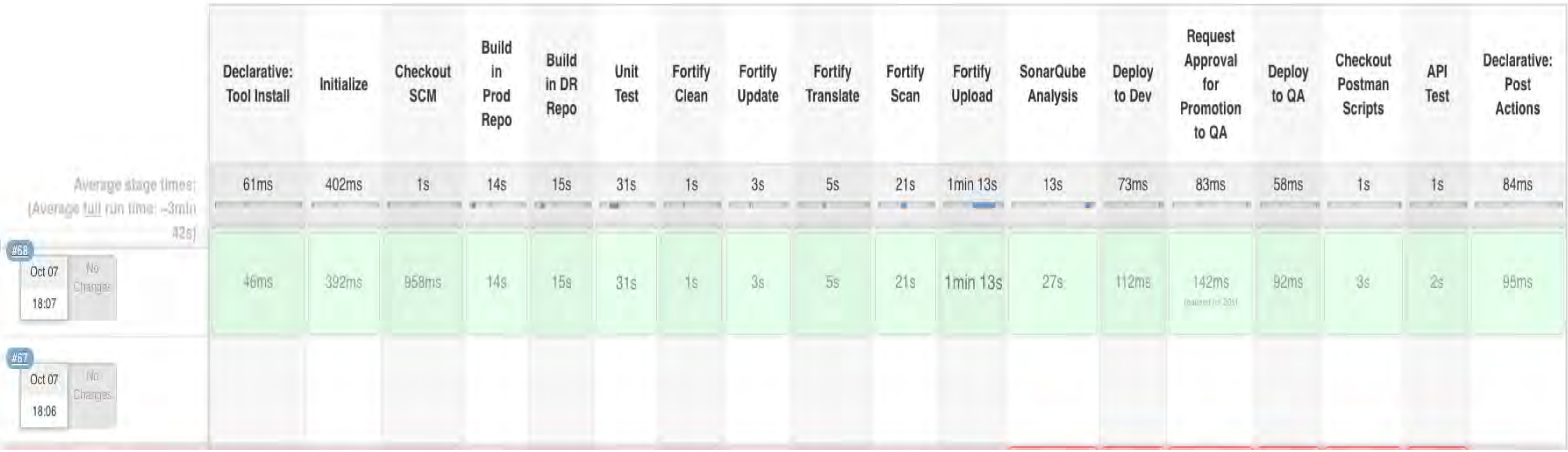
Pipeline Stage view Jenkins (Containers)



Stage View

	Checkout Source	Get artifact version	Build in dev	Deploy to dev	Integration test	Request Approval for Promotion to QA	Deploy to QA	Request Approval for Promotion to PROD	Deploy to Prod
Average stage times: (Average full run time: ~21min 6s)	4s	18s	32s	24s	19ms	42ms	39s	28ms	41s
#14 May 30 15:16 No Changes	4s	26s	33s	23s	15ms	35ms (paused for [illegible])	38s	24ms (paused for [illegible])	47s
#13 May 21 12:13 No Changes	4s	15s	19s	24s	16ms	56ms (paused for [illegible])	47s	31ms (paused for 26s)	34s
#12 Mar 01 12:50 13 commits	4s	32s	27s	23s	27ms	37ms (paused for [illegible])	33s	29ms (paused for [illegible])	41s

Pipeline Stage view Non-Containerized Application



	Declarative: Tool Install	Initialize	Checkout SCM	Build in Prod Repo	Build in DR Repo	Unit Test	Fortify Clean	Fortify Update	Fortify Translate	Fortify Scan	Fortify Upload	SonarQube Analysis	Deploy to Dev	Request Approval for Promotion to QA	Deploy to QA	Checkout Postman Scripts	API Test	Declarative: Post Actions
Average stage times: (Average full run time: ~3min 42s)	61ms	402ms	1s	14s	15s	31s	1s	3s	5s	21s	1min 13s	13s	73ms	83ms	58ms	1s	1s	84ms
#68 Oct 07 18:07 No Changes	46ms	392ms	958ms	14s	15s	31s	1s	3s	5s	21s	1min 13s	27s	112ms	142ms (paused for 20s)	92ms	3s	2s	95ms
#67 Oct 07 18:06 No Changes																		

Upcoming Functionalities F5 Automation

- Ansible integration with F5. Currently some jobs integrated with Brocade and we want to make this functionality available for F5
- Ansible’s network automation framework provides teams with ability to automate F5 BIG-IP via API called Modules.
- Ansible F5 Module enables most common use case such as
- Automated additions of Virtual Servers, Nodes and Pools.
- This will enable Zero Downtime deployment of HTTP and HTTPS Applications with automated disabling of pool members, where possible.

Other Upcoming initiatives

- End to End Pipeline – On-perm to Cloud
- UFT
- Cloud Pipeline
- *Selenium* integration