



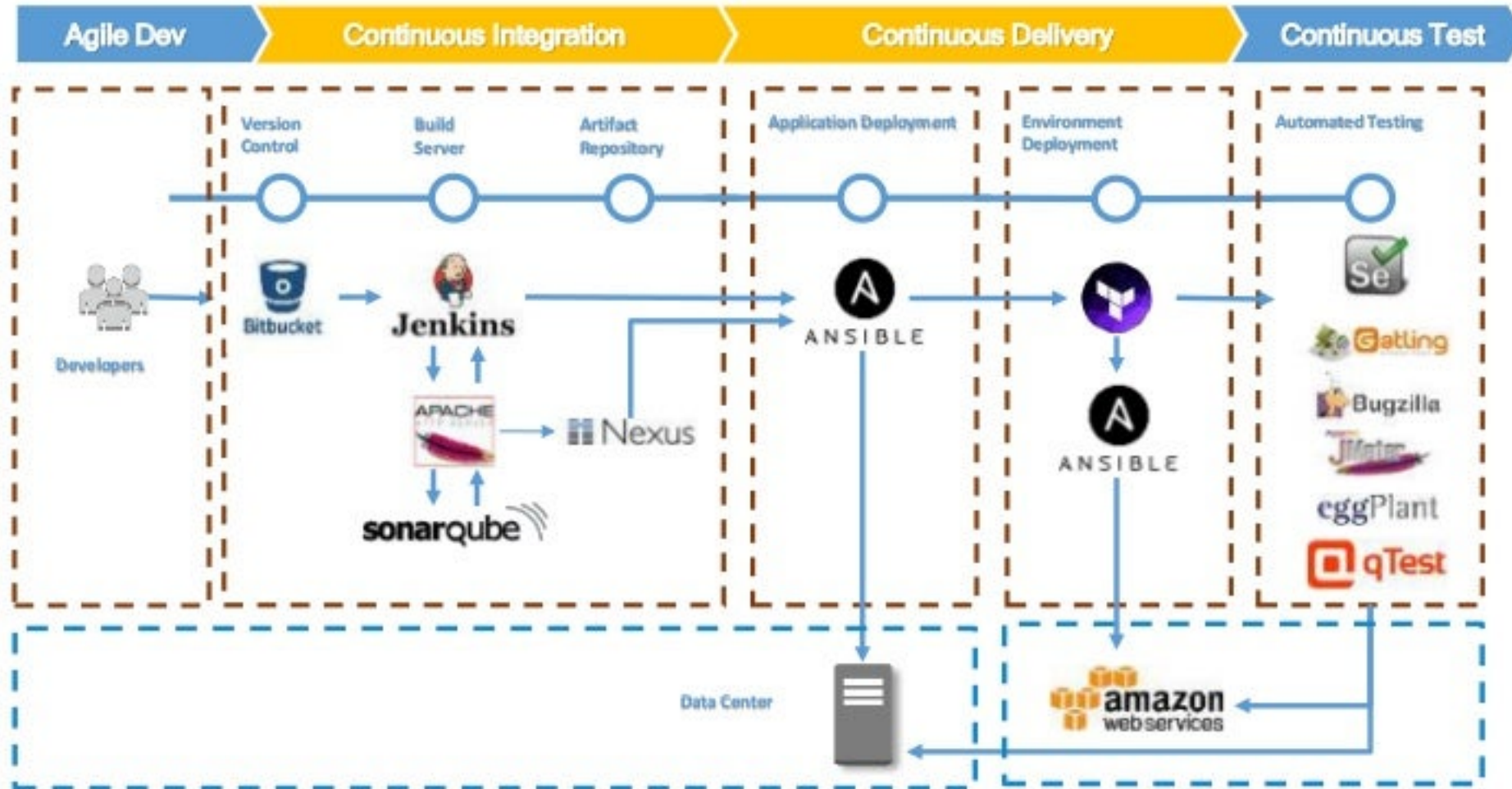
DevOps and Continuous Delivery Reference Architectures

Derek E. Weeks
VP and DevOps Advocate
Sonatype

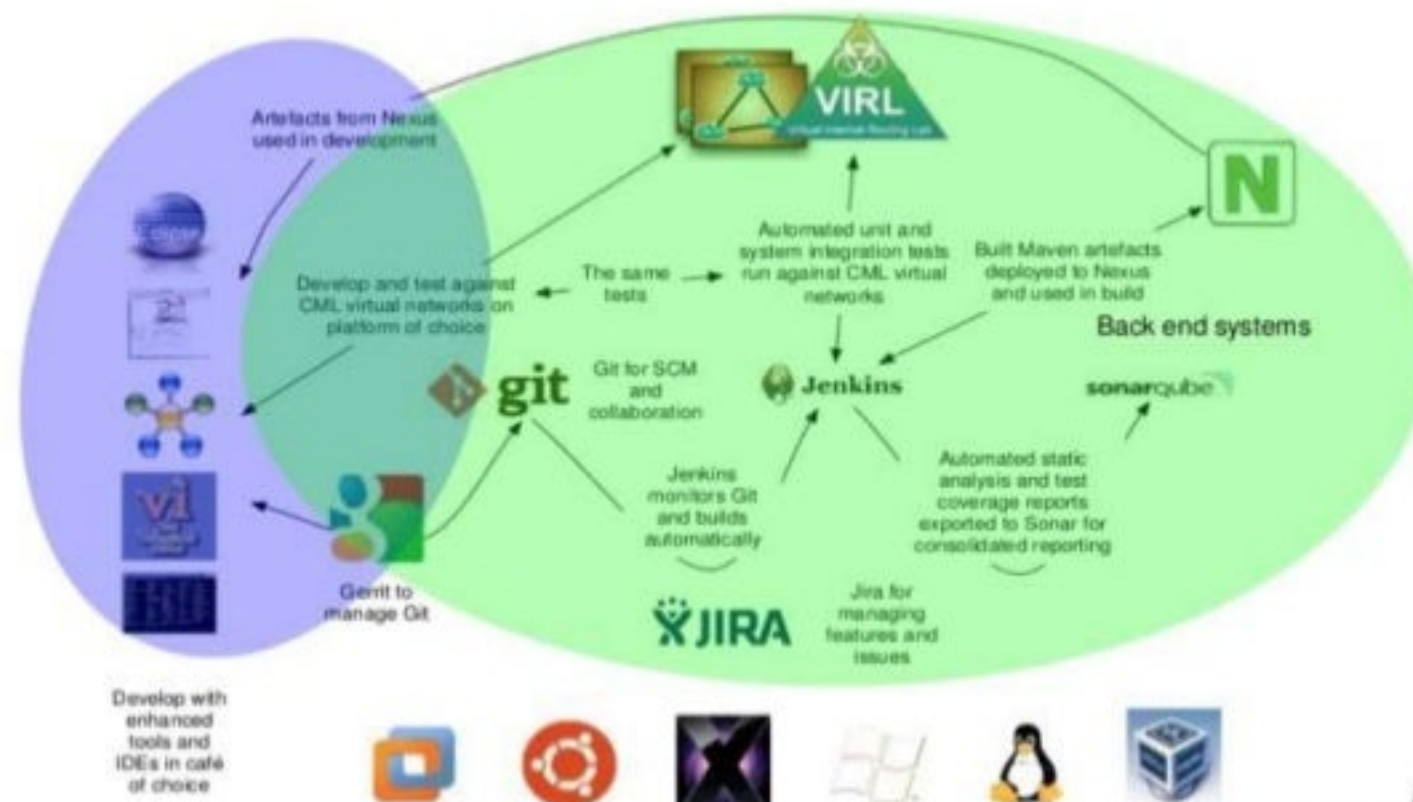
Common Elements of the Software Supply Chain



According to TUI



According to Cisco



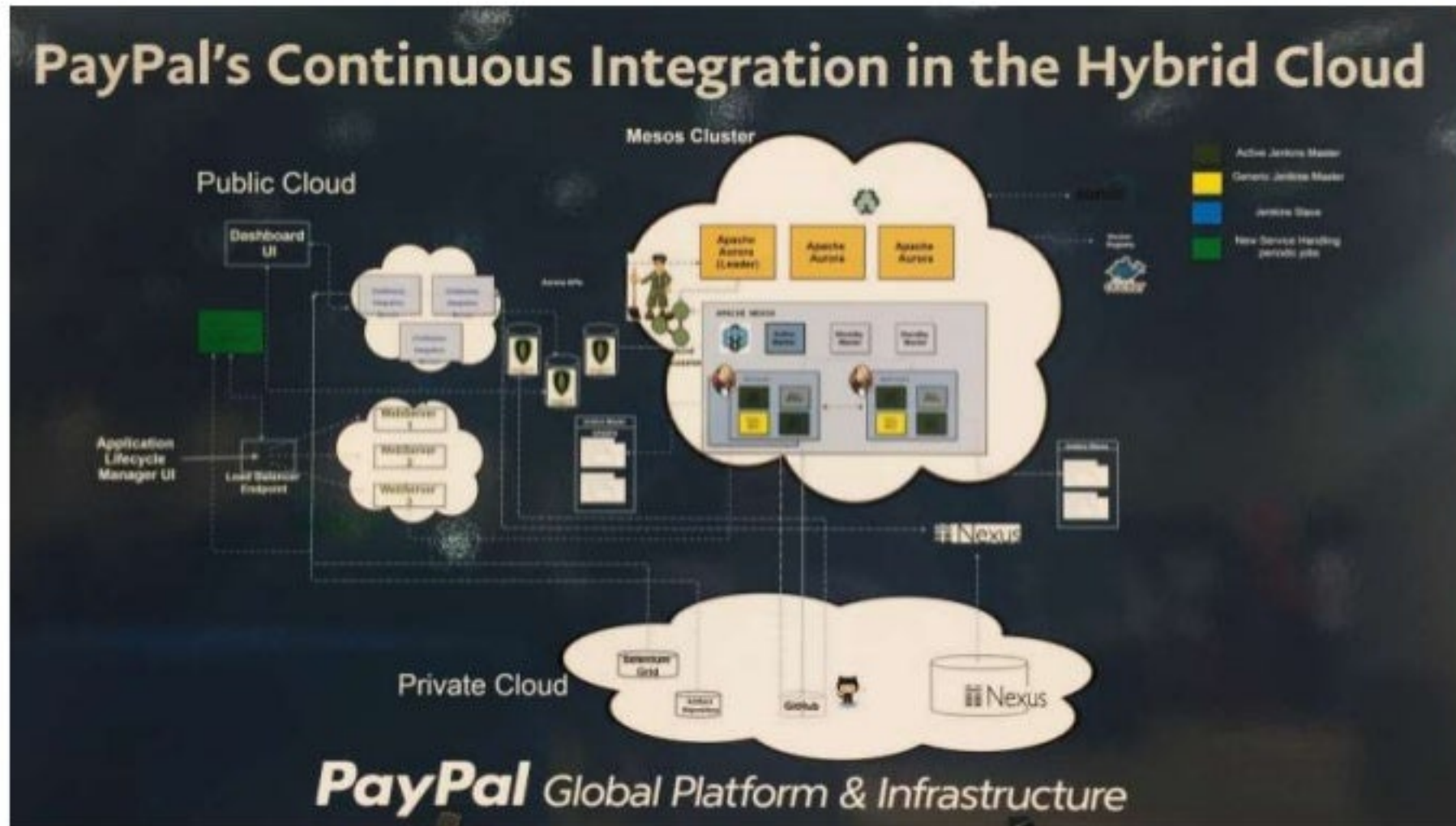
© 2014 Cisco and/or its affiliates. All rights reserved.

Cisco Public

9

CiscoLive!

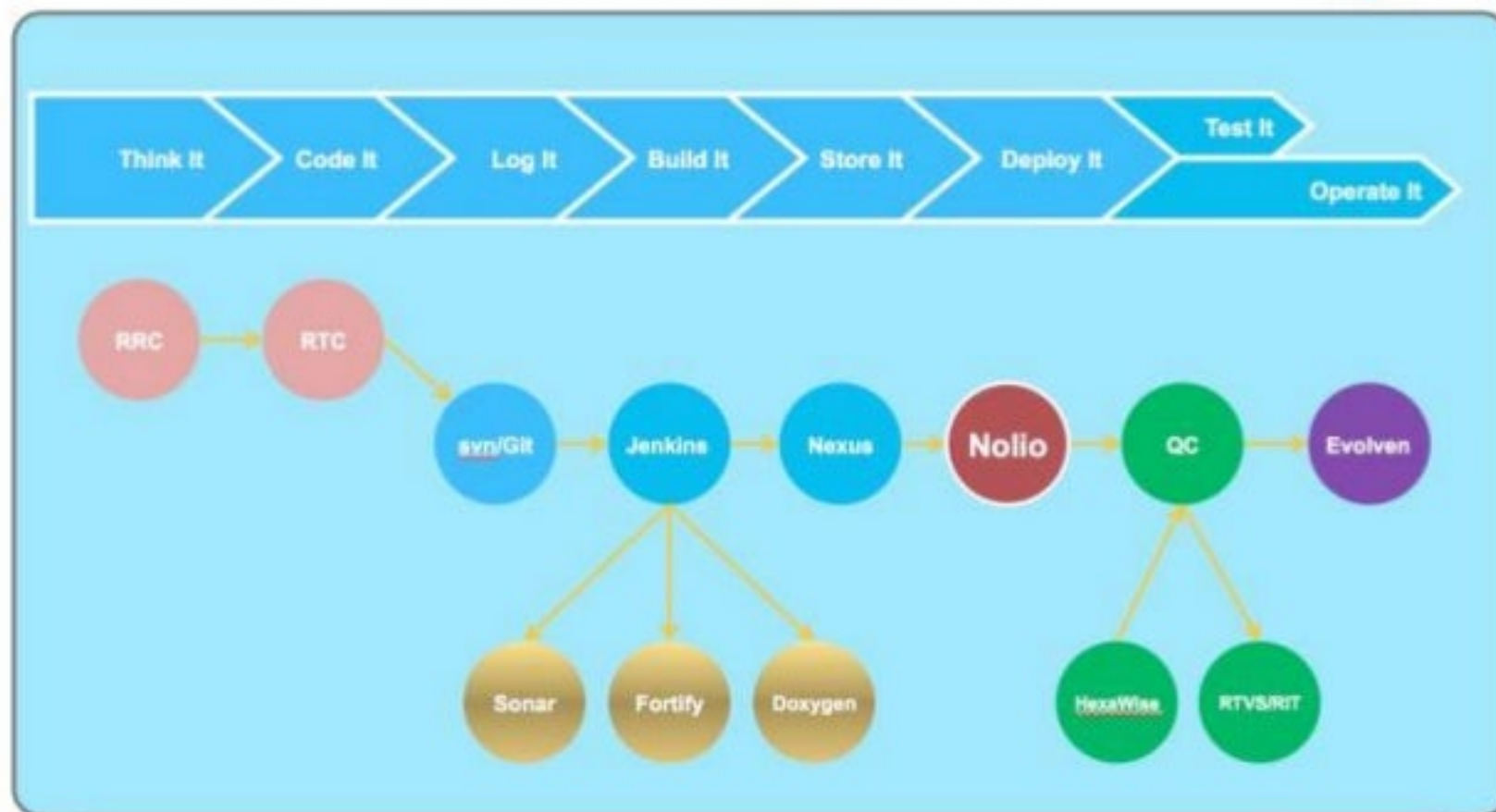
According to PayPal



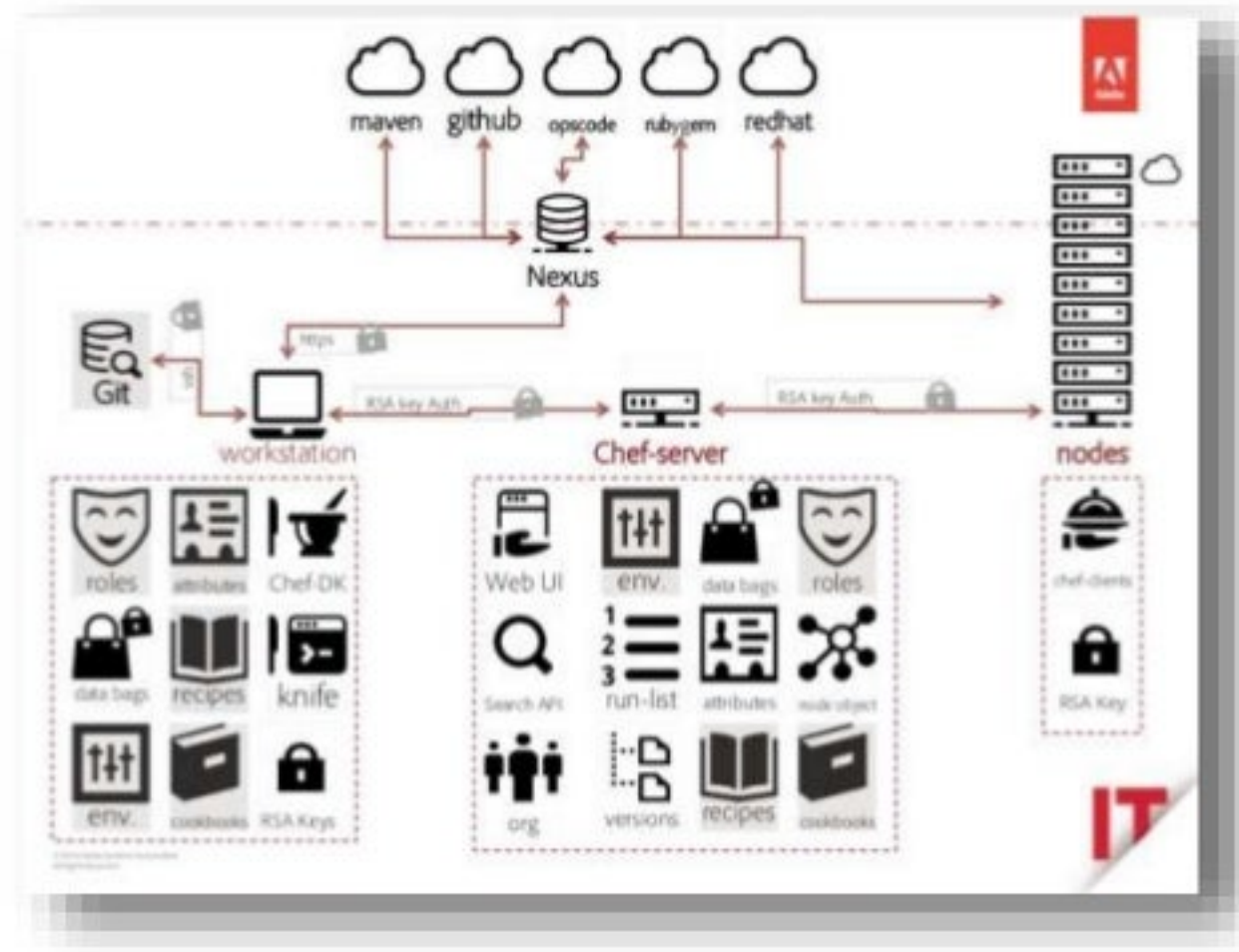
According to BARCLAYS



Quantum – Automated Deployment



According to Adobe



DevSecOps: How to Seamlessly Integrate Security Into DevOps

Published: 30 September 2016 ID: G00311292

Analyst(s): Neil MacDonald, Ian Head

Information security architects must integrate security at multiple points into DevOps workflows in a collaborative way that is largely transparent to developers, and preserves the teamwork, agility and speed of DevOps and agile development environments, delivering "DevSecOps."

Key Challenges

- DevOps compliance is a top concern of IT leaders, but information security is seen as an inhibitor to DevOps agility.
- Security infrastructure has lagged in its ability to become "software defined" and programmable, making it difficult to integrate security controls into DevOps-style workflows in an automated, transparent way.
- Modern applications are largely "assembled," not developed, and developers often download and use known vulnerable open-source components and frameworks.

Recommendations

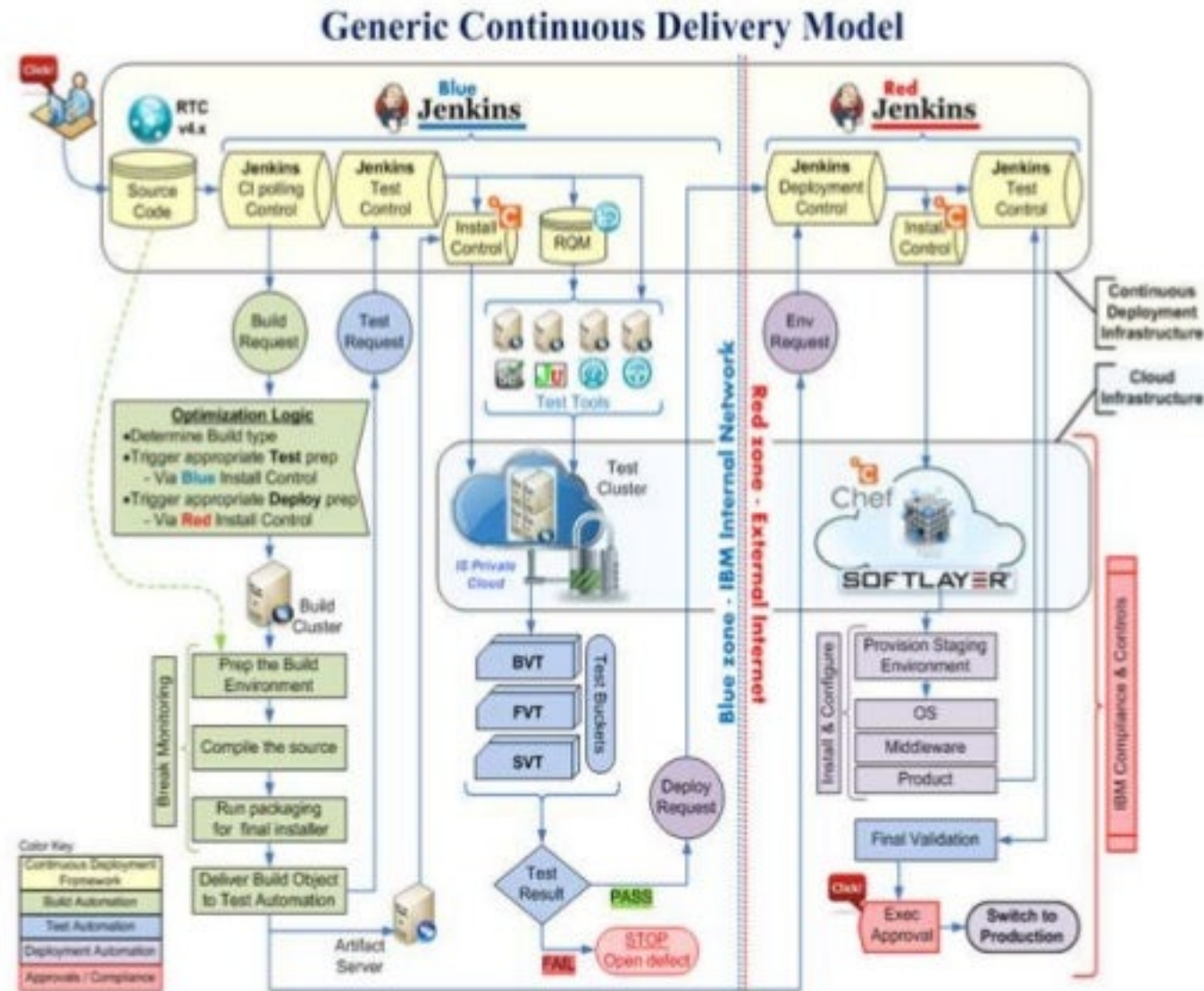
Information security architects should:

- Start with secure development and training, but don't make developers become security experts or switch tools.
- Embrace the concept of people-centric security and empower developers to take personal responsibility for security compensated for with monitoring. Embrace a "trust and verify" mindset.
- Require all information security platforms to expose full functionality via APIs for automatability.
- Use proven version control practices and tools for all application software and, equally as important, for all scripts, templates and blueprints used in DevOps environments.
- Adopt an immutable infrastructure mindset where production systems are locked down and changed via development.

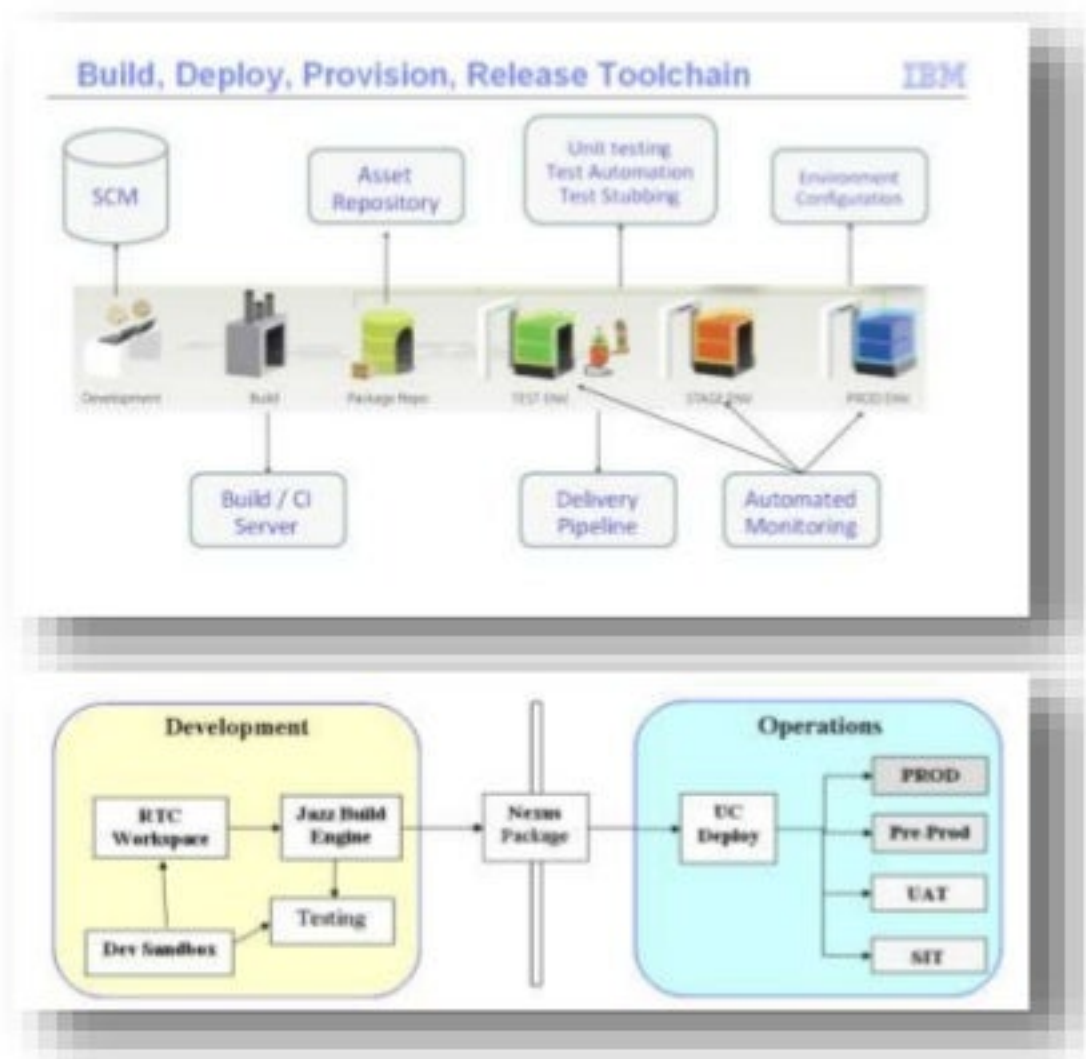
Learn more about DevOps and Security.

www.sonatype.com/DevSecOps

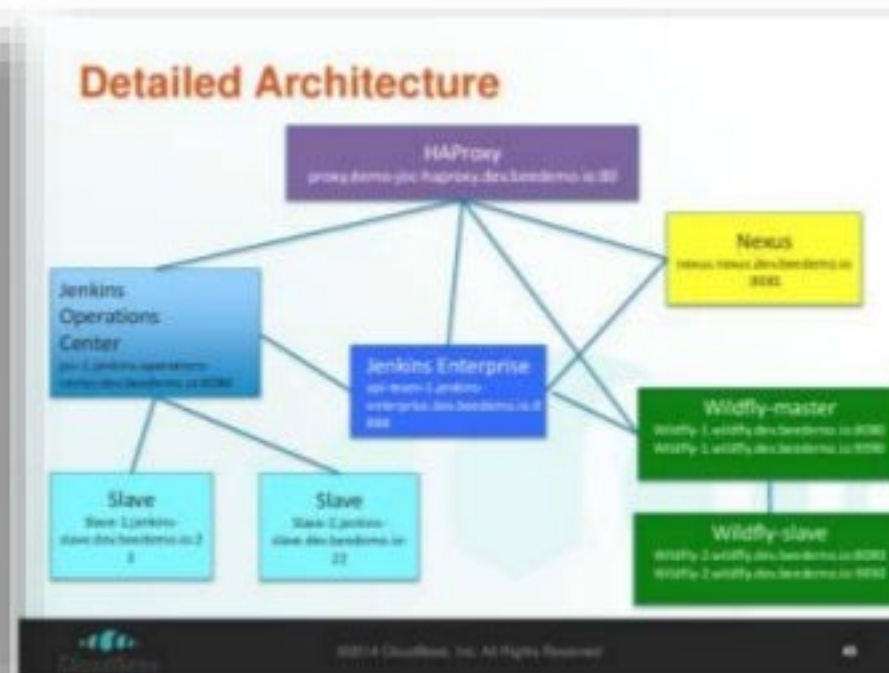
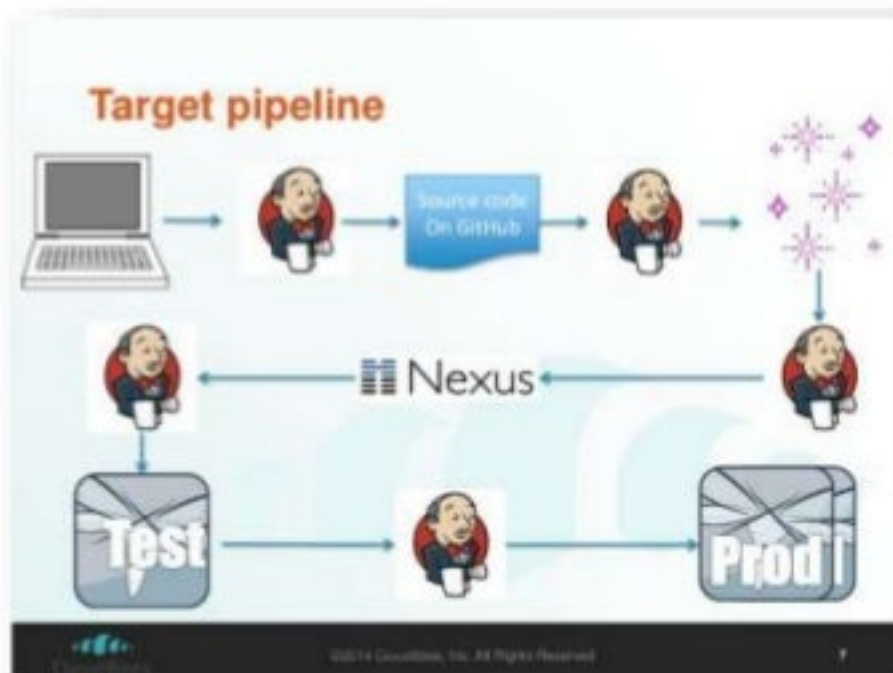
According to IBM



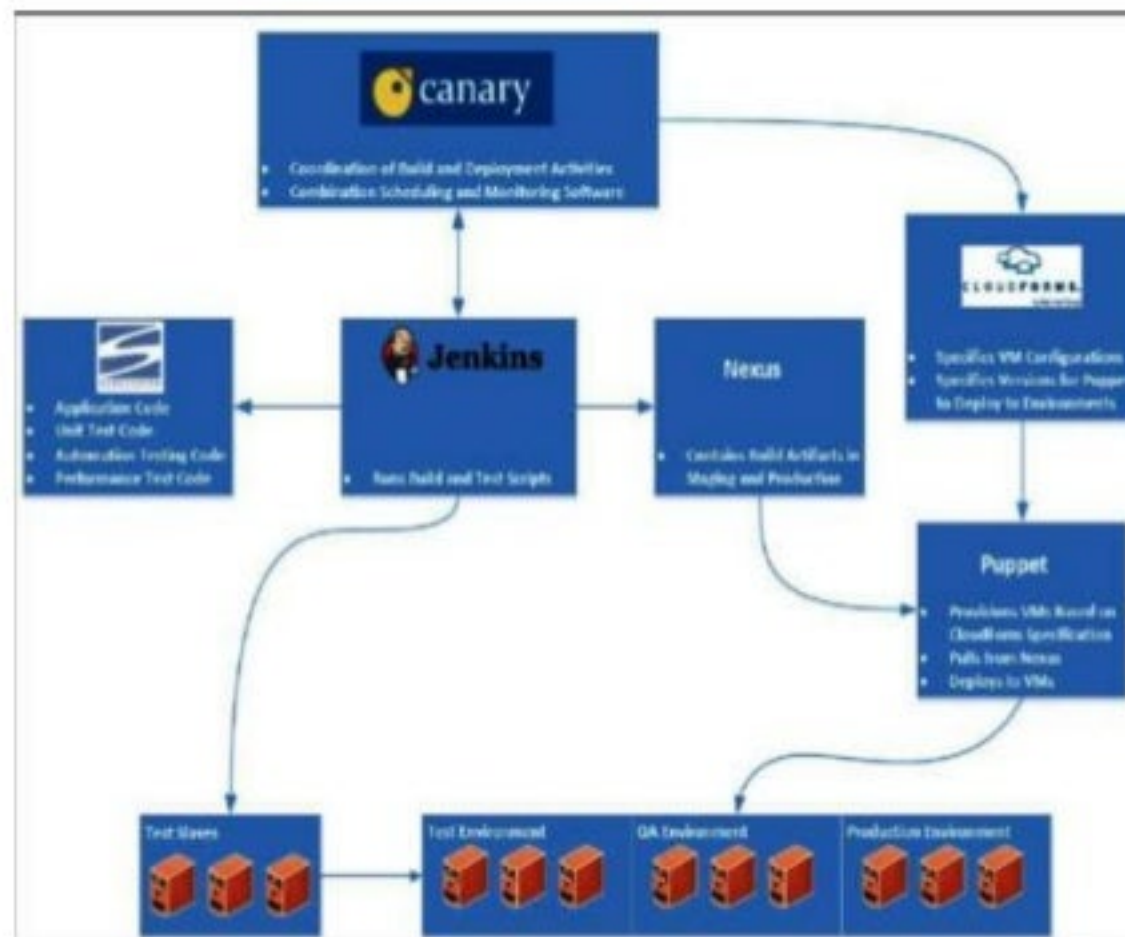
According to IBM



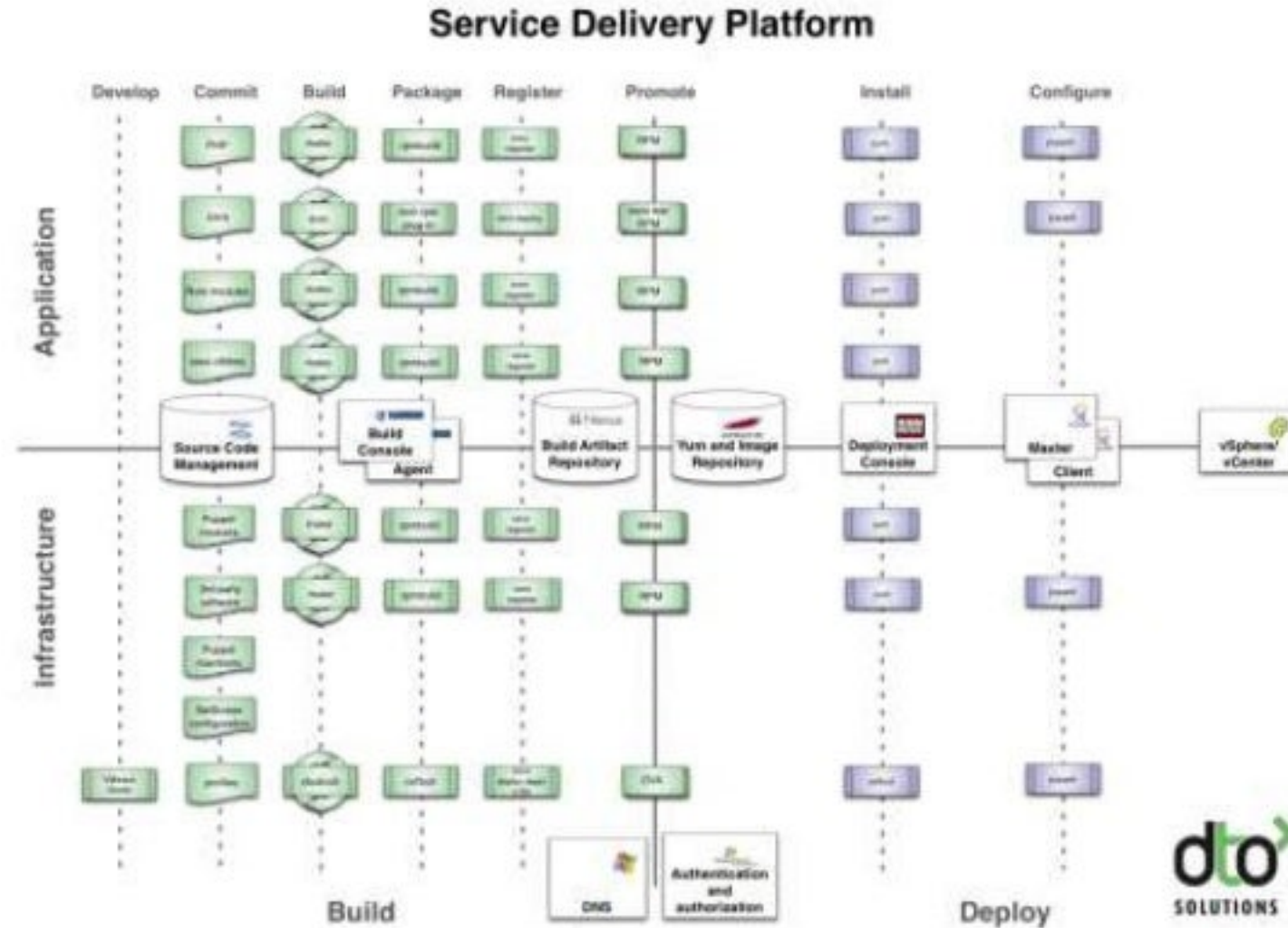
According to Cloudbees



According to US Patent & Trade Office (USPTO)



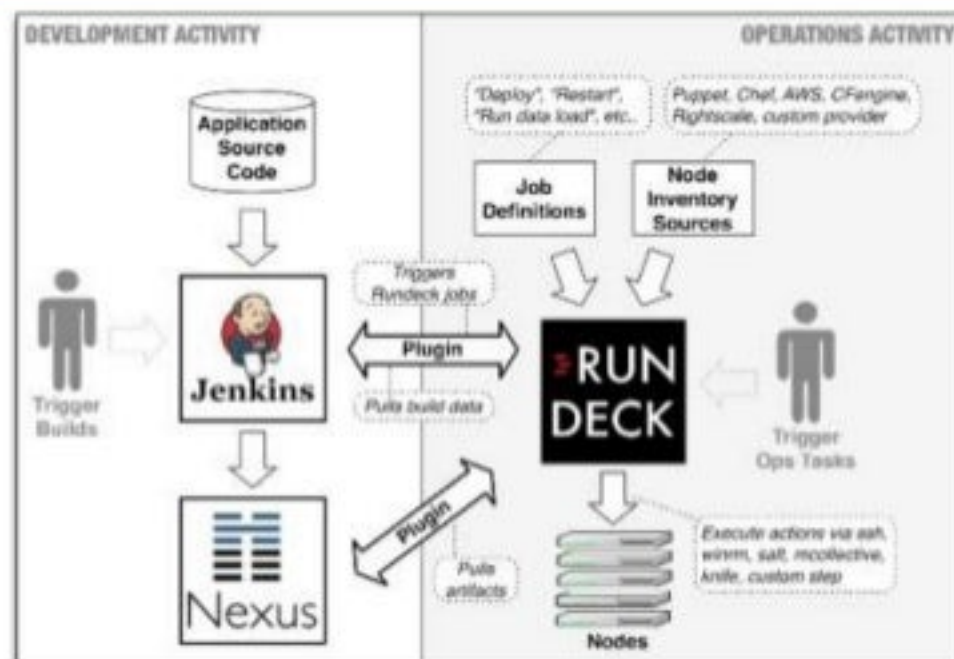
According to Rundeck



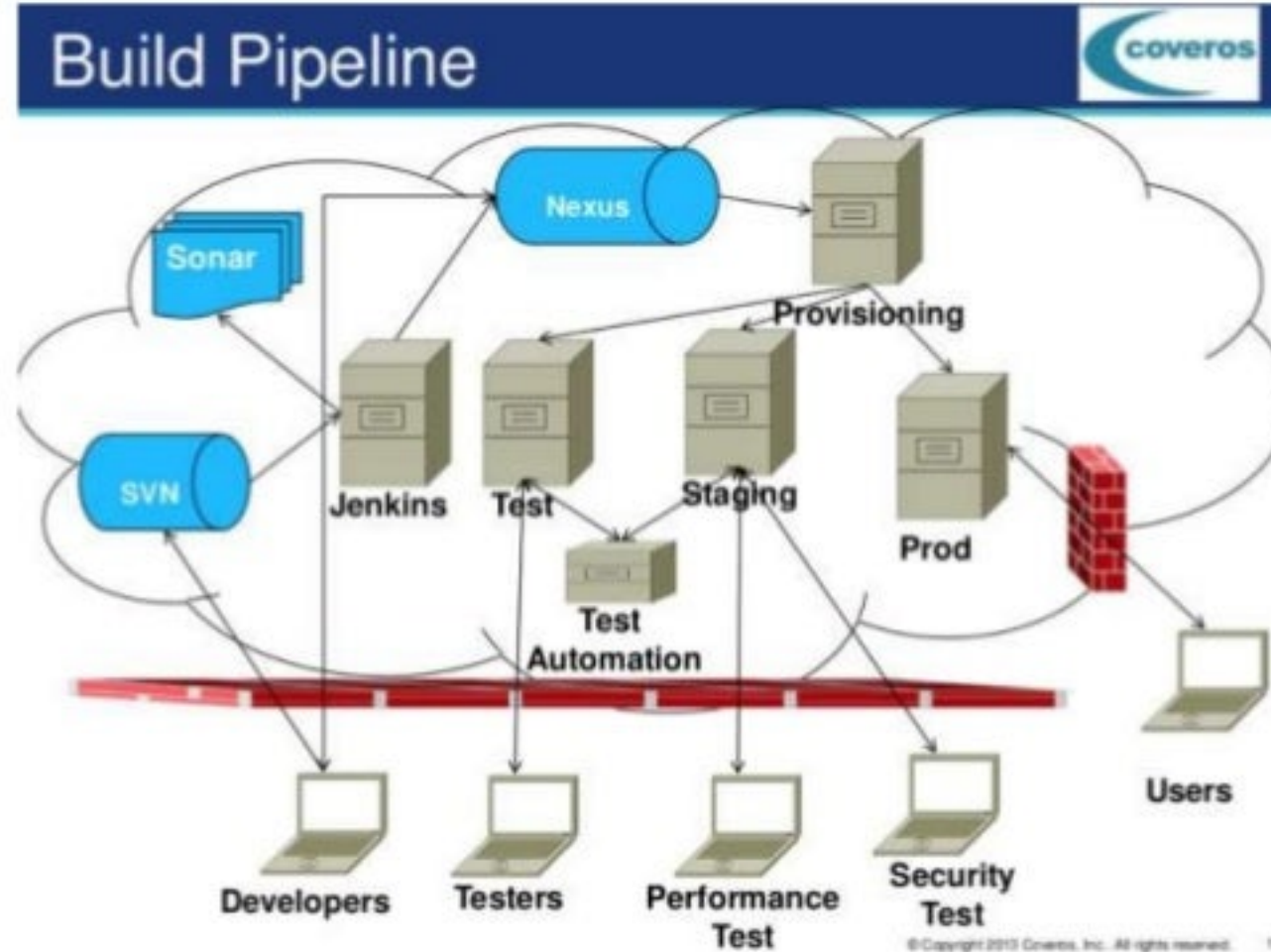
According to Rundeck



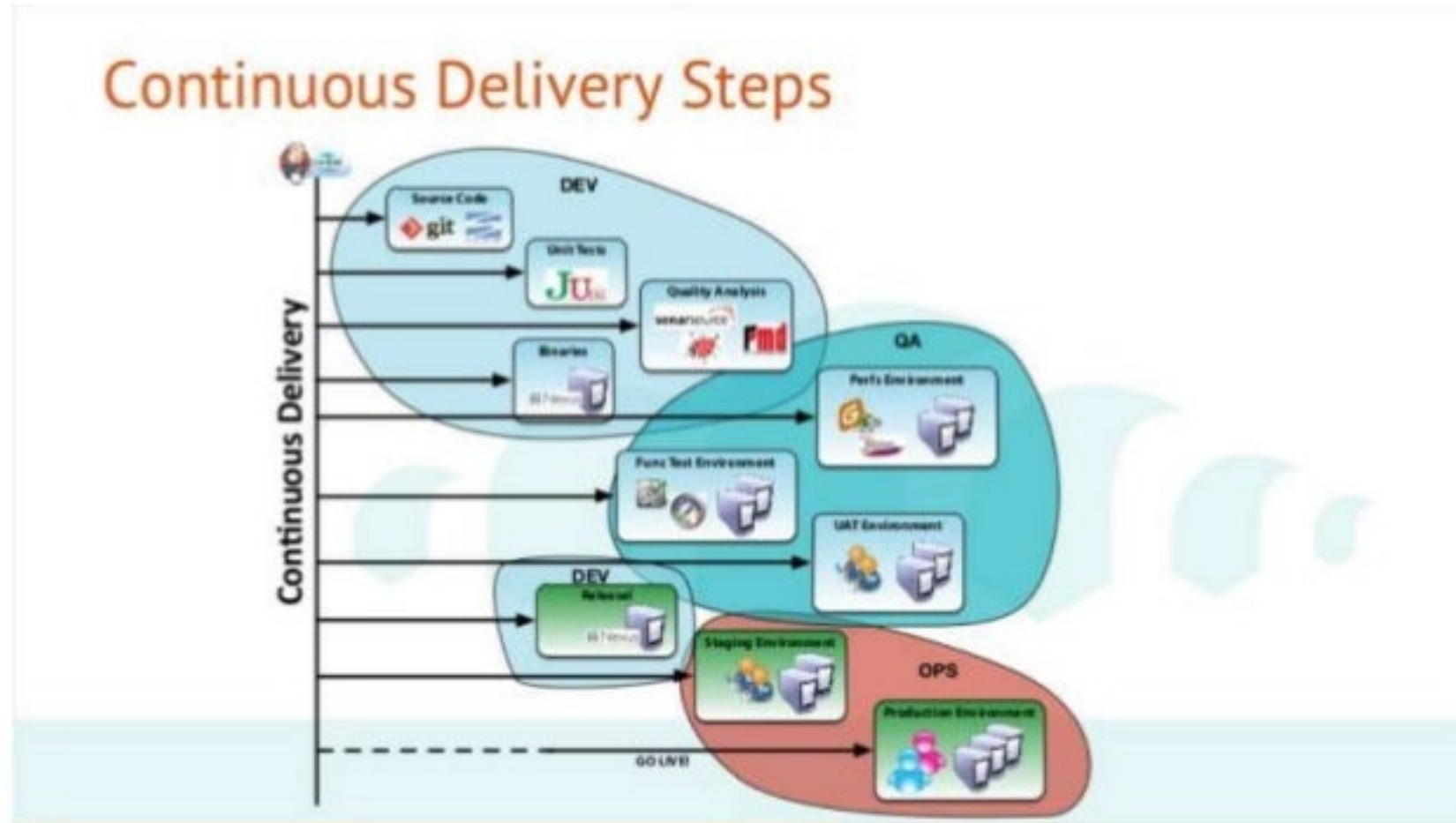
Popular Trio: Jenkins + Nexus + Rundeck



According to Coveros

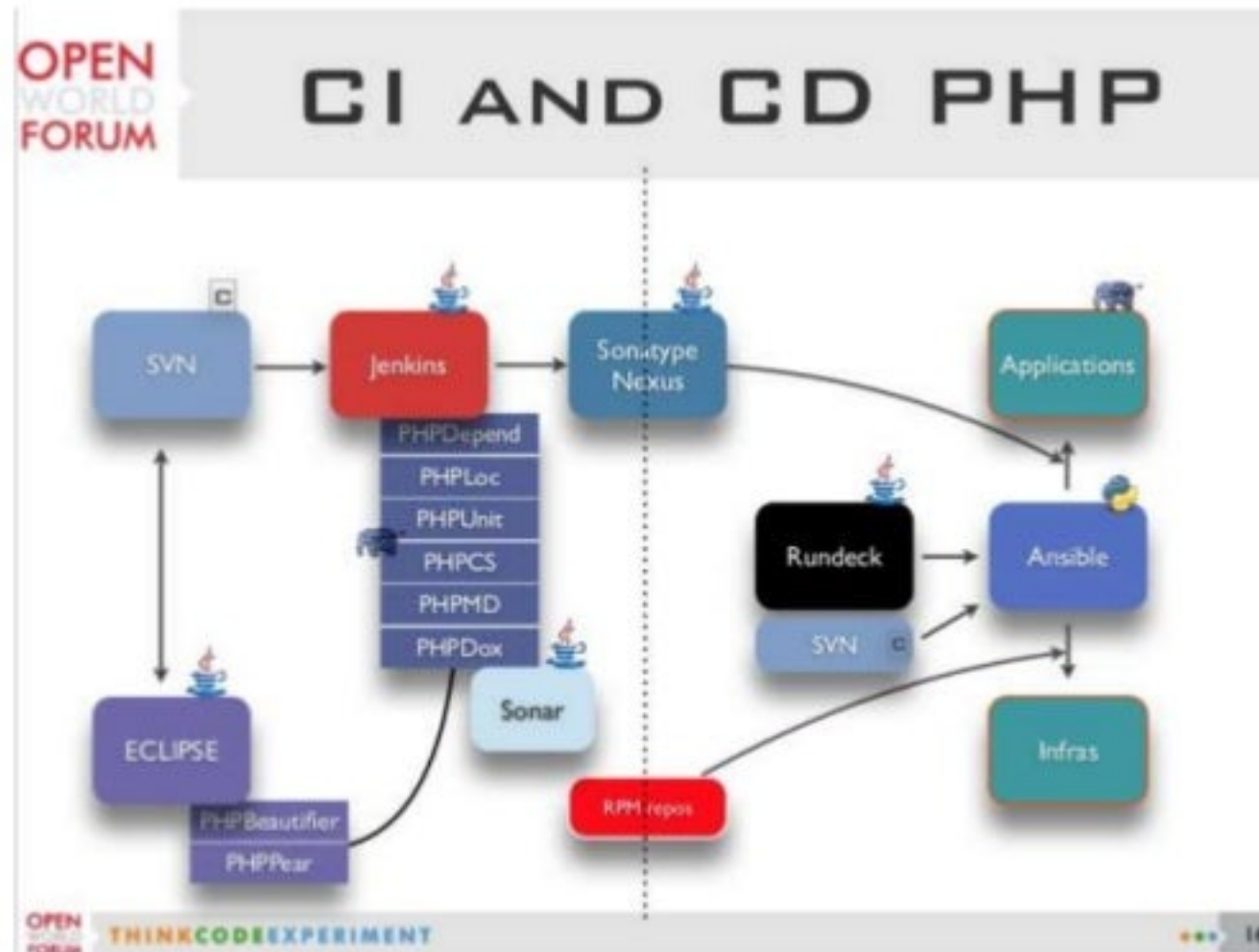


According to Xebia

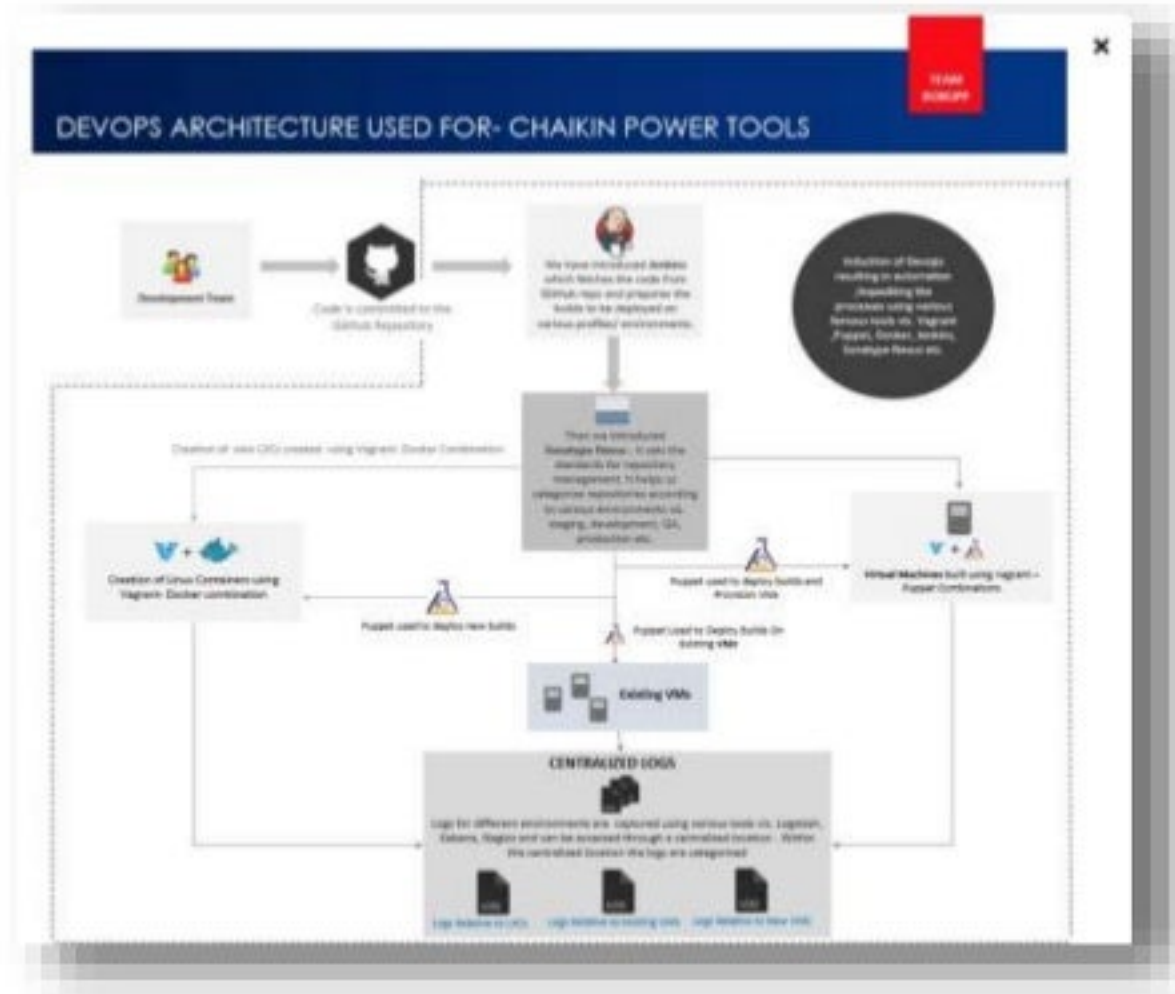


Nexus at Alter Way

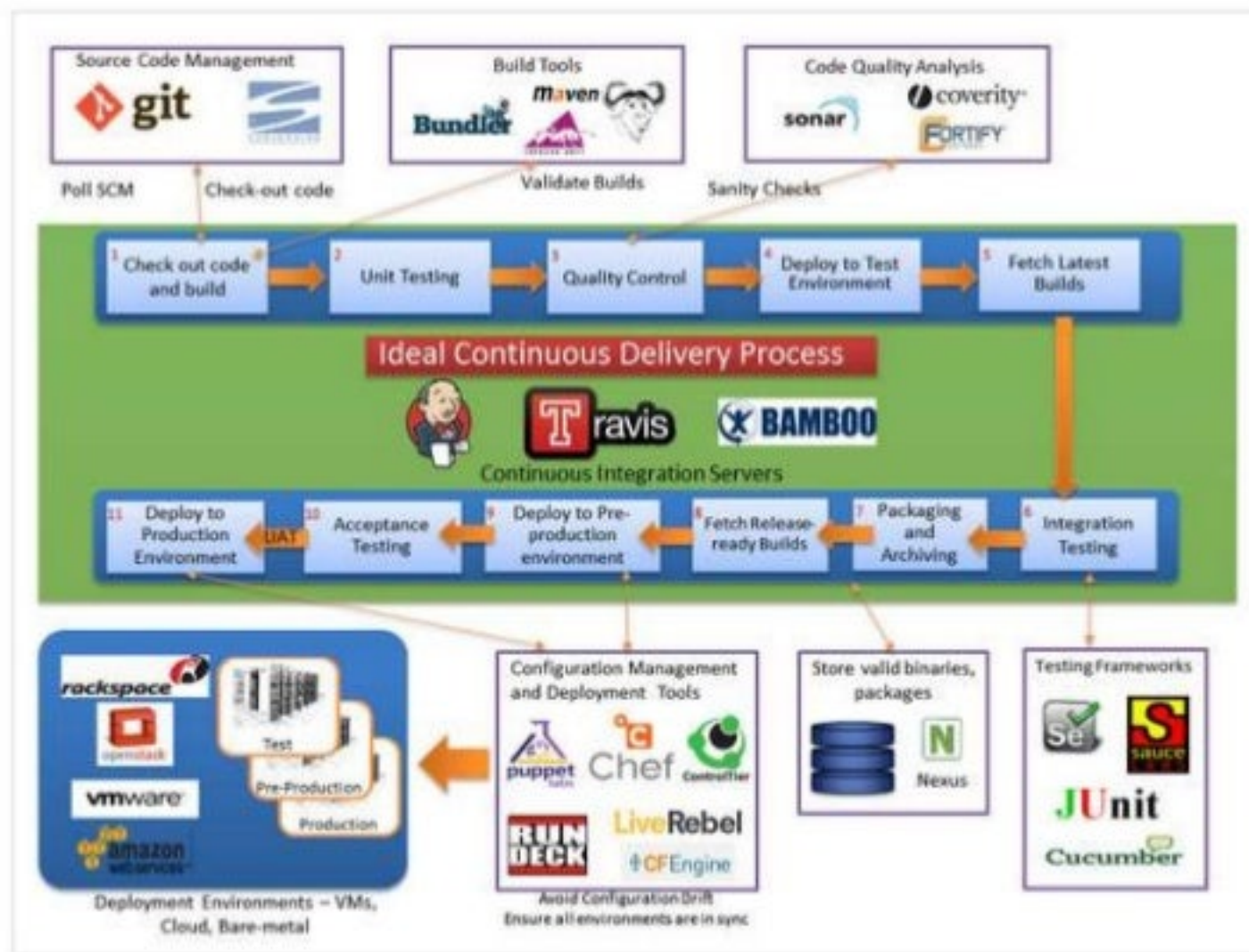
Alter Way



According to Boxupp

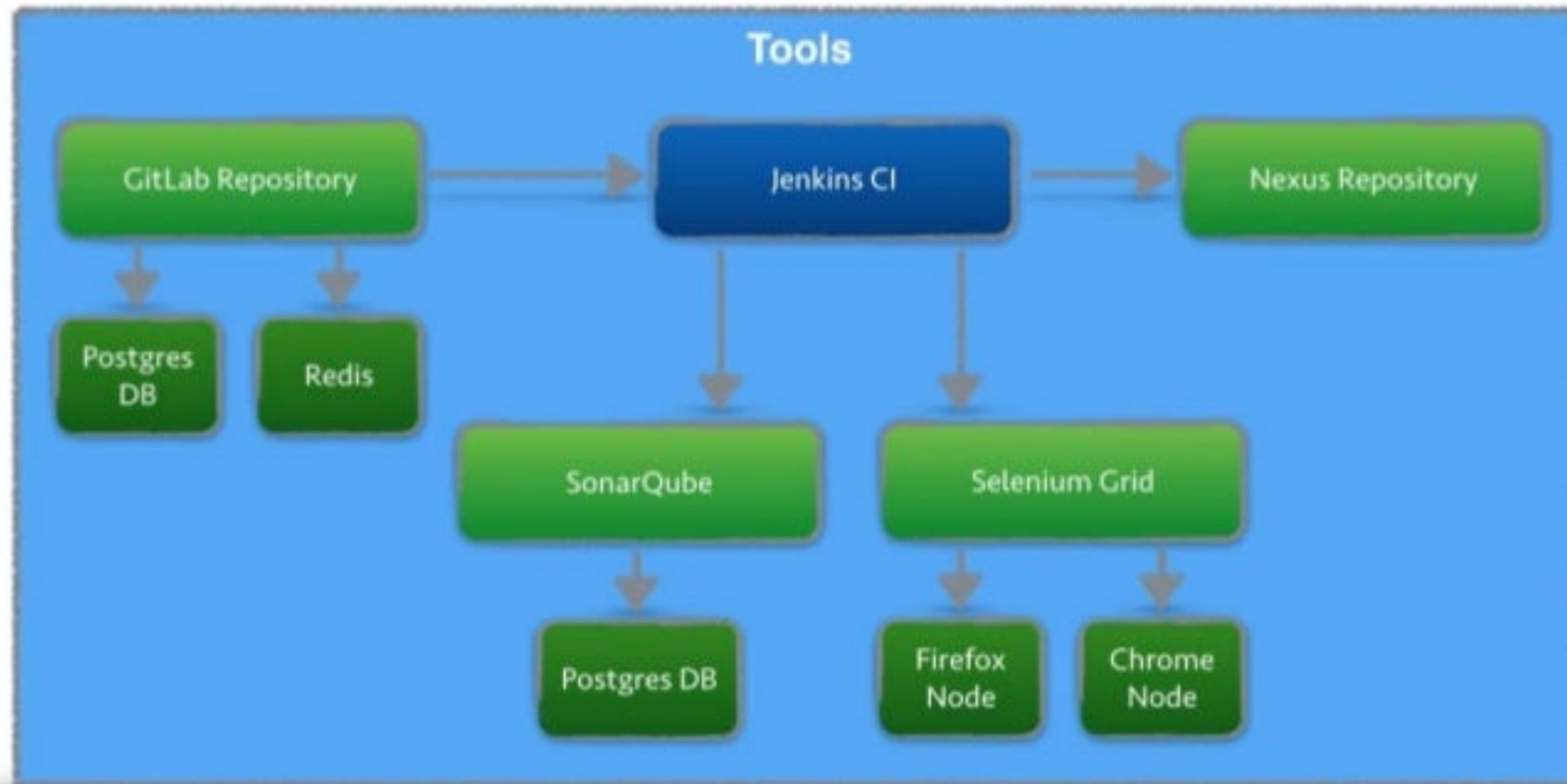


According to bogotobogo



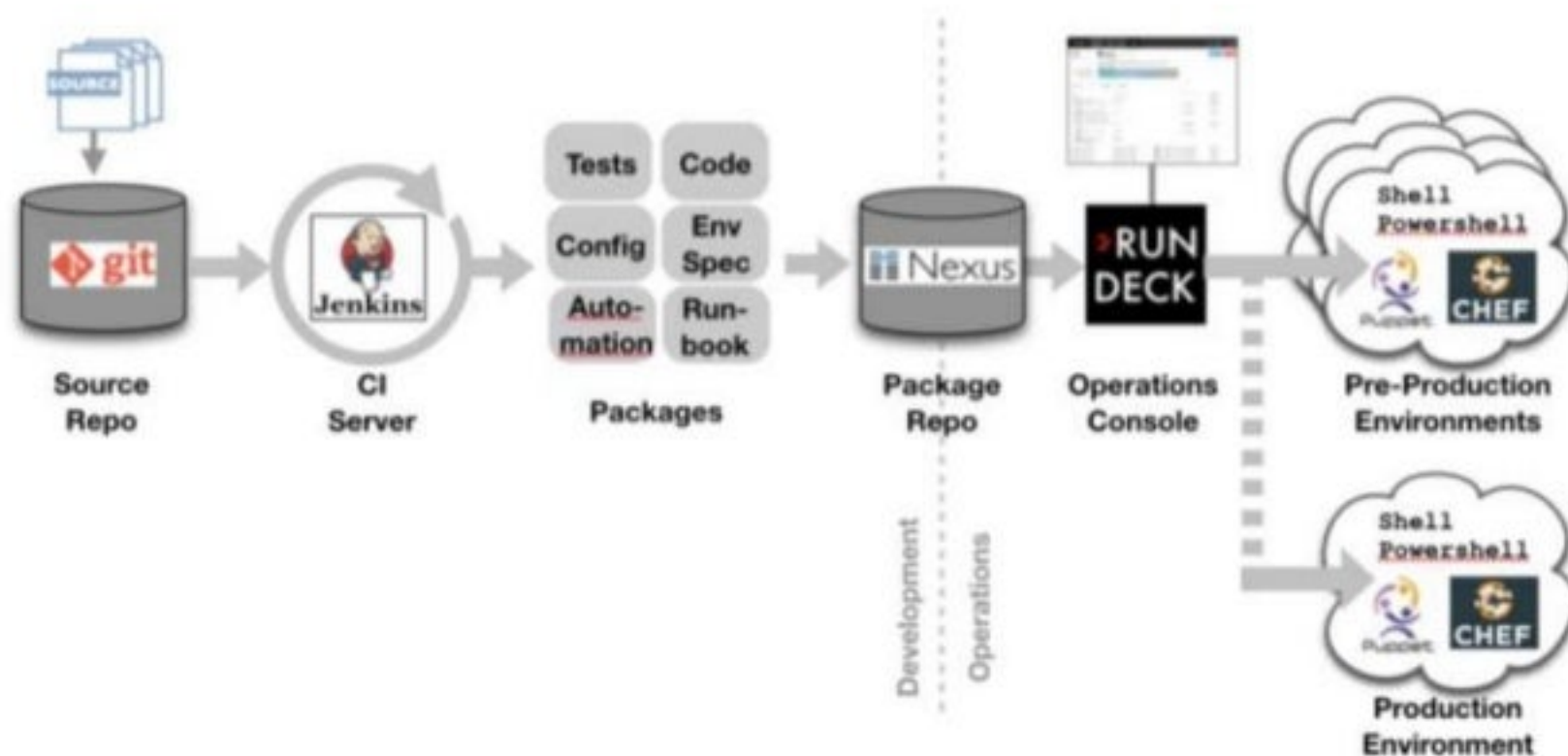
According to codecentric

Continuous Integration Platform Using Docker Containers: Jenkins, SonarQube, Nexus, GitLab



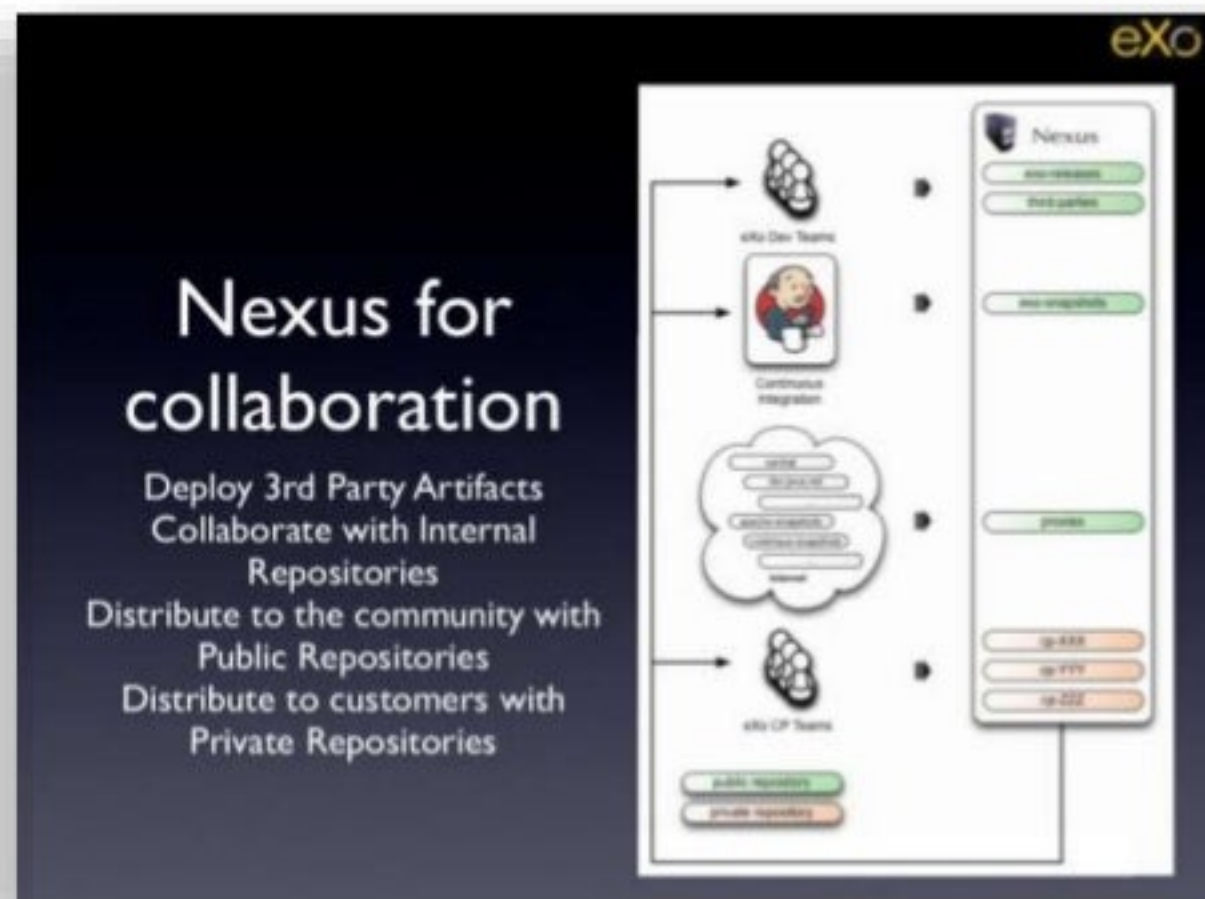
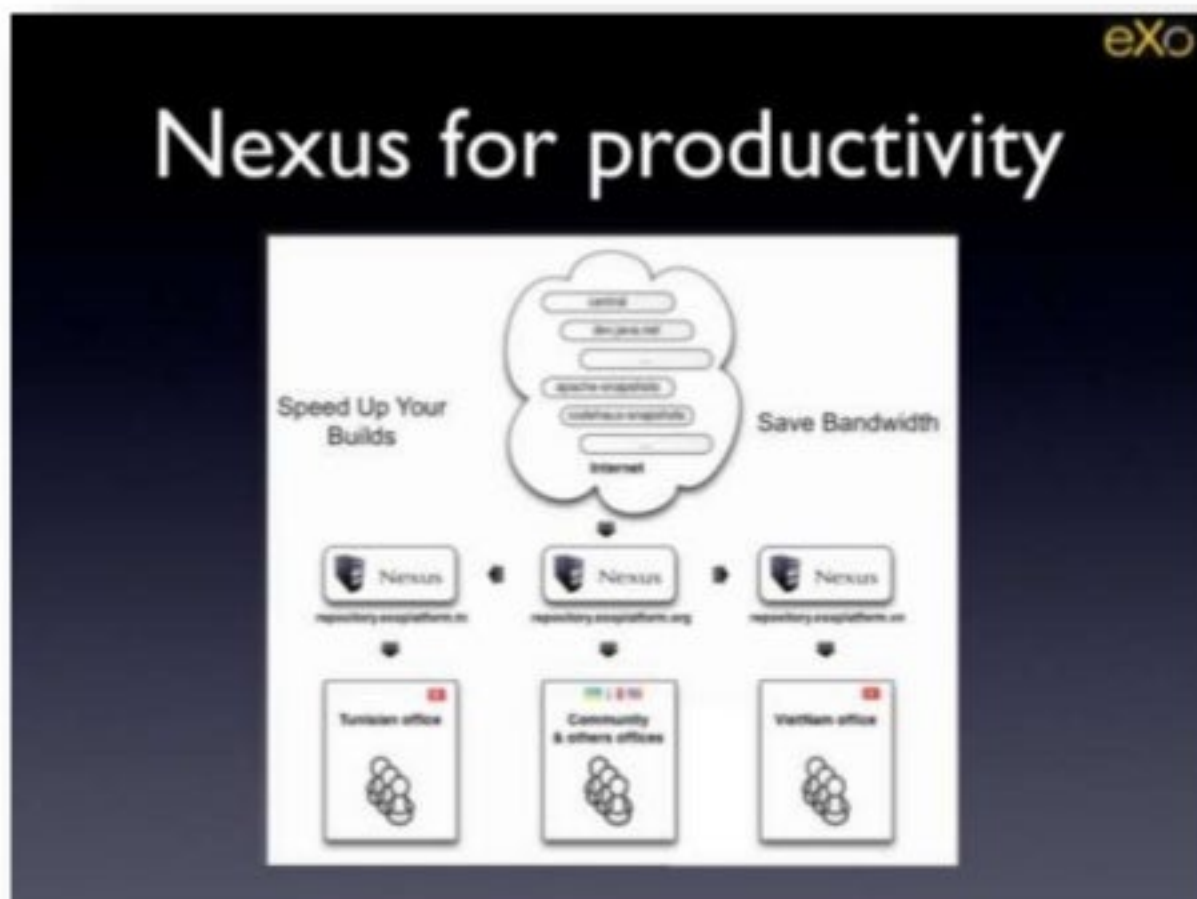
According to SimplifyOps

#SimplifyOps



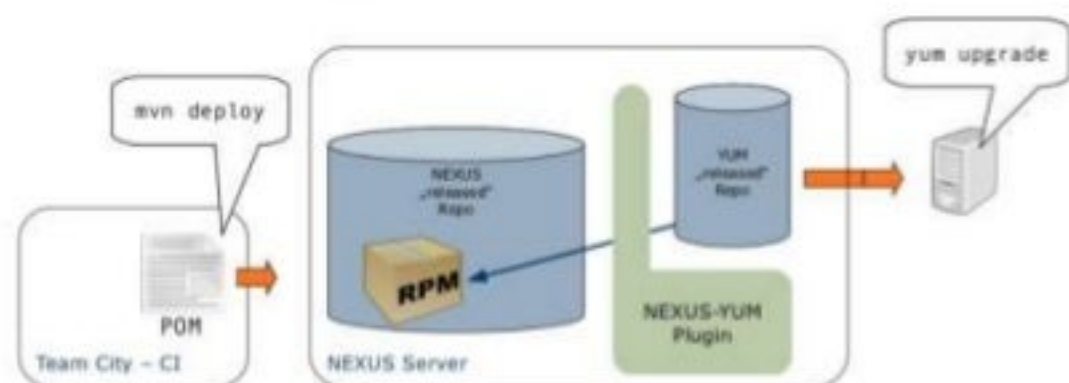
#SimplifyOps

According to eXo Software



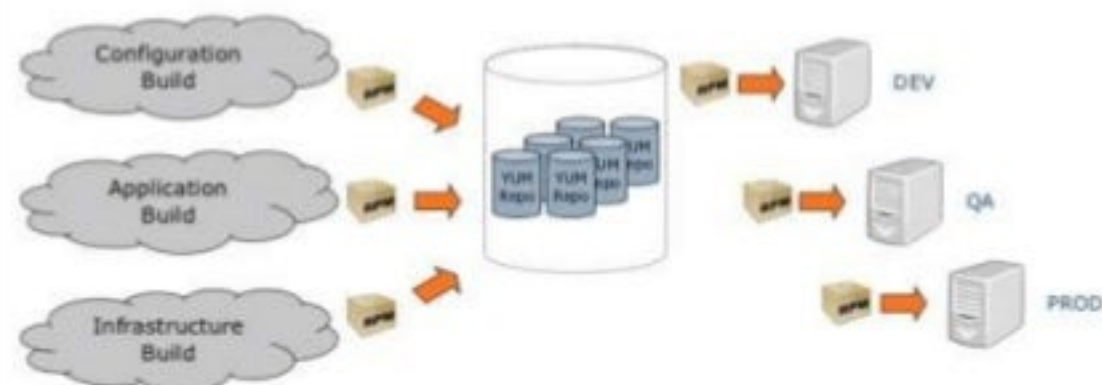
According to ImmobilienScout24

Application Build – Site-wide Releases



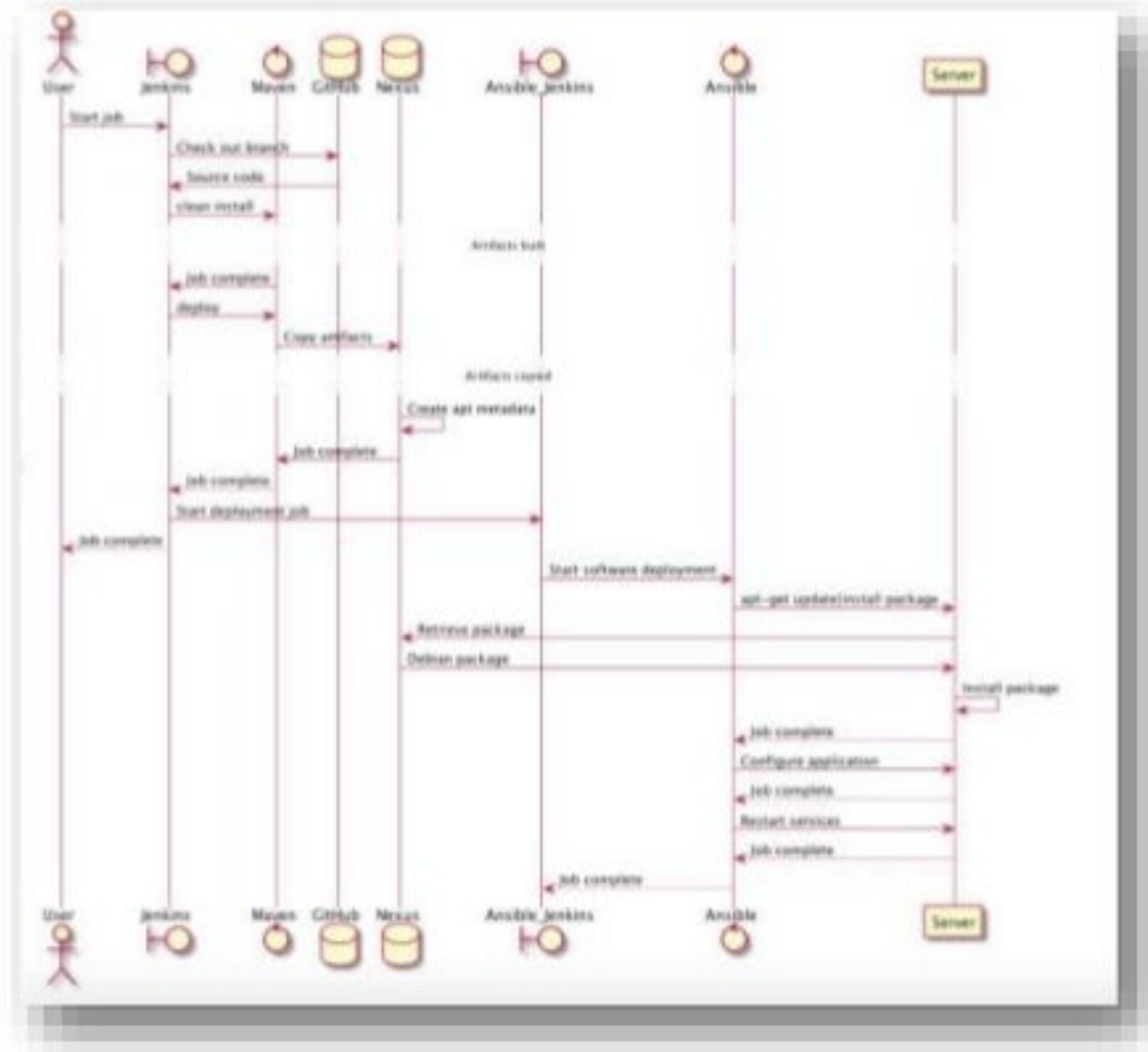
20 Overview **File Mgmt** Systems Mgmt Lessons Learned

File Management through many YUM repositories

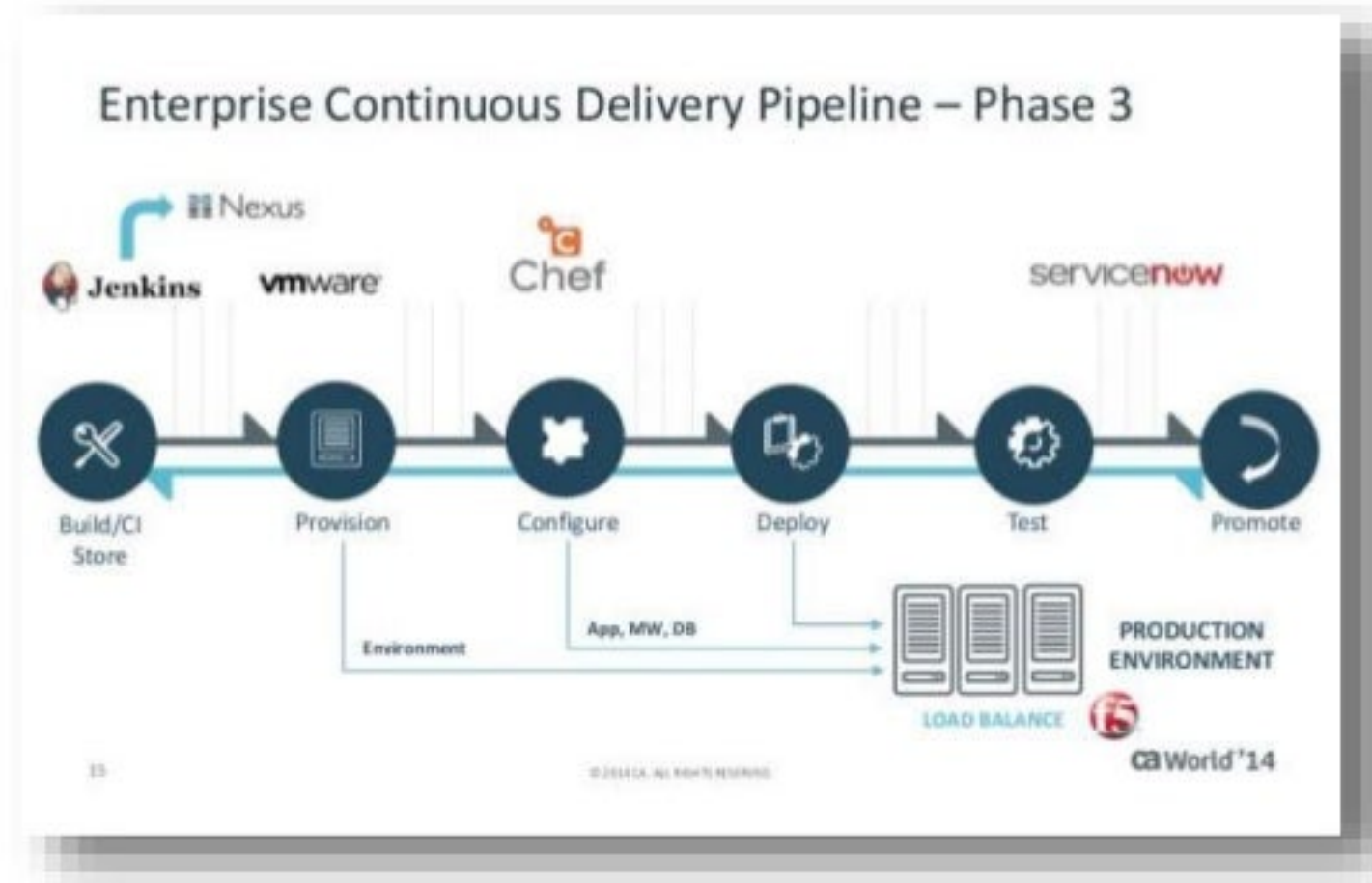


21 Overview **File Mgmt** Systems Mgmt Lessons Learned

According to IHTSDO



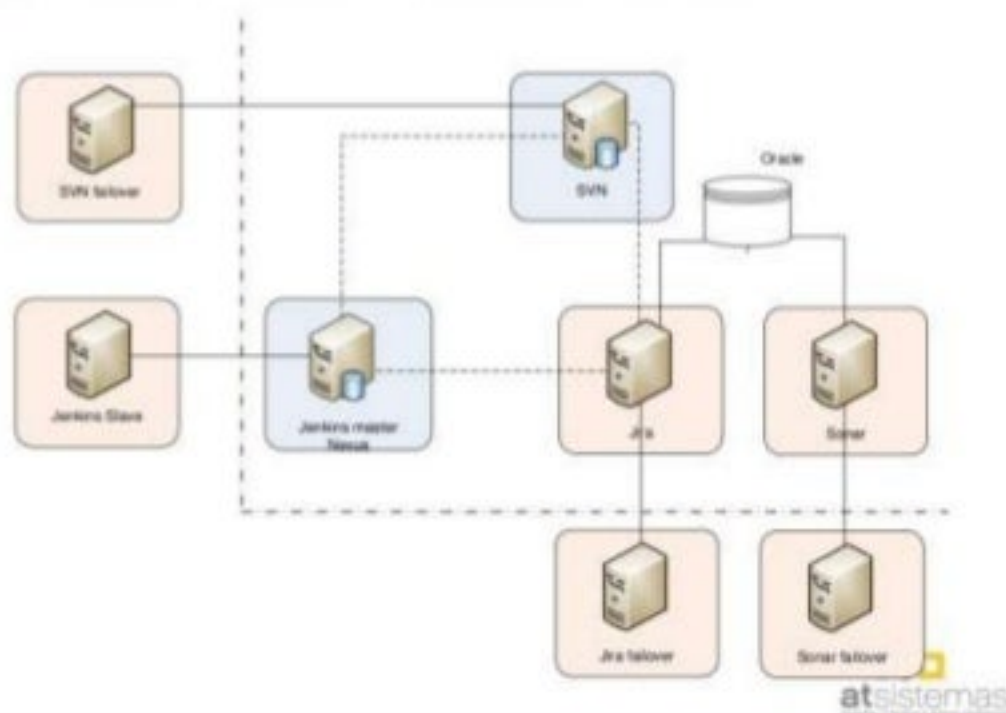
According to CA Technologies



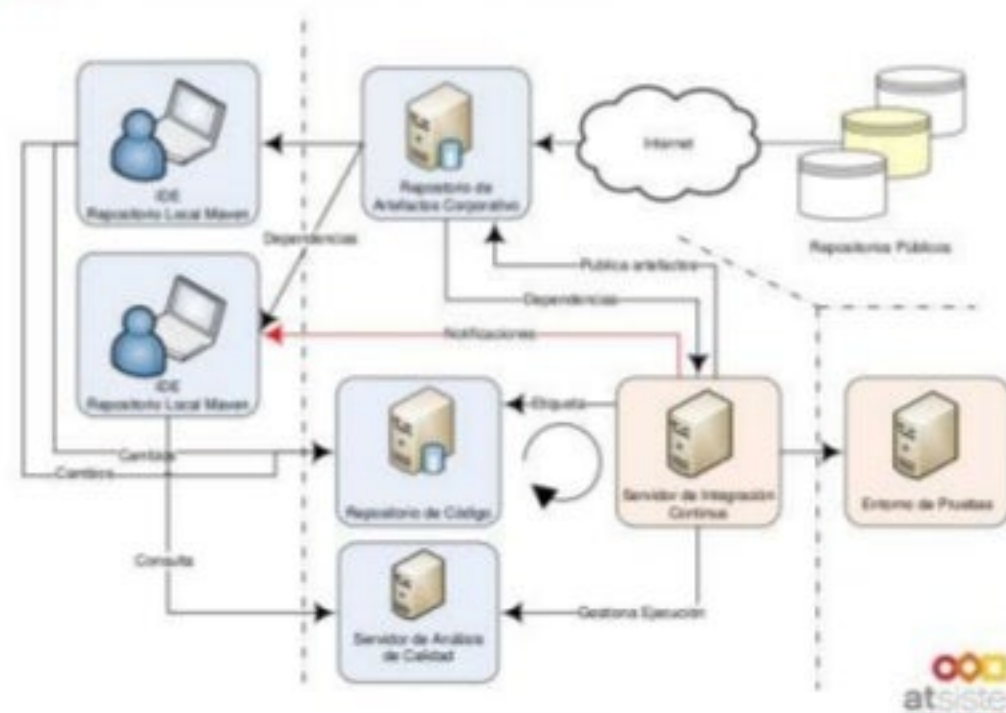
According to atSistemas



3. Caso práctico – Diagrama de sistemas



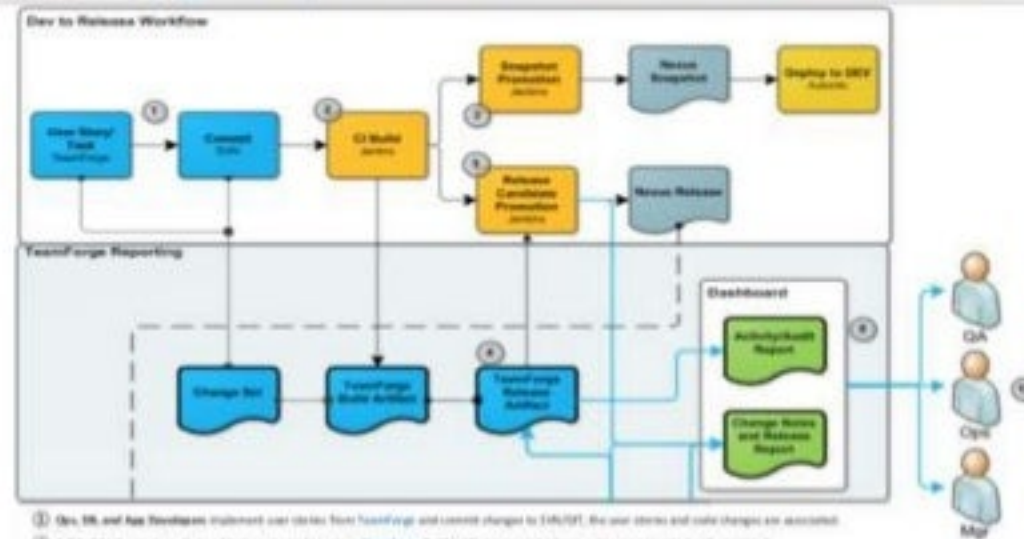
3. Caso práctico - Desarrollo



According to CollabNet

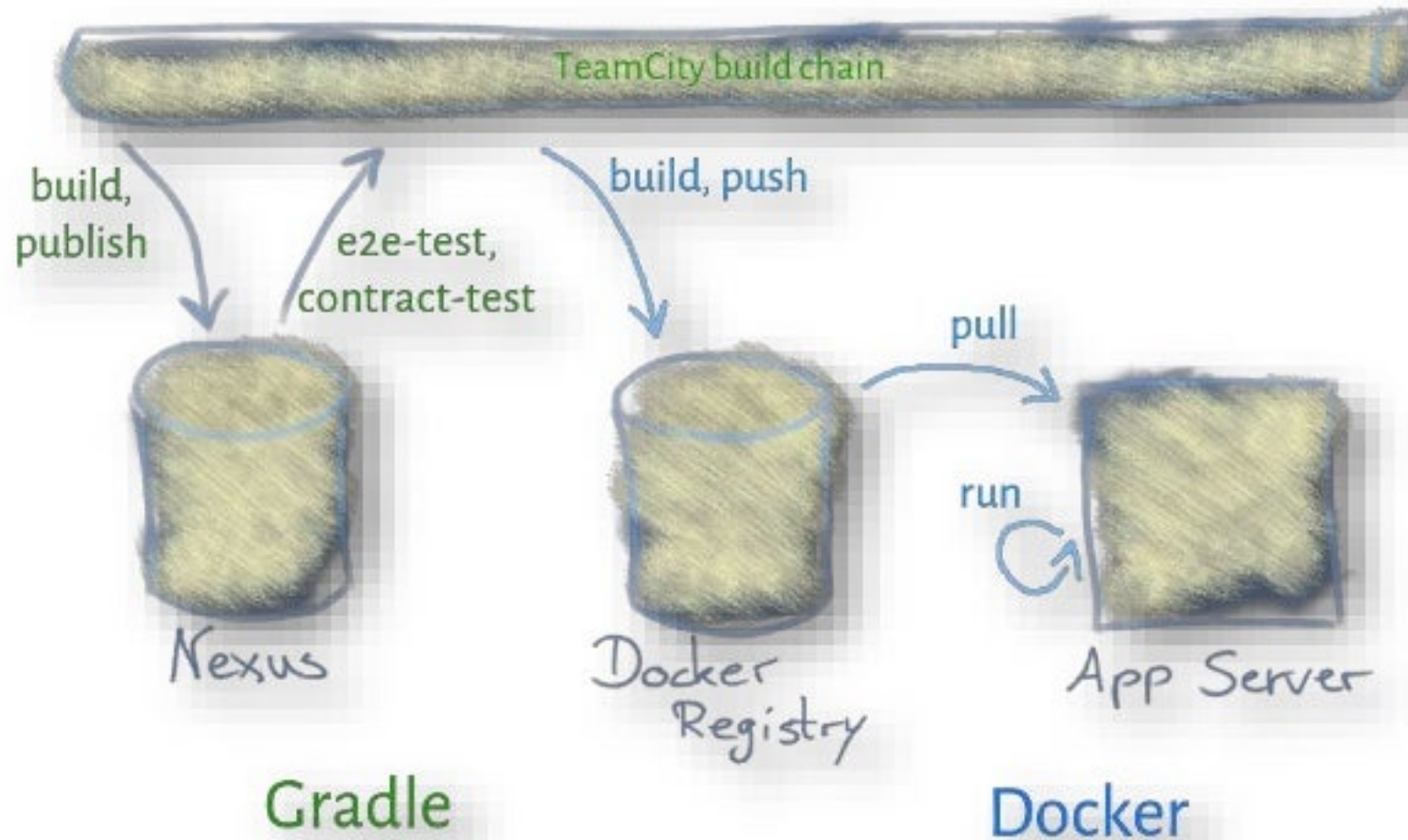


Connect Agile Upstream to Downstream: Example



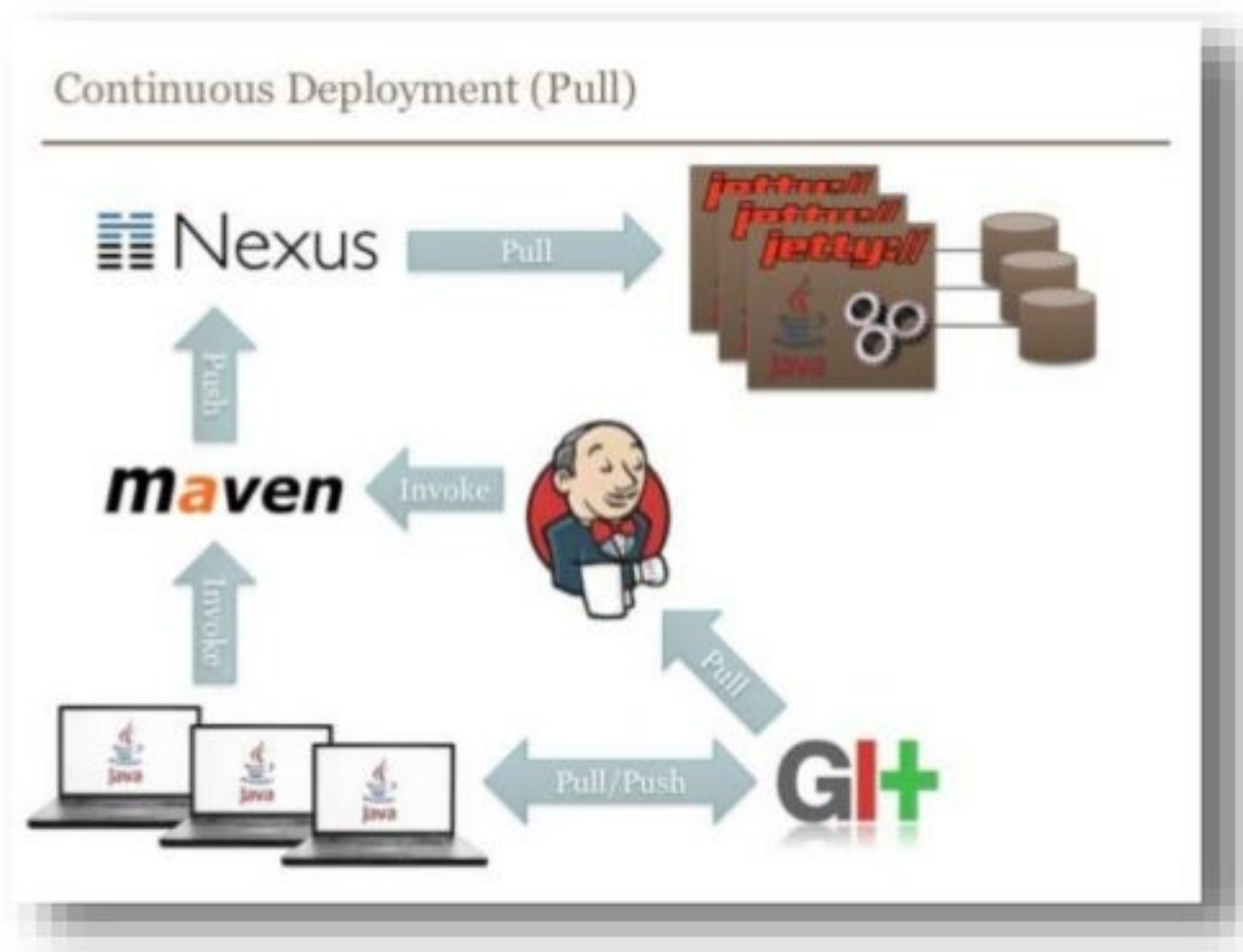
1. Dev, QA, and Ops implement user stories from TeamForge and commit changes to CM/DBT, the user stories and code changes are associated.
2. A CI build job runs, executing and tests, code analysis, etc. a TeamForge build artifact is created, change notes are generated and associated.
3. Every 15 minutes or on demand a snapshot build is published to the TeamForge repository. The snapshot build is deployed to the Development servers by Jenkins.
4. The Project Manager reviews the Change Set and Release folder in TeamForge and using the data decides to promote a build as a Release Candidate, she creates a Release Manager Artifact (RMA) and sets it to "Ready to Release".
5. A Release Candidate Promotion job runs executing the application build, publishing the artifacts to DevOps and providing Jenkins with the release note data.
6. An Jenkins package references the DevOps release holding the release candidate to be deployed.
7. Operations, QA and/or Change Management promote and deploy the release candidate using Jenkins Workflow.
8. At each promotion and deployment step the TeamForge Release Artifact is updated, from Jenkins and team members receive email and feedback.
9. The Release Dashboard in TeamForge provides up-to-date project Activity Report and Release Notes.
10. Dev, QA, Ops and Management can view the Release Dashboard in TeamForge to track activity and make informed decisions.

According to Hypoport AG

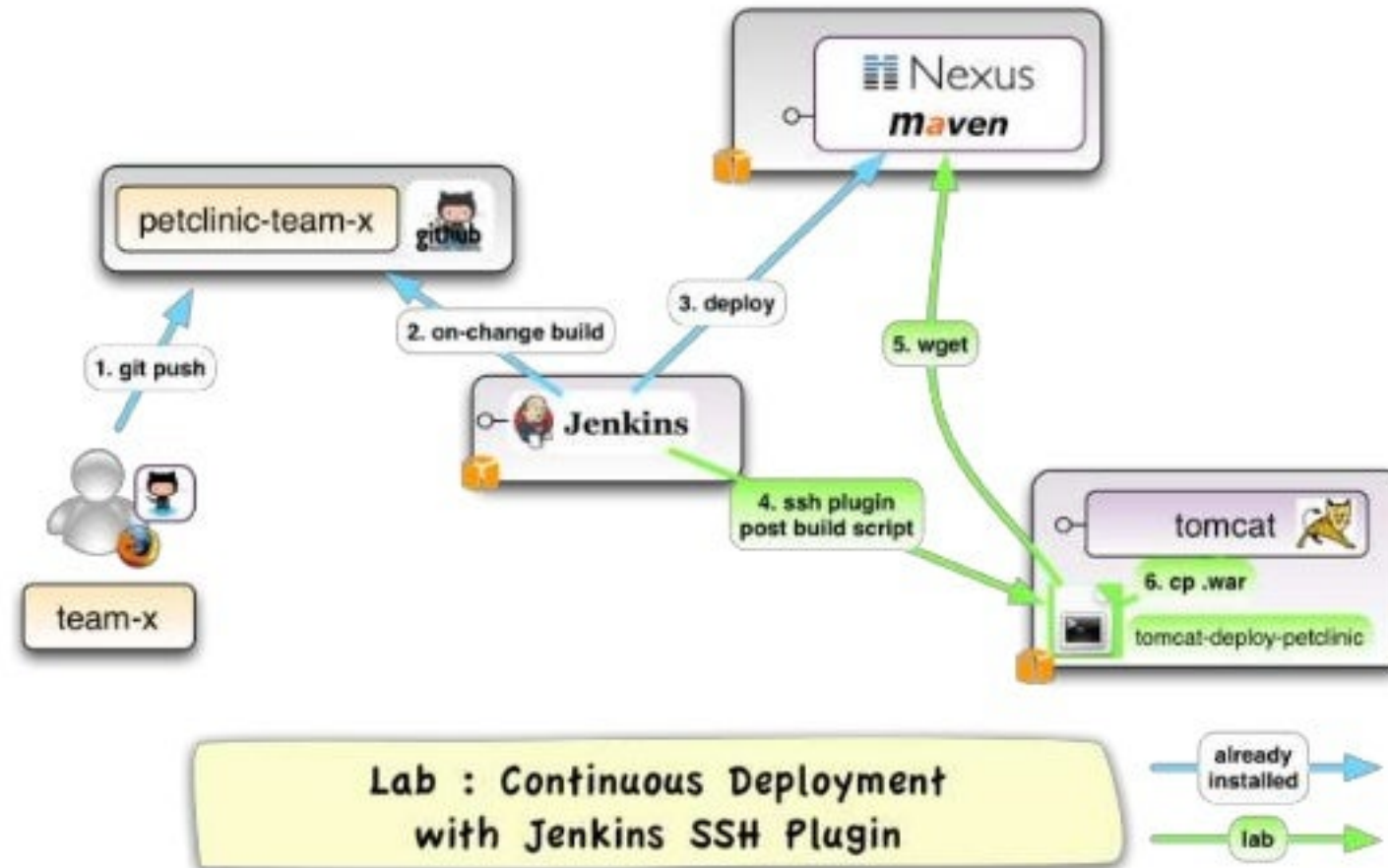


According to BEKK

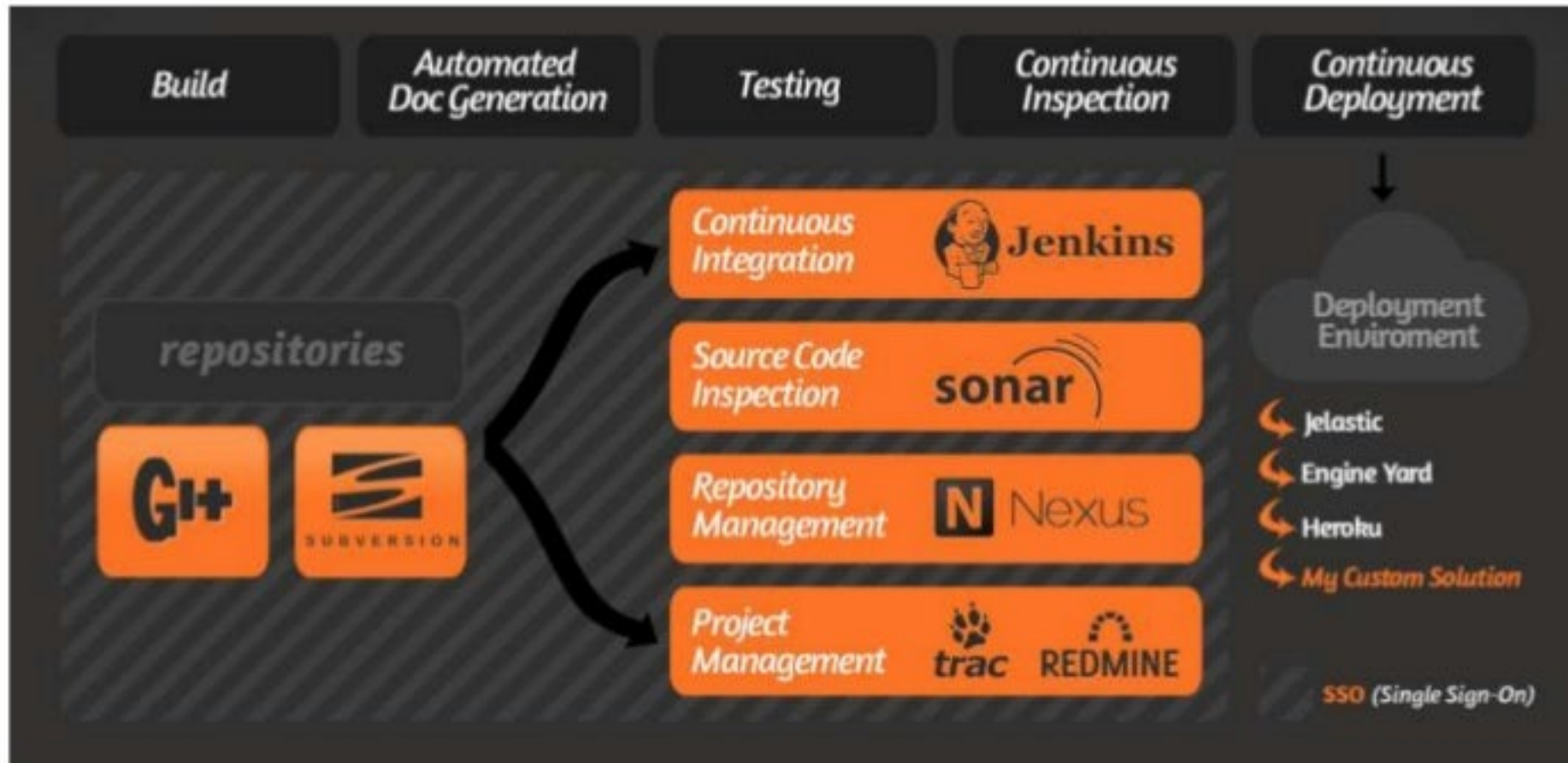
BEKK



According to Xebia



According to ClinkerHQ



According to Zanox

zanax.

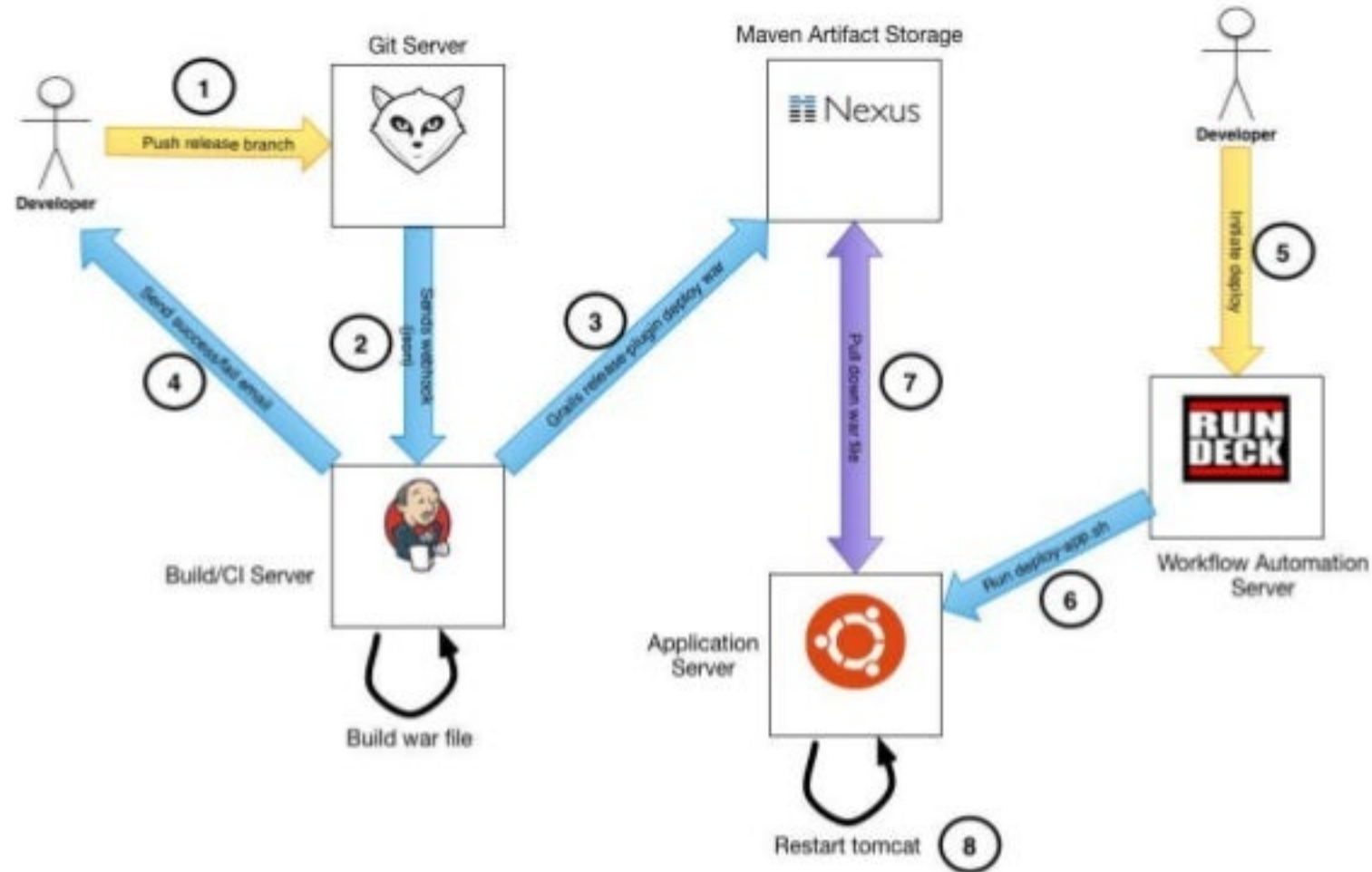


WJAXIS | Business Technology Days

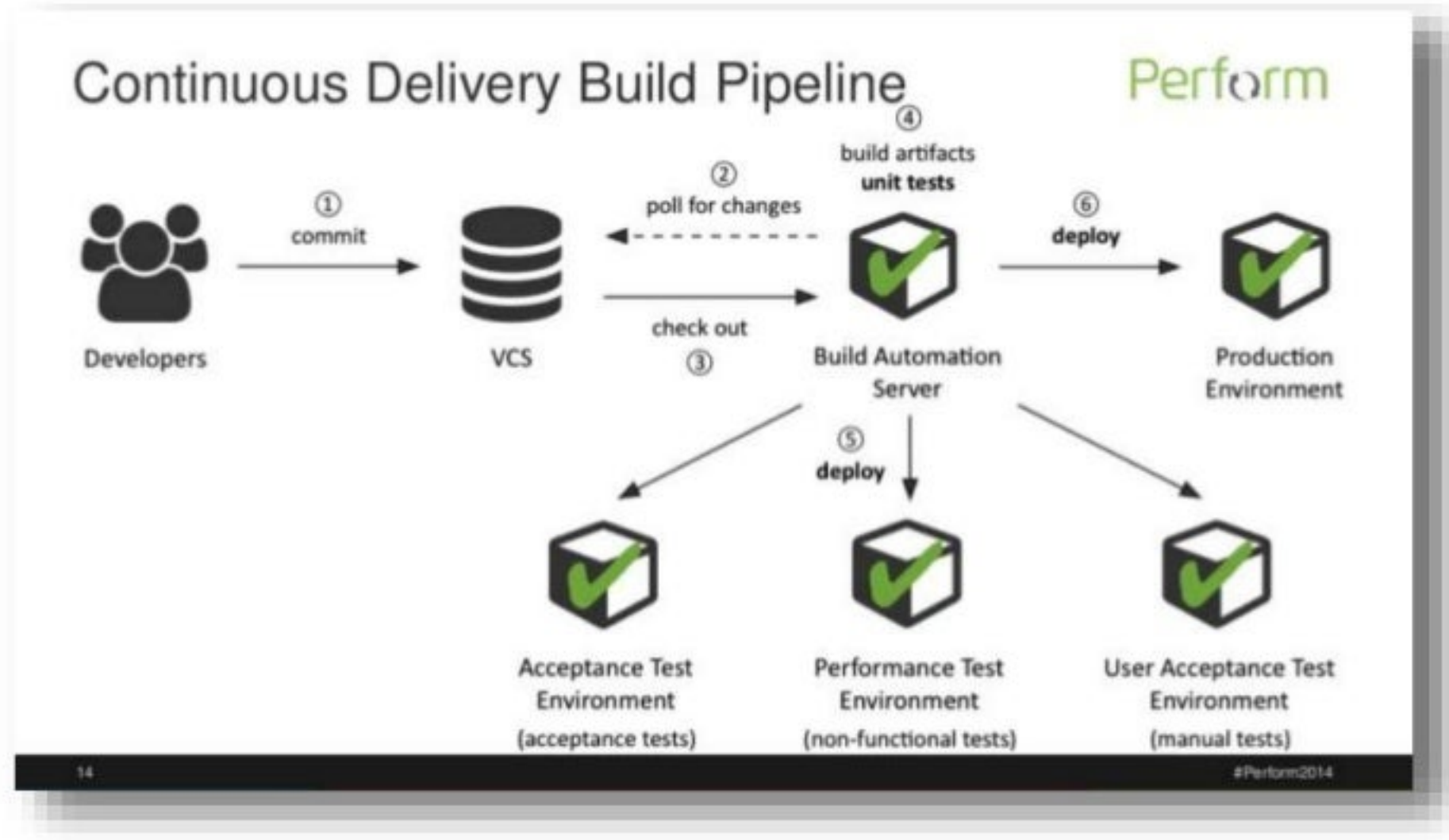
Chef & JBoss

- Fabric
 - Python library and command-line tool
 - Streamlining the use of SSH
 - Application deployment
 - Used to deploy apps from Nexus
 - <https://github.com/fabric/fabric> 

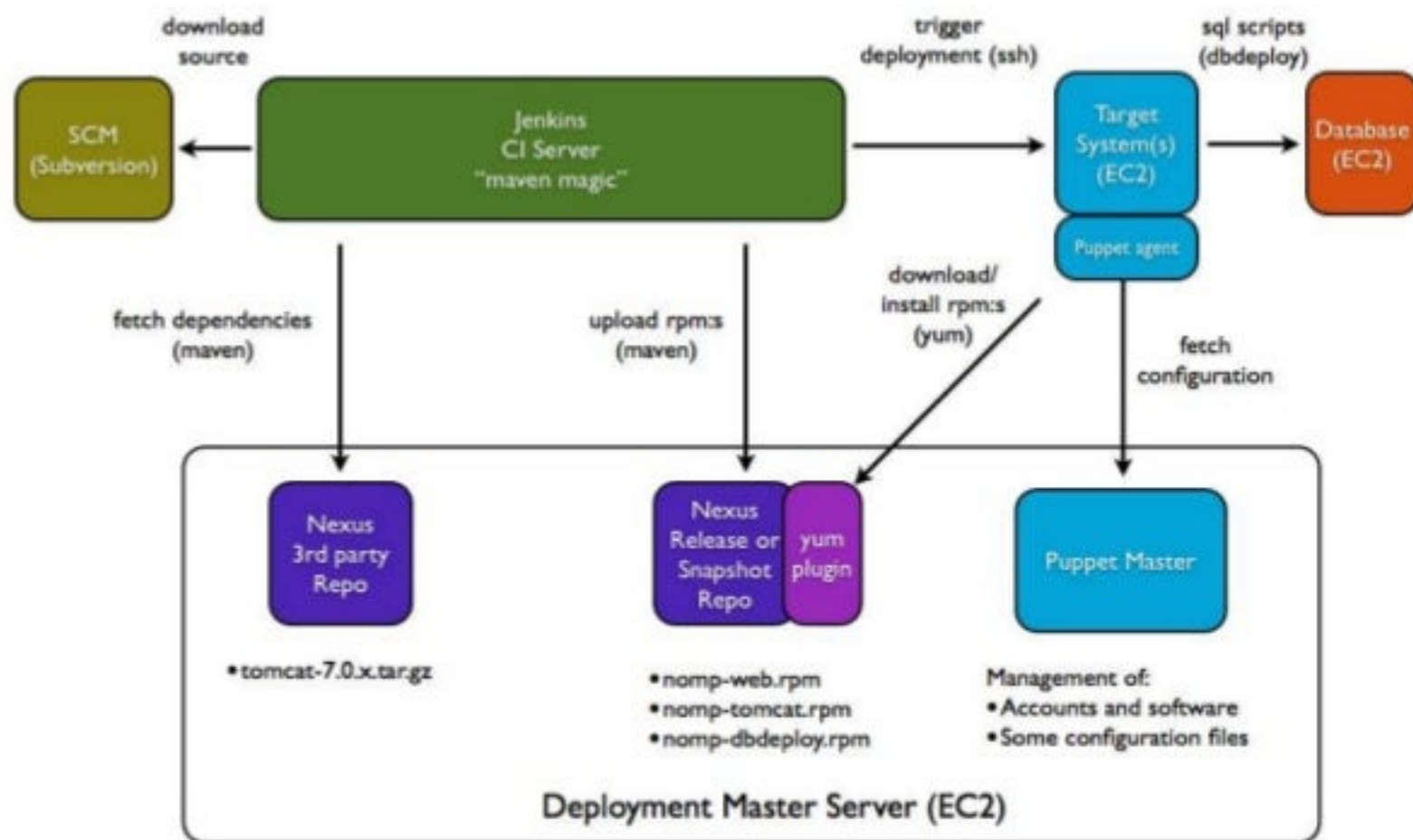
According to Riverside I/O



According to Dynatrace

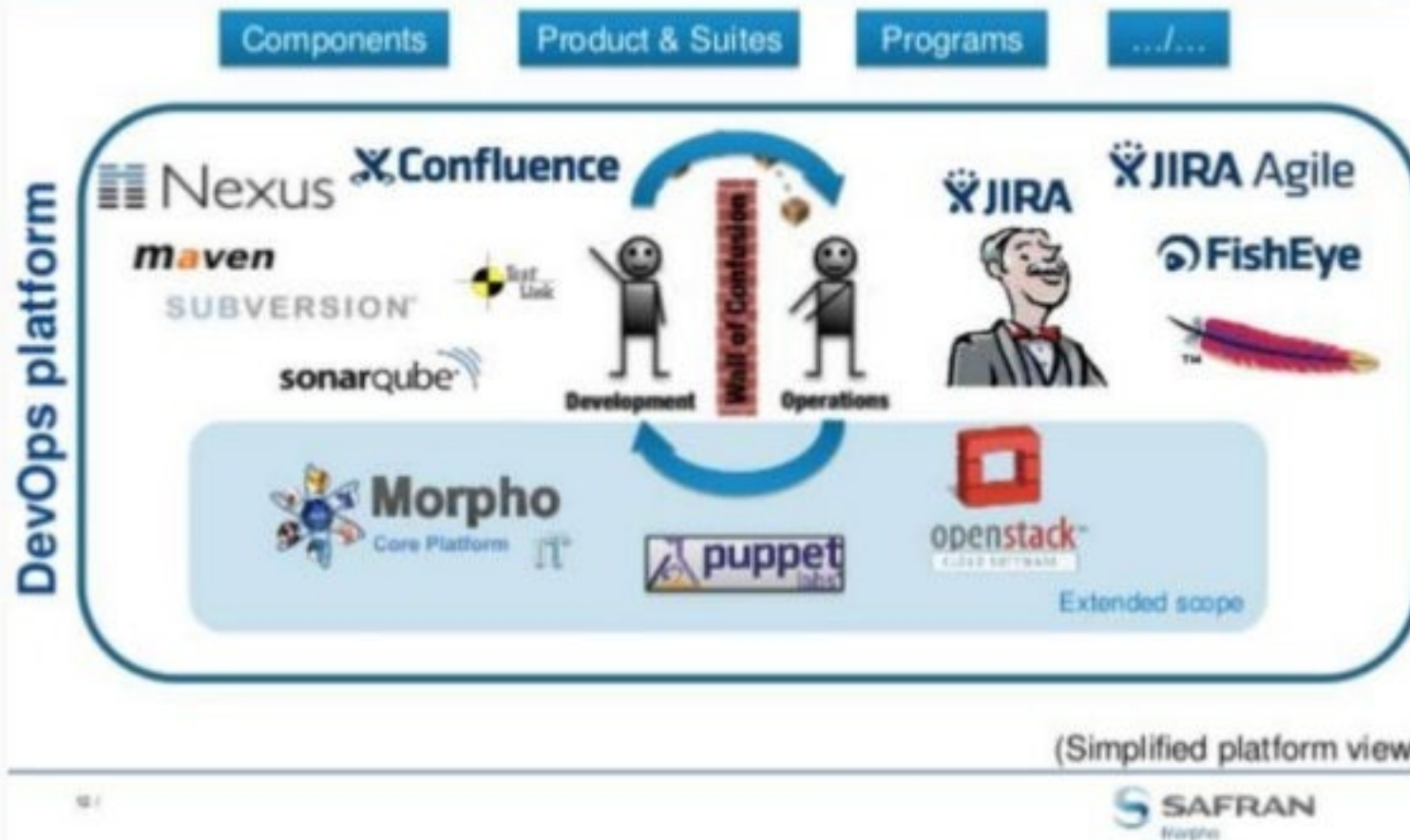


According to Stefan Norberg

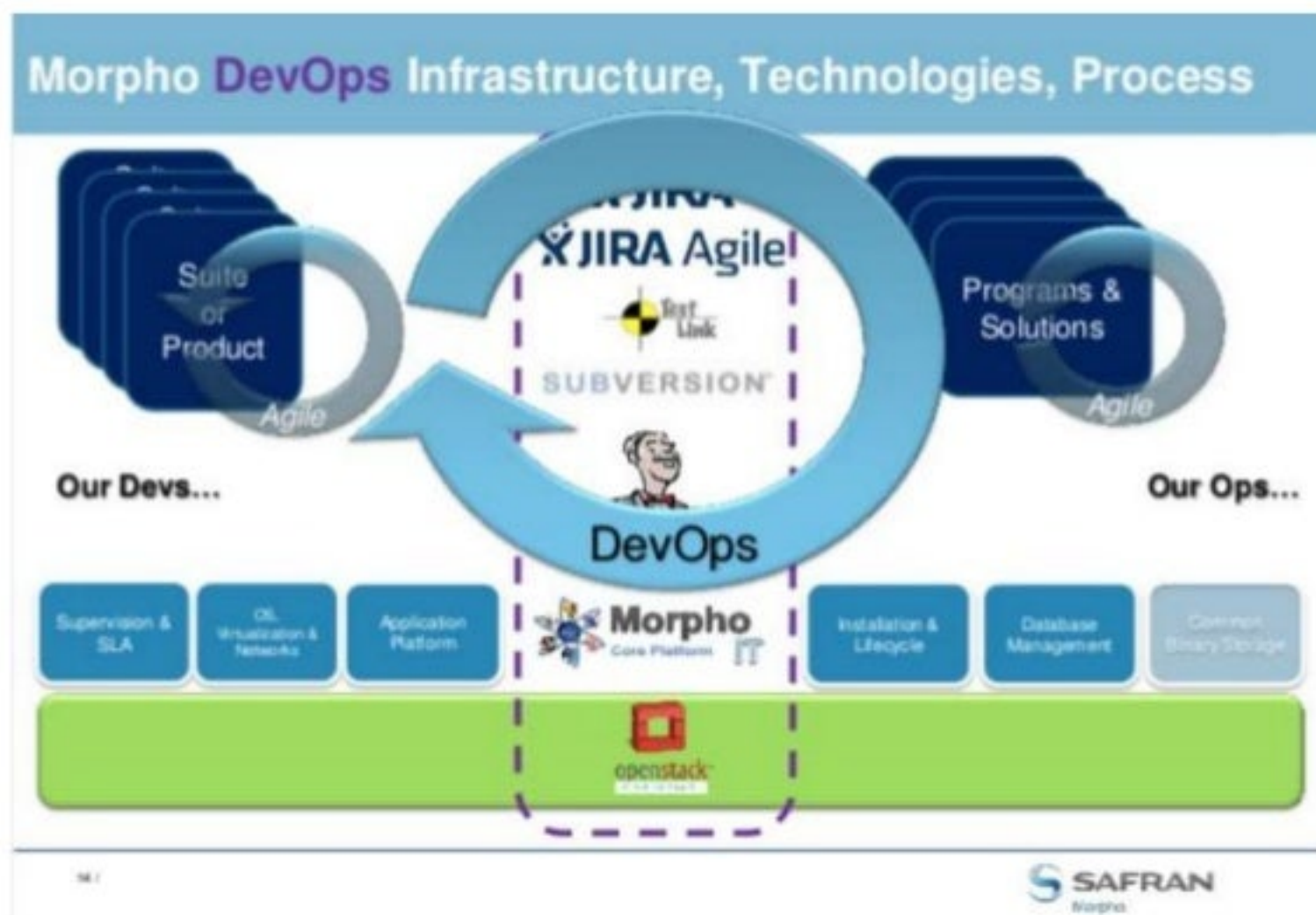


Nexus at Morpho

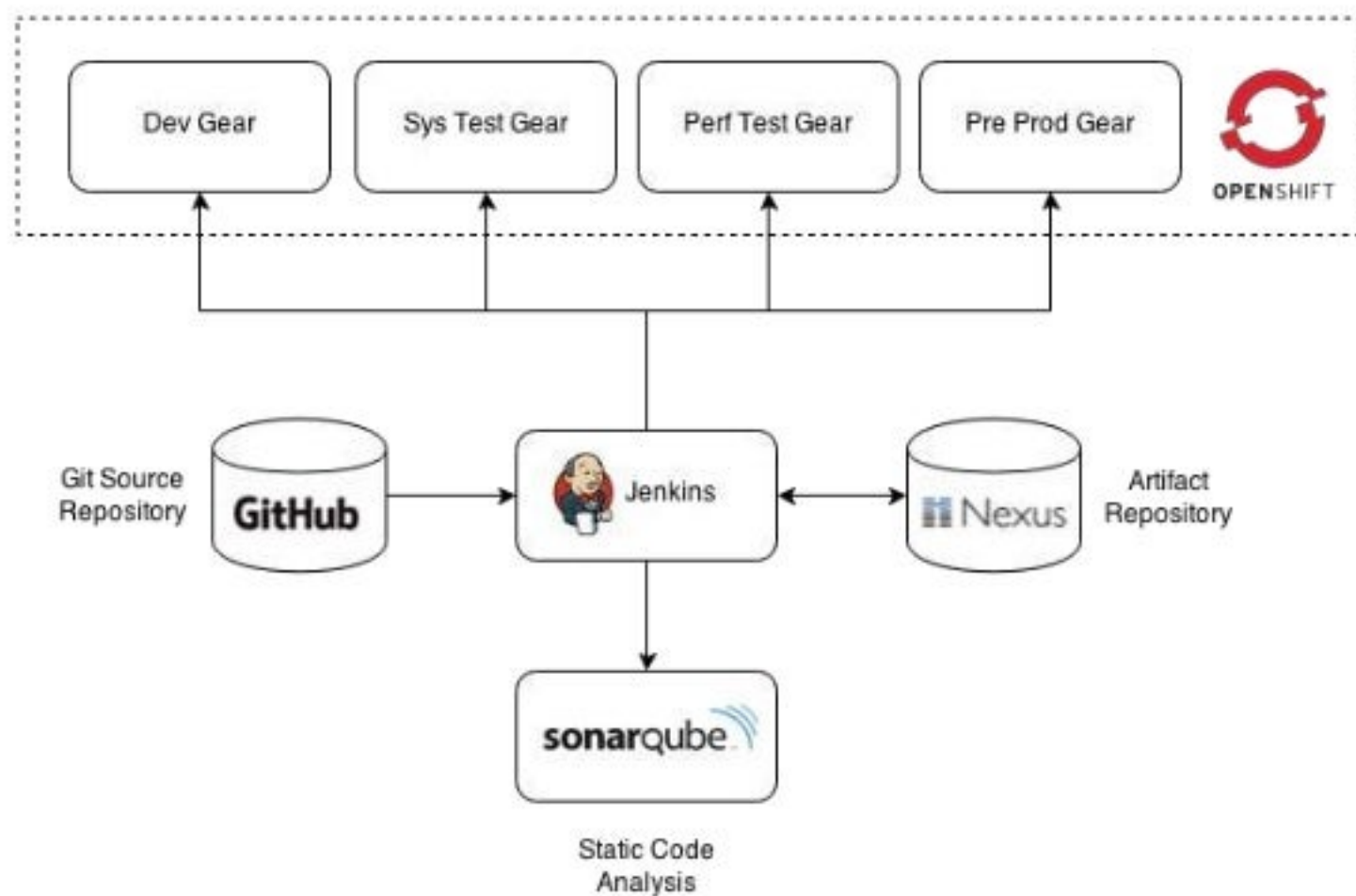
Mid-2013 : First **DevOps** platform concept



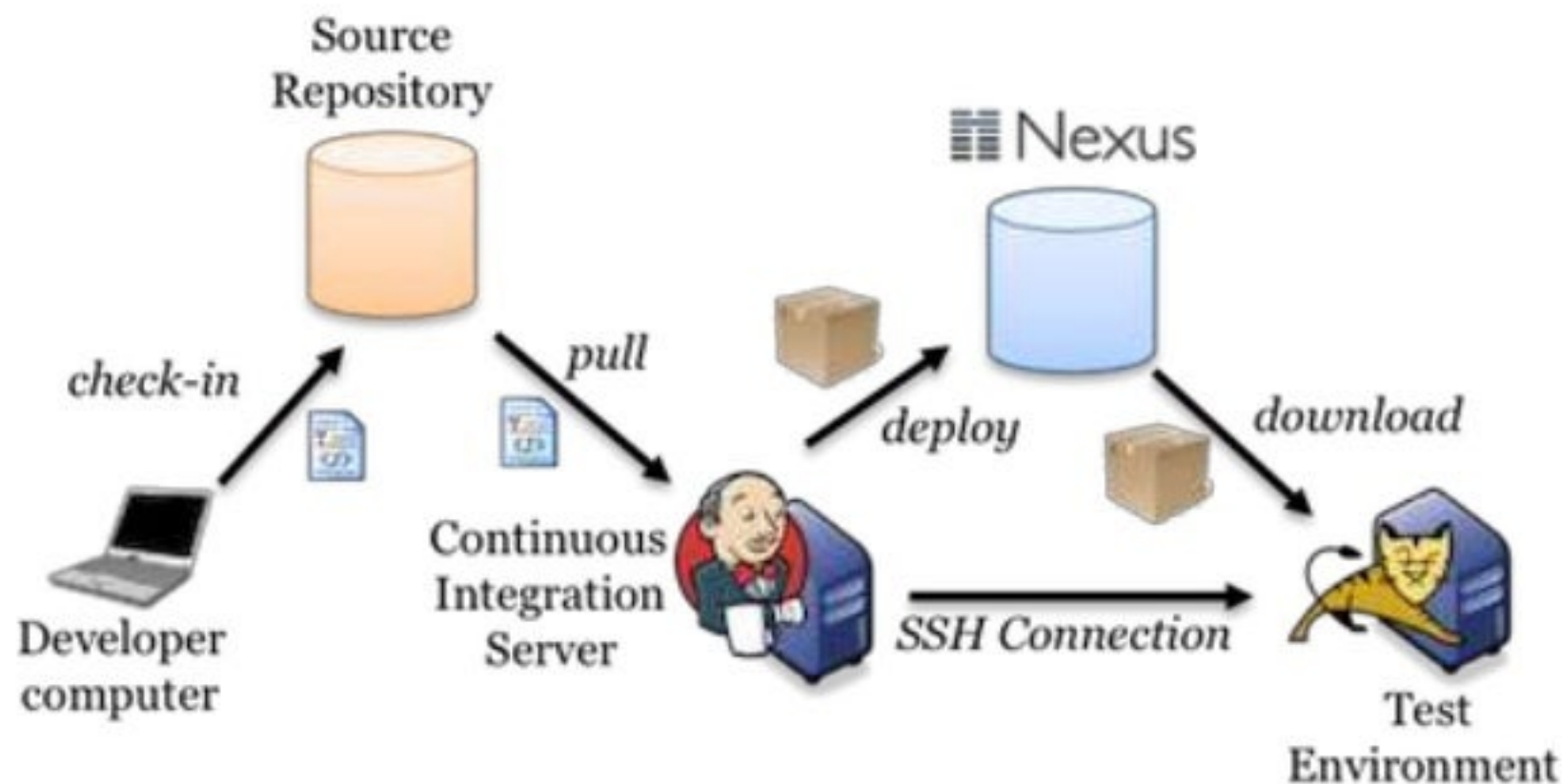
According to Morpho



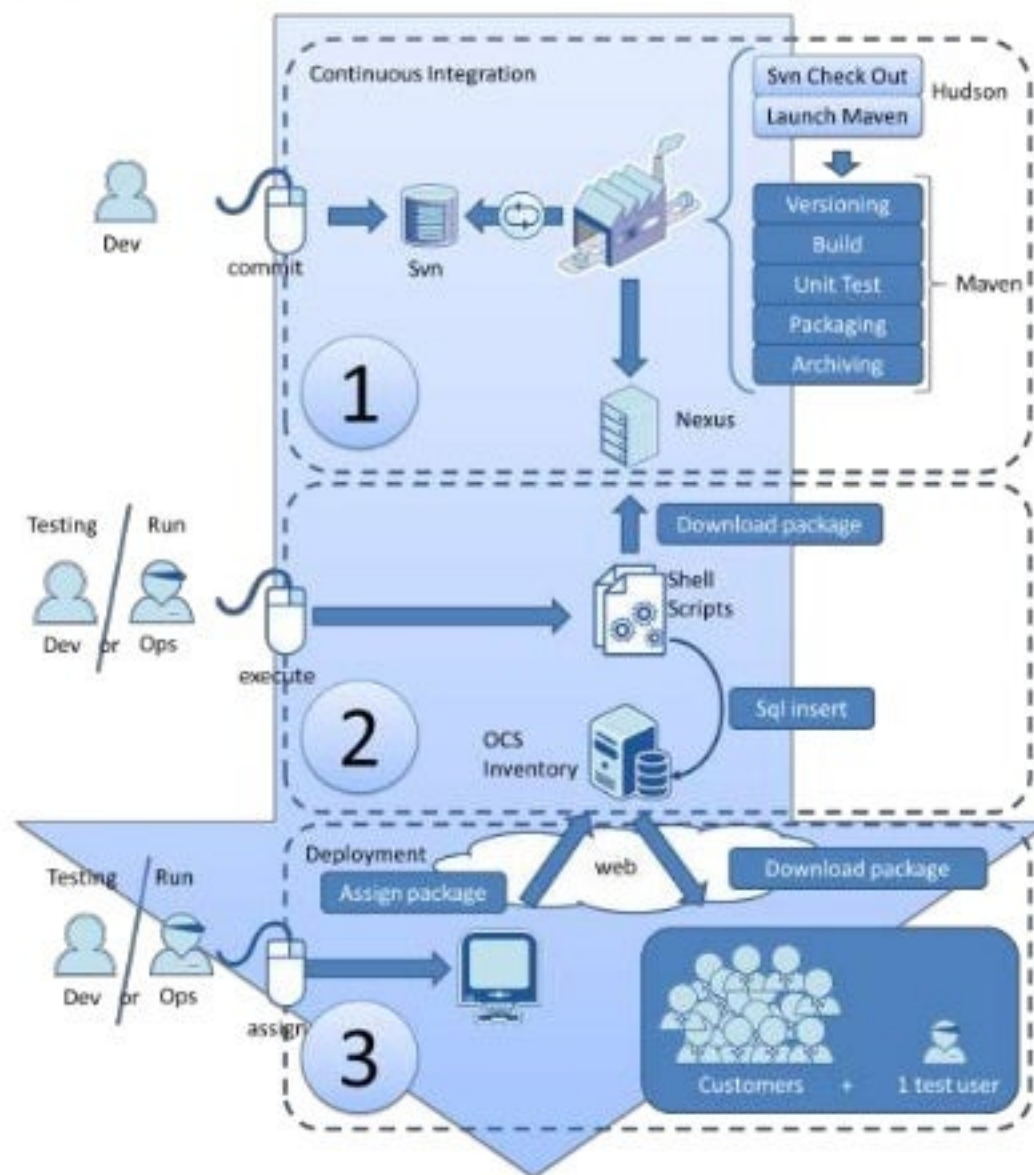
According to OPENSIFT



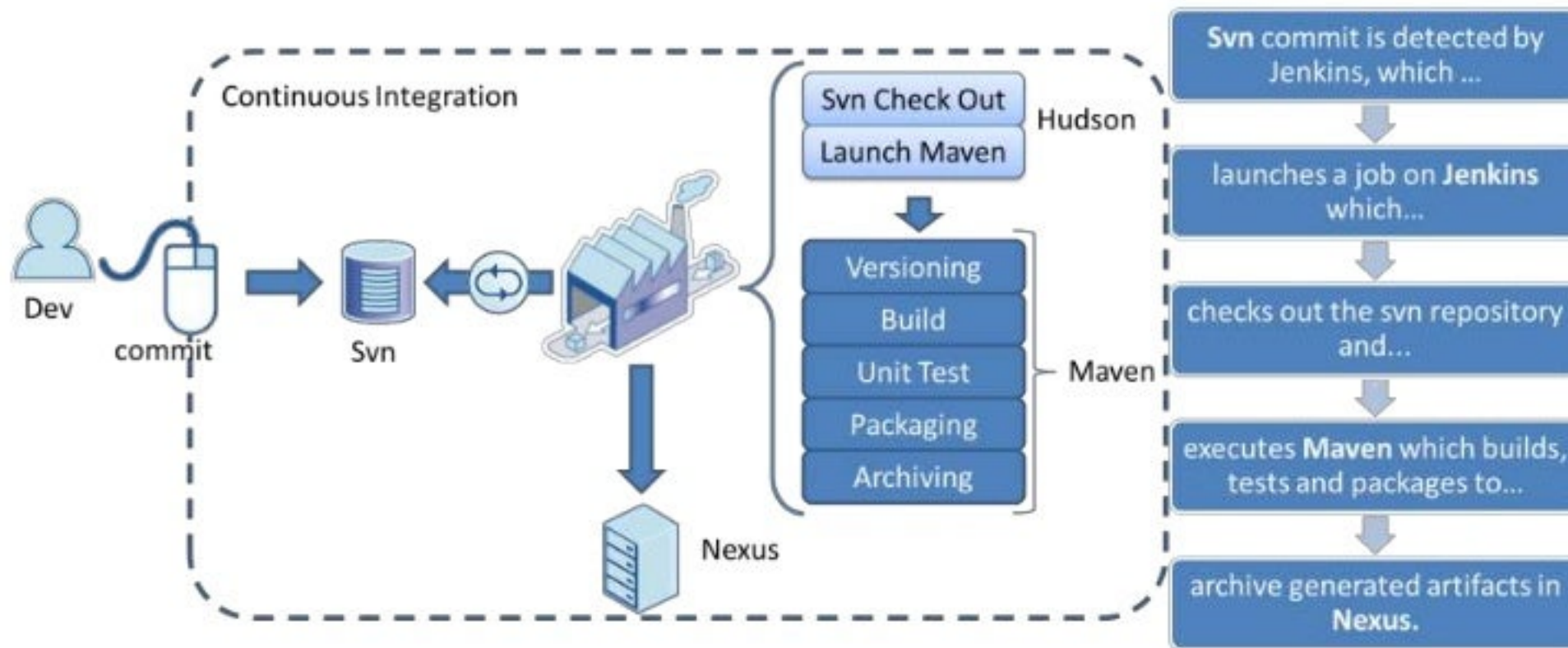
According to akquinet



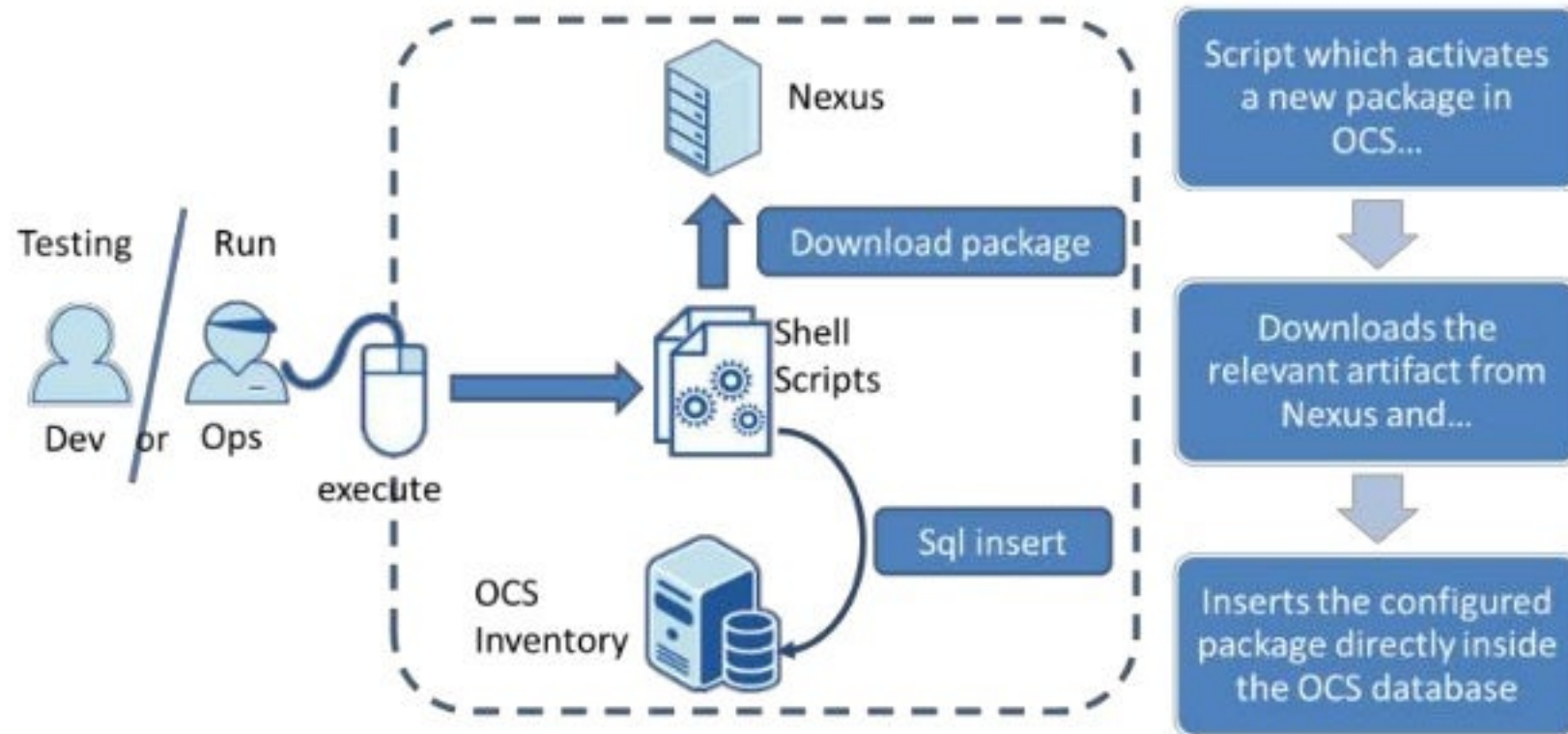
According to OCTO



According to OCTO



According to OCTO



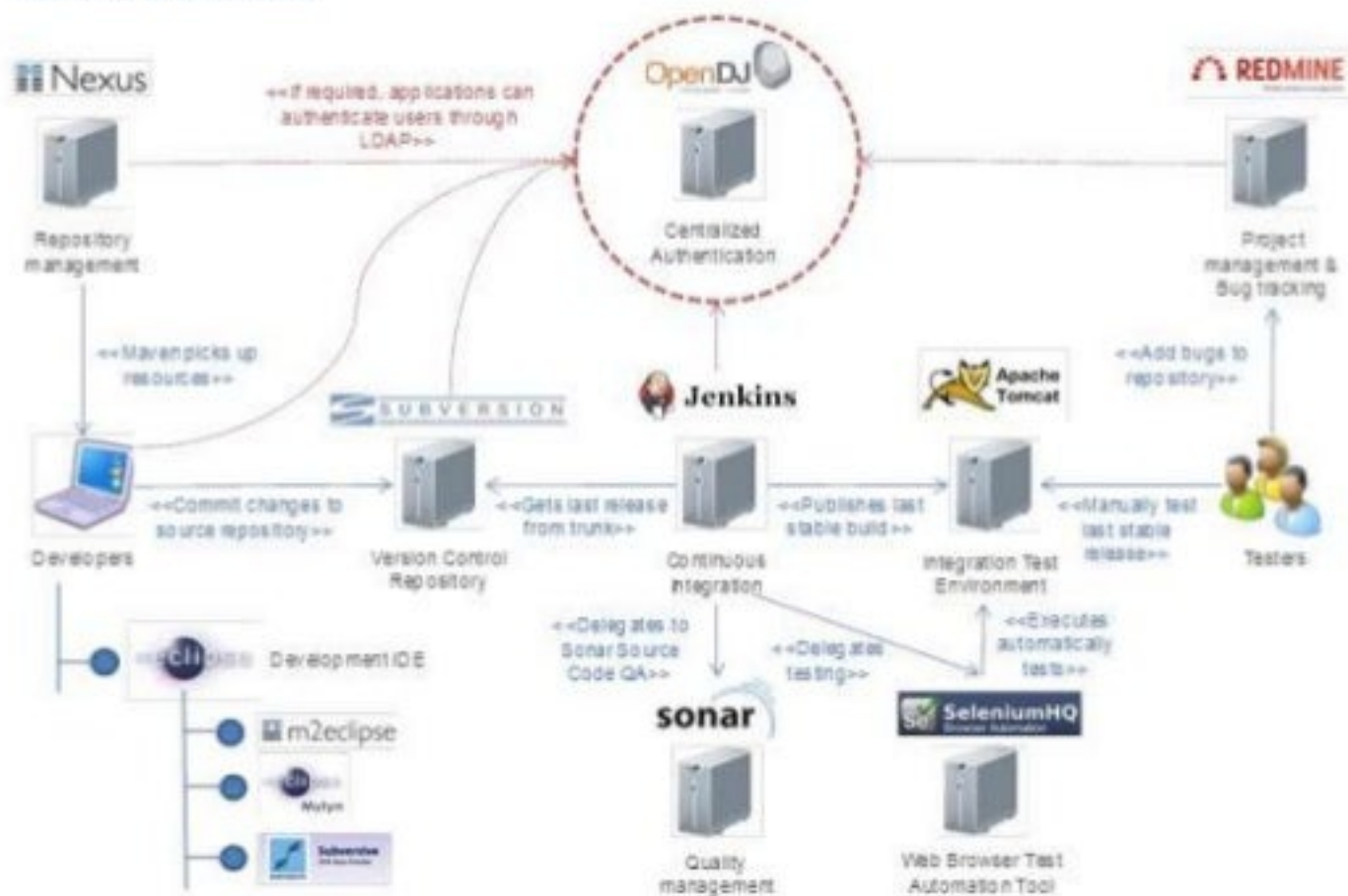
According to Jordi Cuenca-Aubets

Free continuous integration platform @Glance

Runtime architecture

Who Am I?

Jordi Cuenca Aubets (LinkedIn)
@jordicua (Twitter)



OPEN SOURCERERS

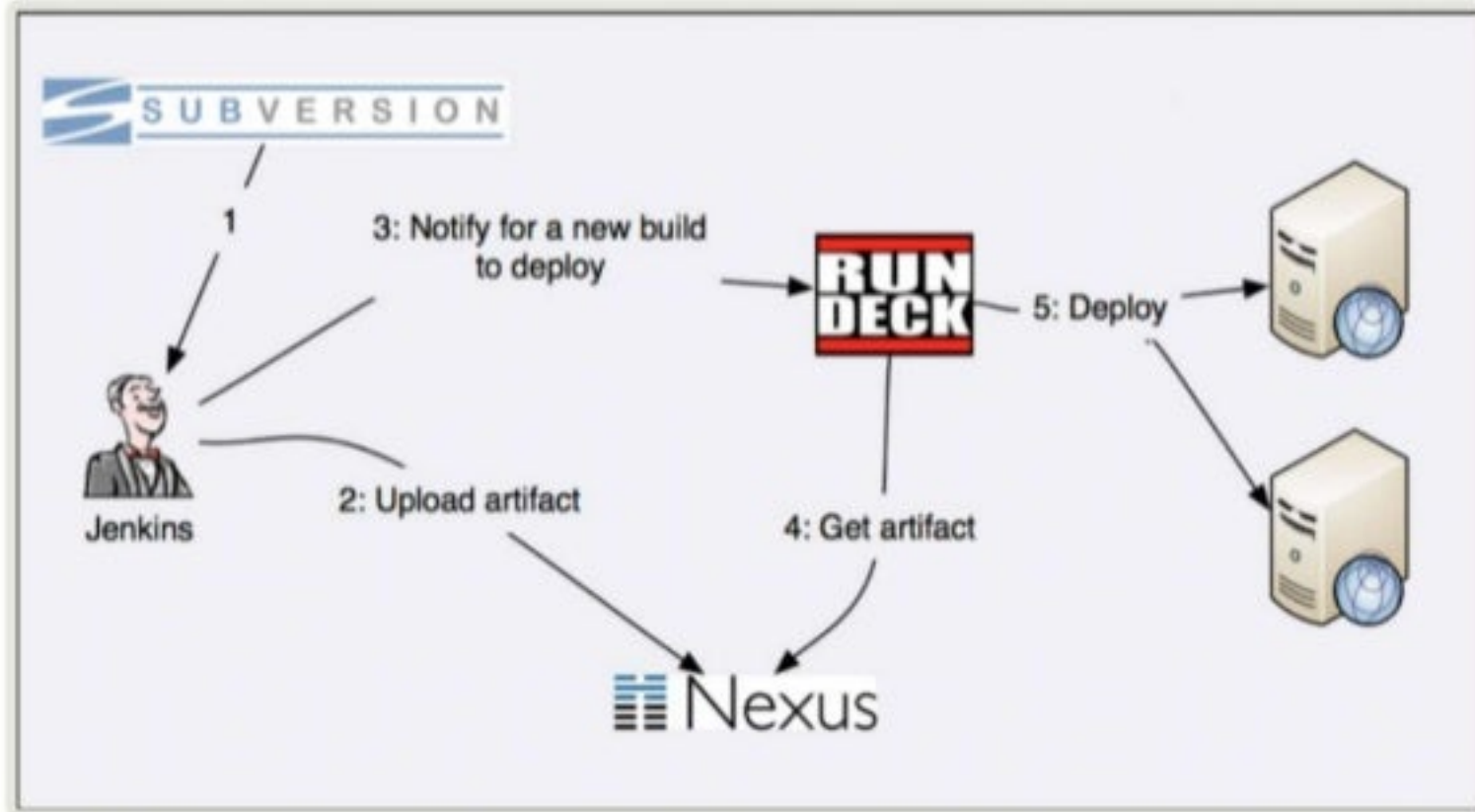


<http://bit.ly/1WNMIq0>

According to Michael Rumpf



According to Cardlife



According to Paolo Antinori, RedHat

Let's start describing the component of our **sample Continuous Integration setup**:

1) JBoss Fuse 6.1

It's the **runtime** we are going to deploy onto. It lives in a dedicated box. It interacts with *Nexus* as the source of the artifacts we produce and publish.

2) Nexus

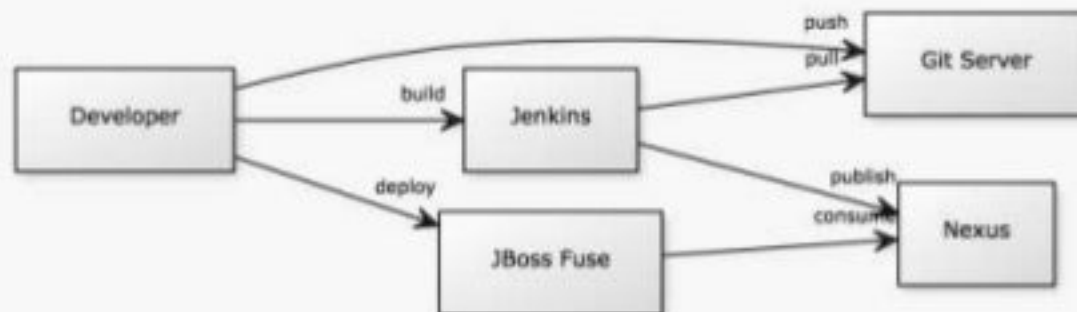
It's the software we use to **store the binaries** we produce from our code base. It is accessed by *JBoss Fuse*, that downloads artifacts from it but it is also accessed from *Jenkins*, that publishes binaries on it, as the last step of a successful build job.

3) Jenkins

It's our **build jobs invoker**. It publishes its outputs to *Nexus* and it builds its output if the code it checked out with *Git* builds successfully.

4) Git Server

It's the **remote code repository** holder. It's accessed by *Jenkins* to download the most recent version of the code we want to build and it's populated by all the developers when they share their code and when they want to build on the Continuous Integration server. In our case, **git server is just a filesystem accessed via ssh**.



<http://yum1.mw/ed07e75feb05>

According to Atlassian



Build Engineering today @ Atlassian

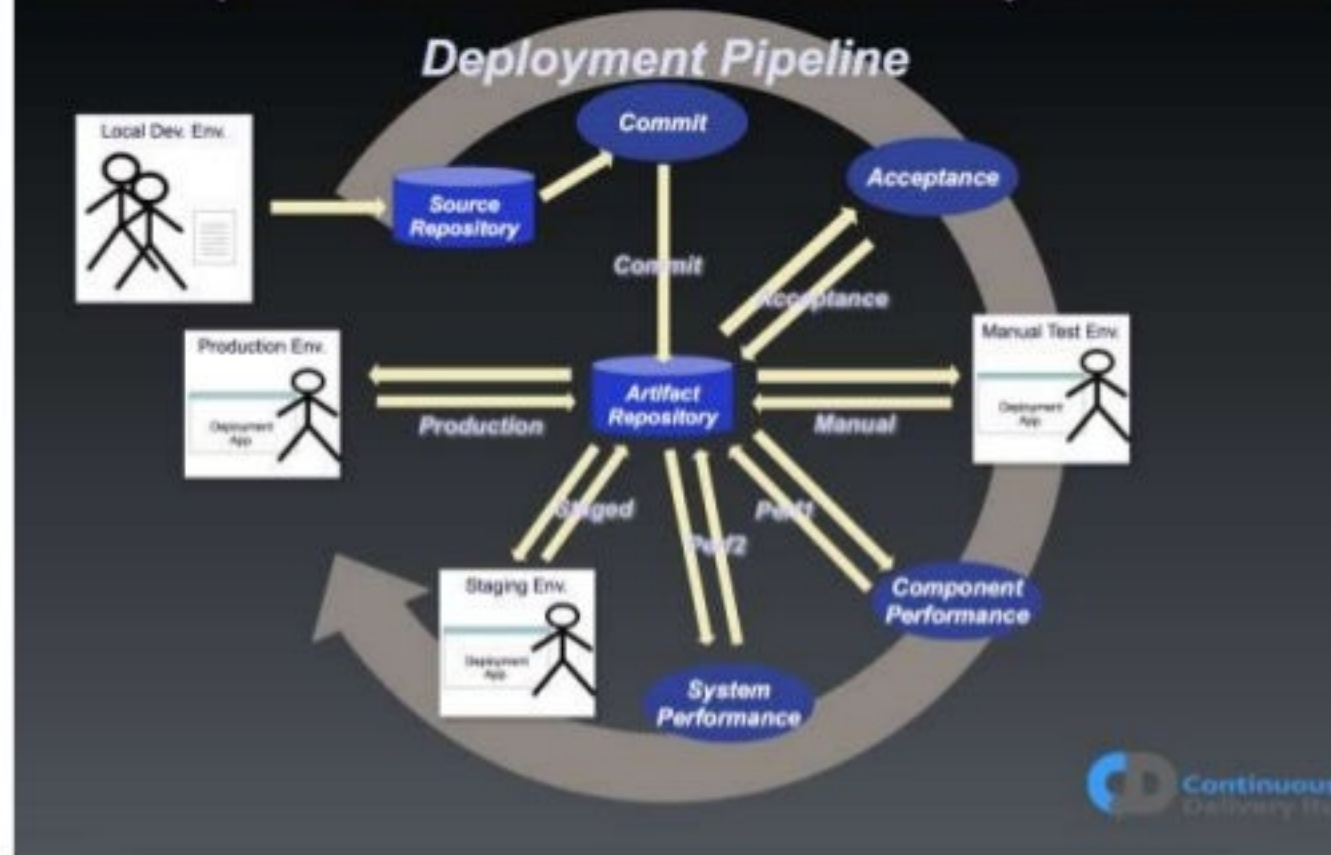


- 600 build agents (own hardware + EC2 instances)
 - include SCM clients, JDKs, JVM build tools, databases, headless browser testing, python builds, NodeJS, installers & more
- Maintain 20 AMIs of various build configurations
- 6 Bamboo Servers
- maven.atlassian.com / 6 Nexus instances
- Monitoring - opsview / graphite / statsd

According to Continuous Delivery Ltd

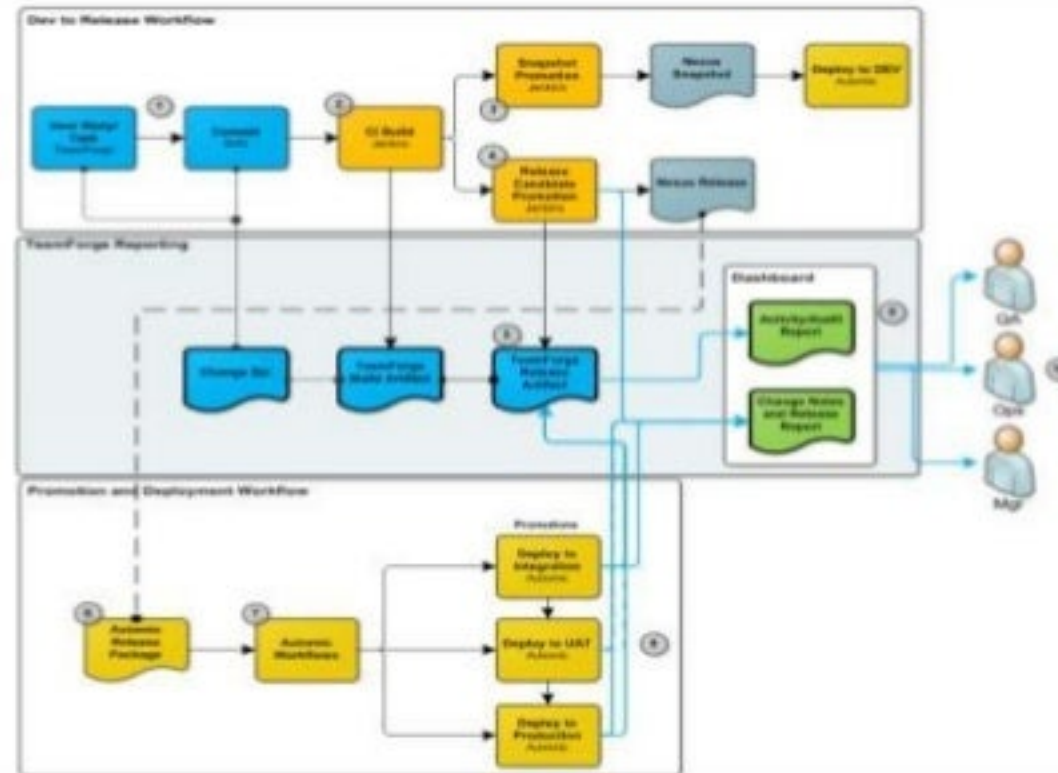


Example Continuous Delivery Process



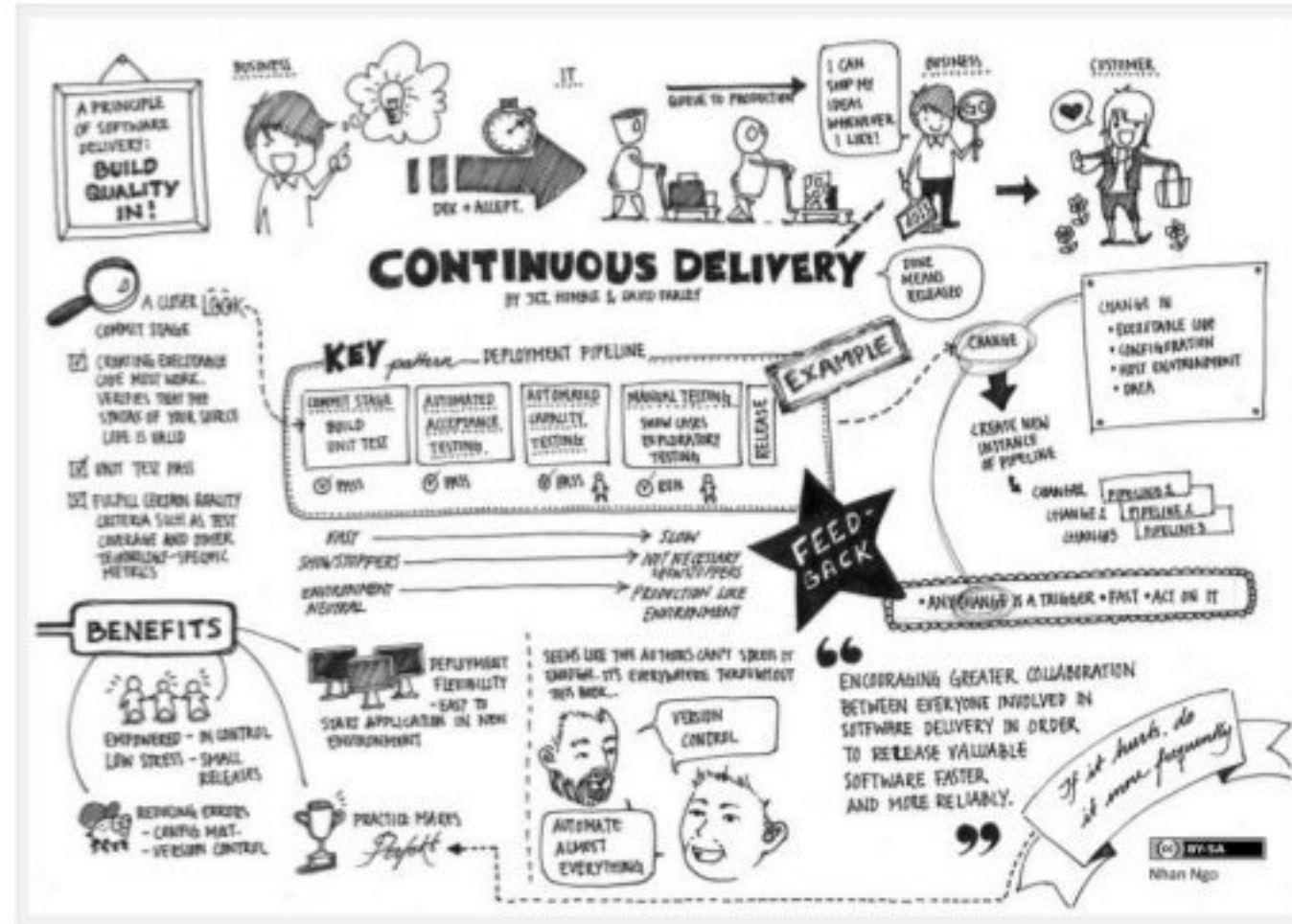
According to CollabNet

A scalable DevOps reference architecture



Copyright ©2014 CollabNet, Inc. All Rights Reserved

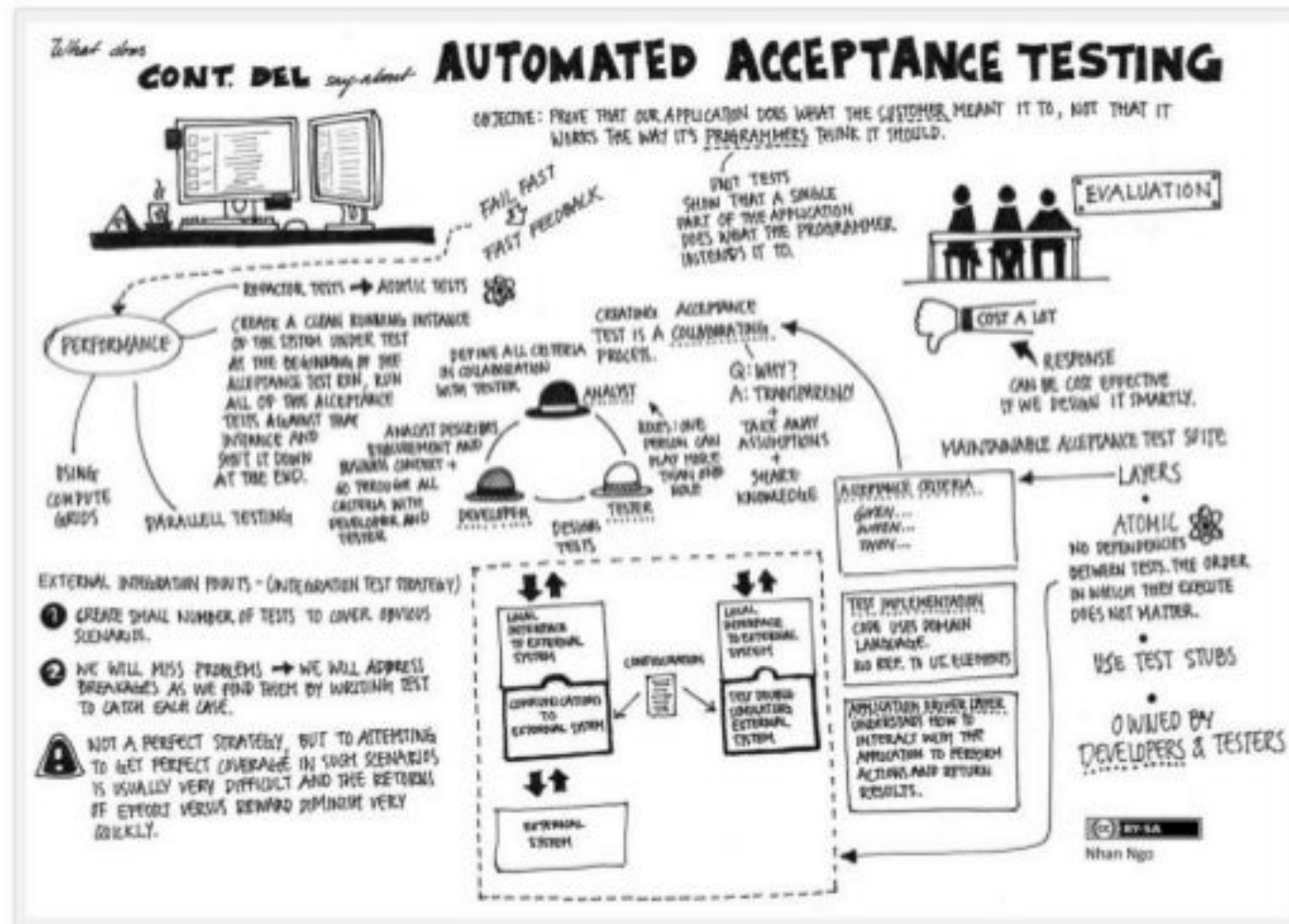
According to Nhan Ngo



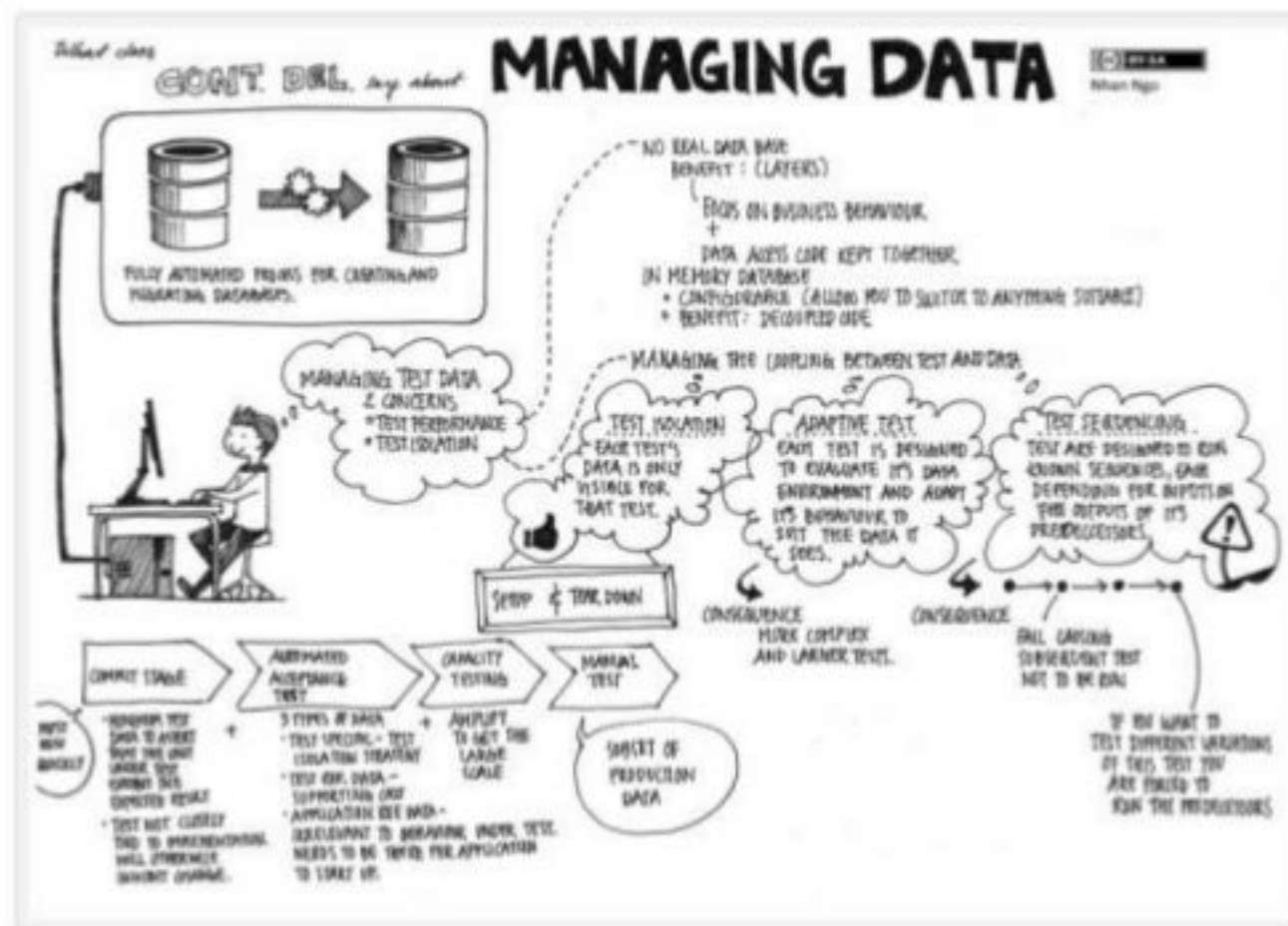
According to Nhan Ngo



According to Nhan Ngo



According to Nhan Ngo



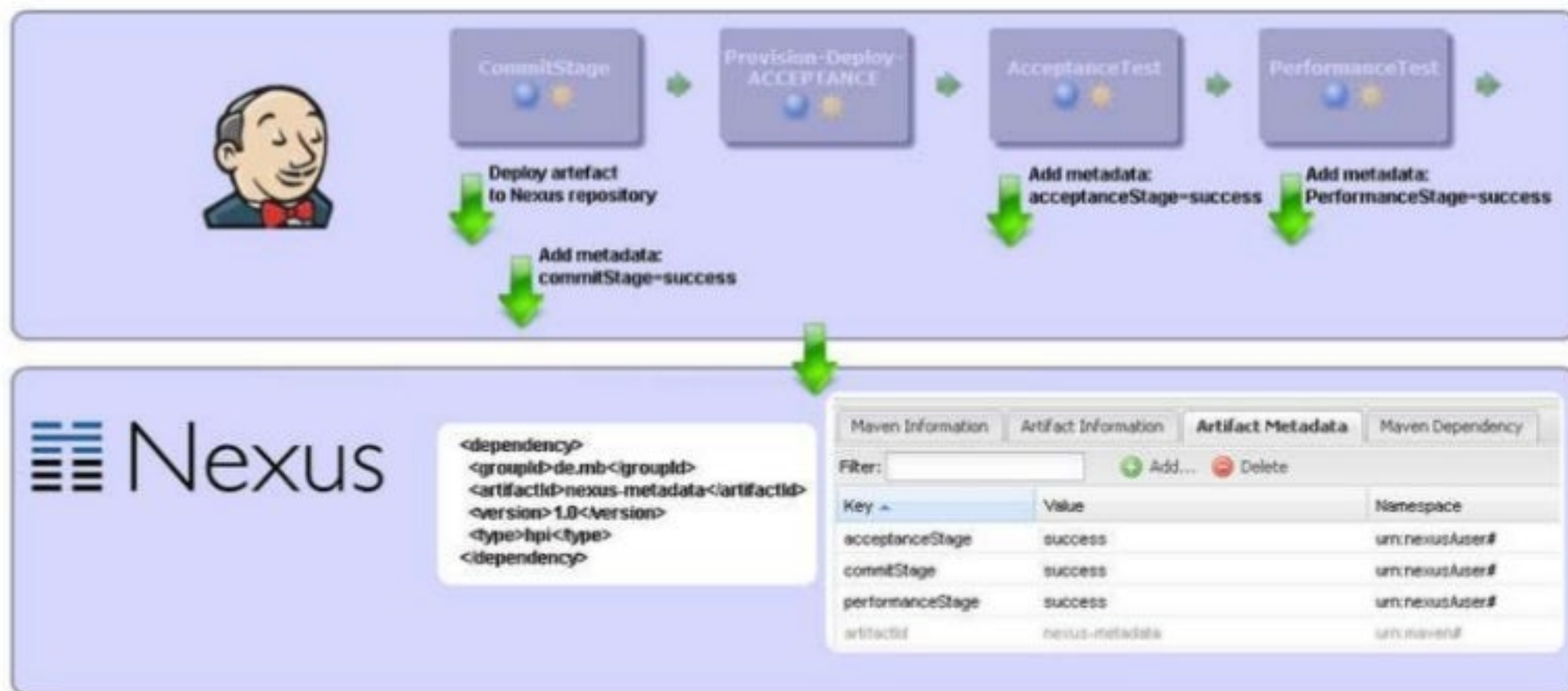
According to WS02



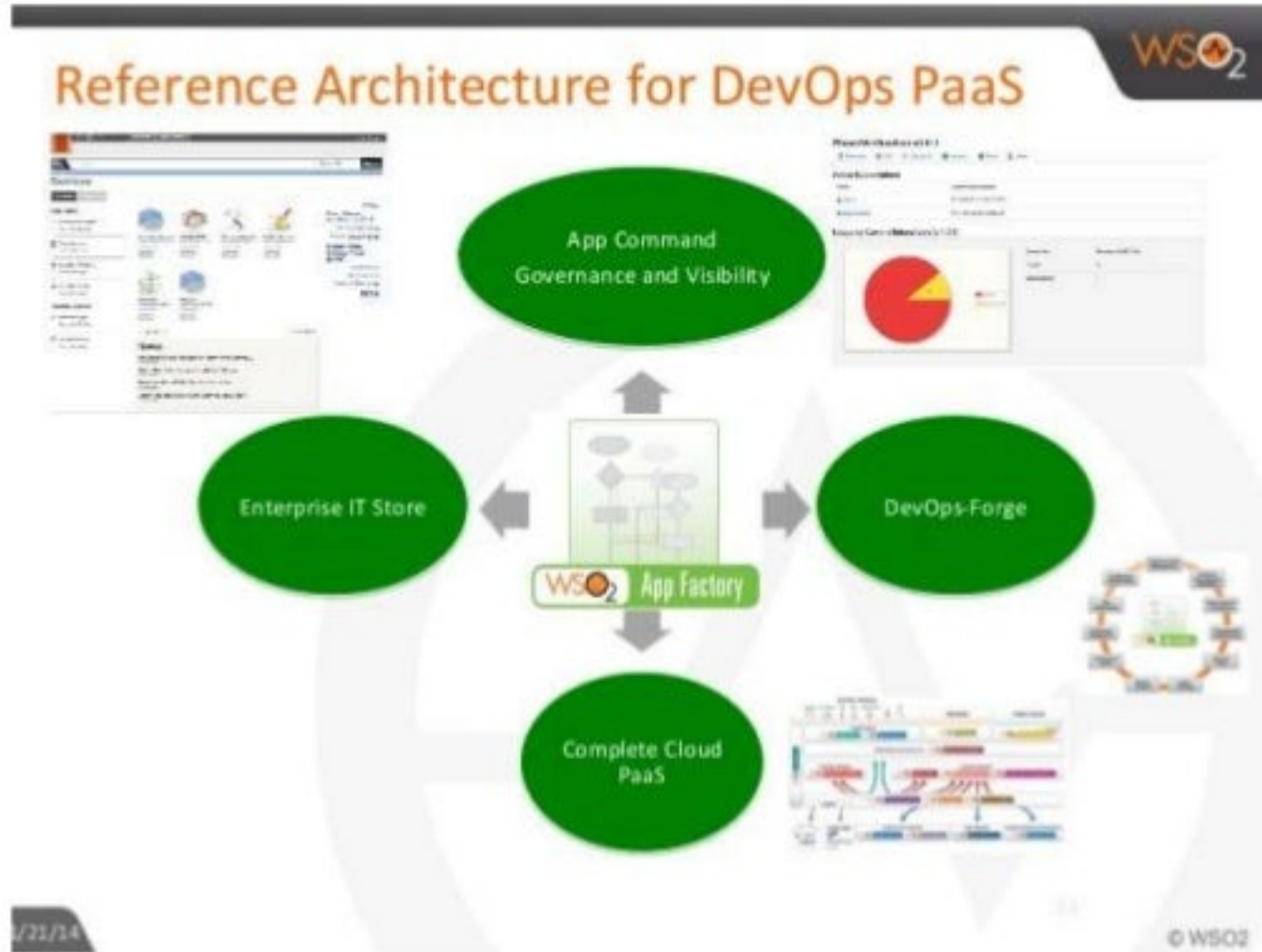
A Reference Architecture DevOps both dev and ops perspective



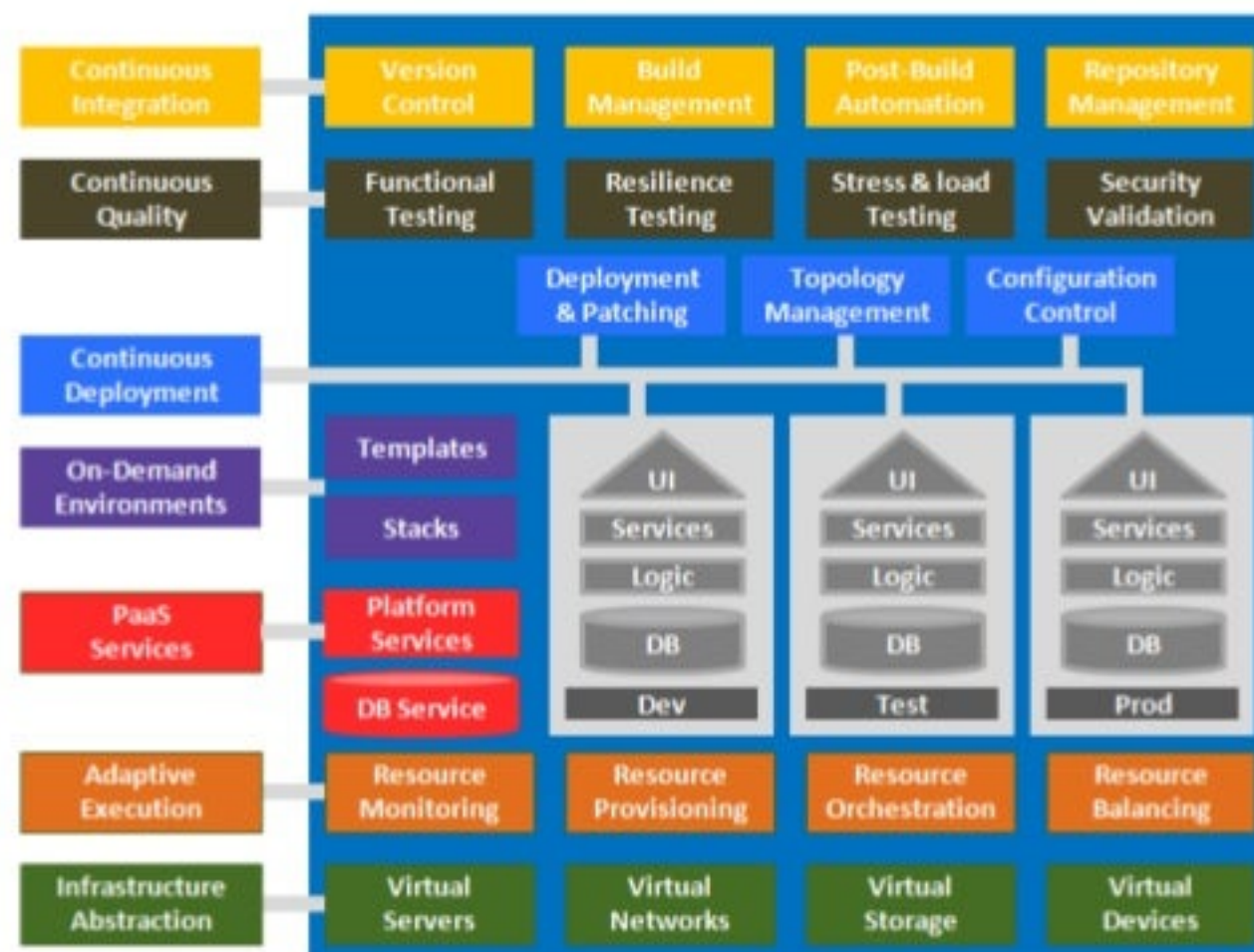
According to CodeCentric



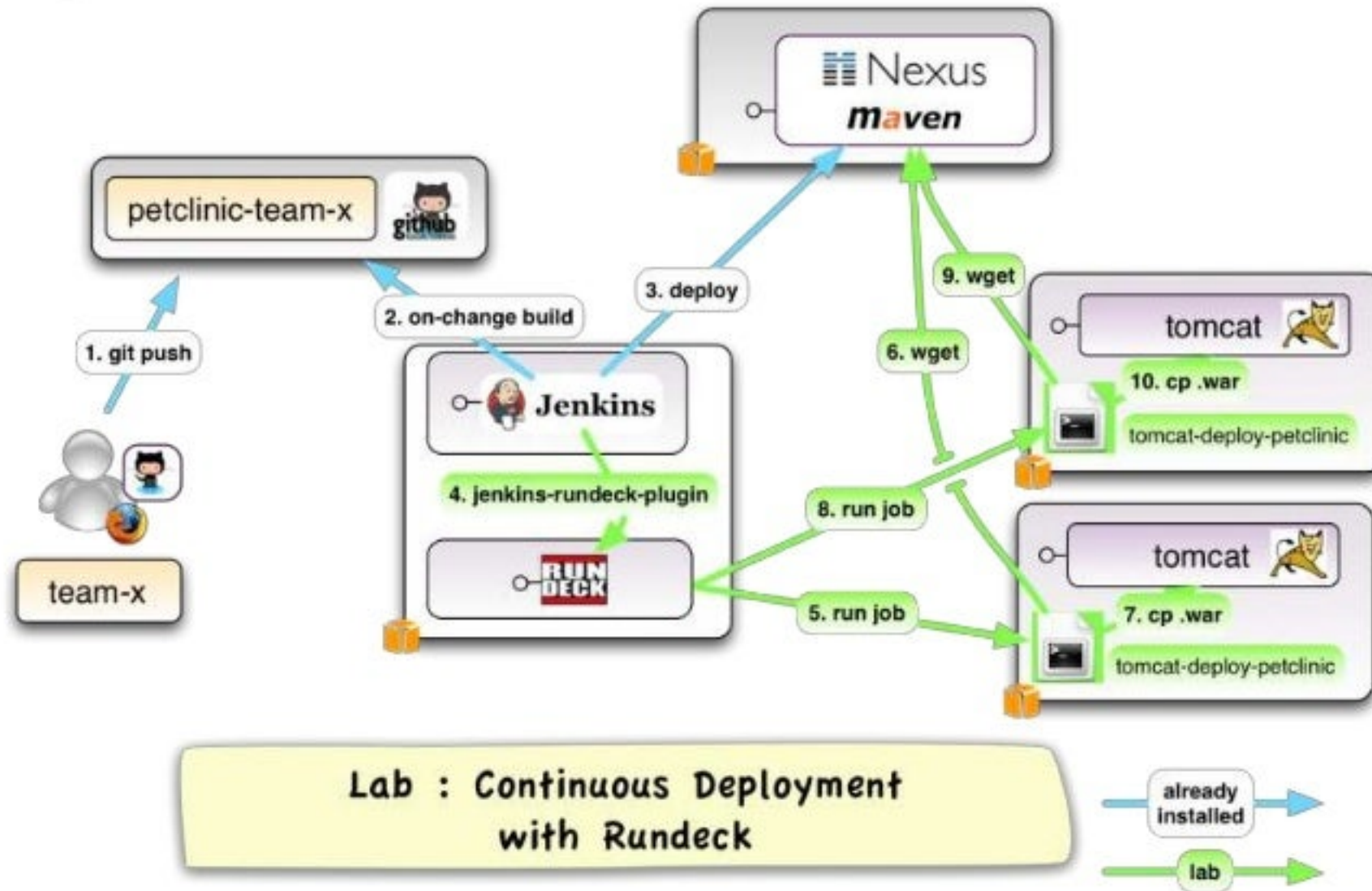
According to WS02



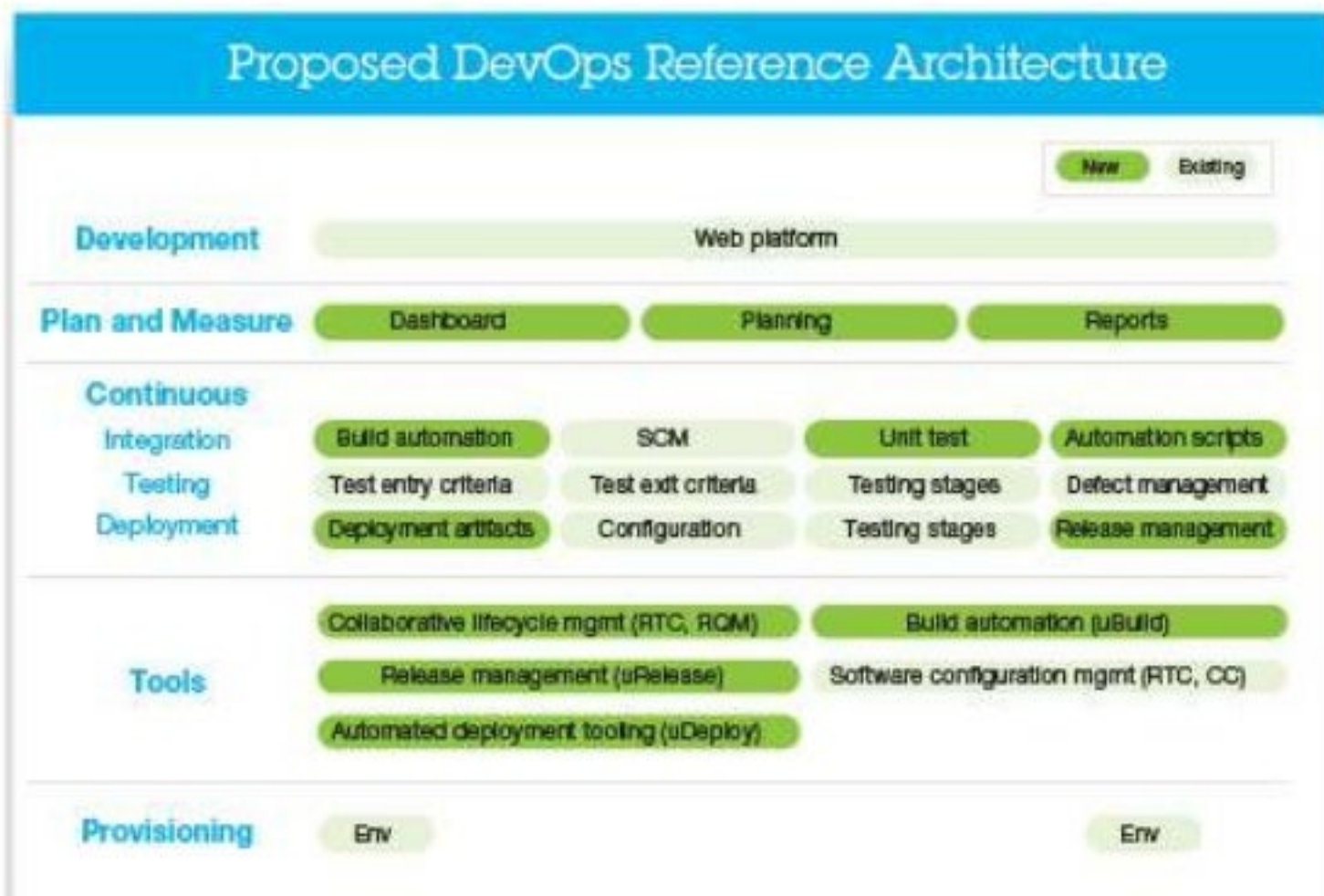
According to Momentum SI



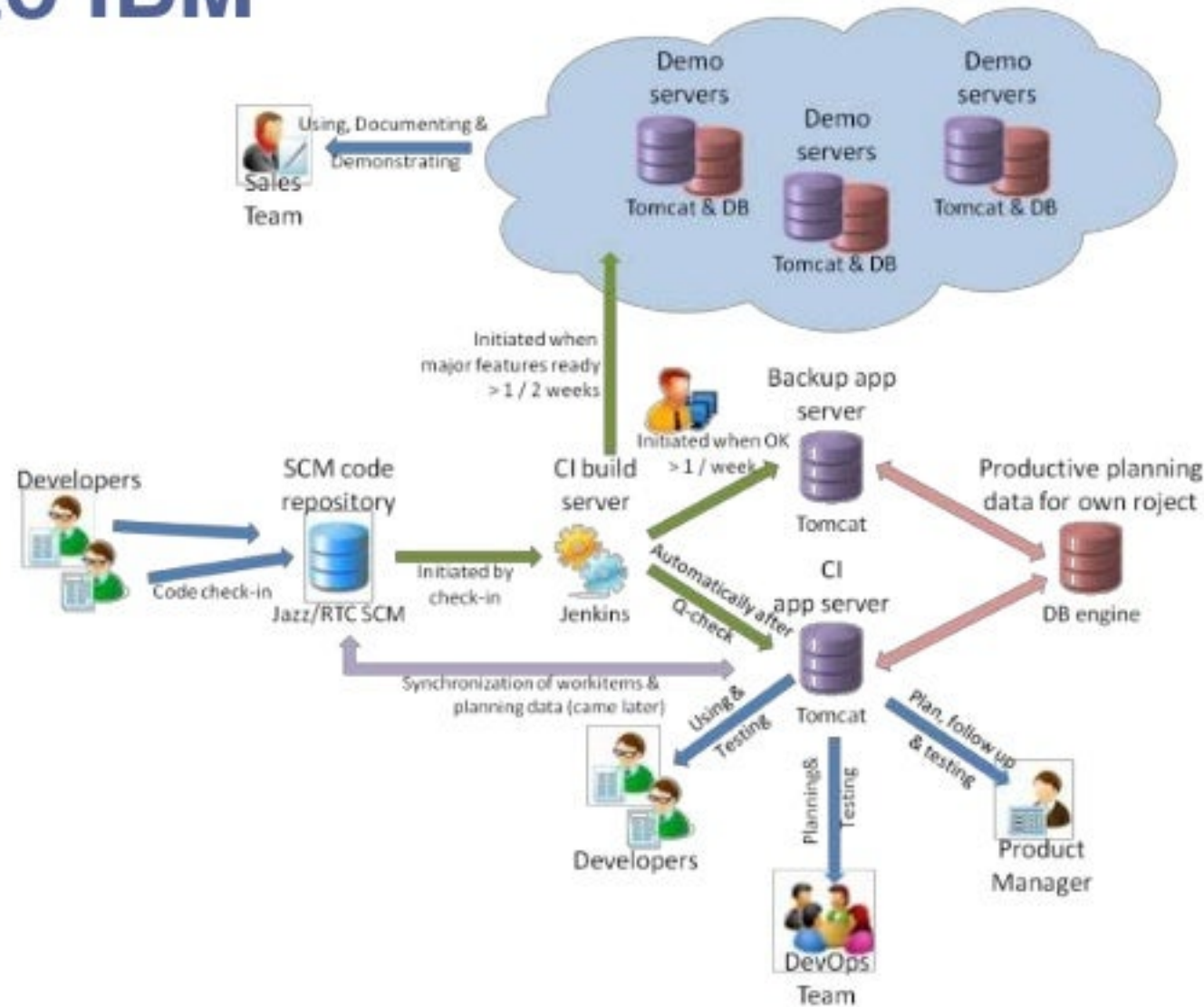
According to Goobbe



According to IBM



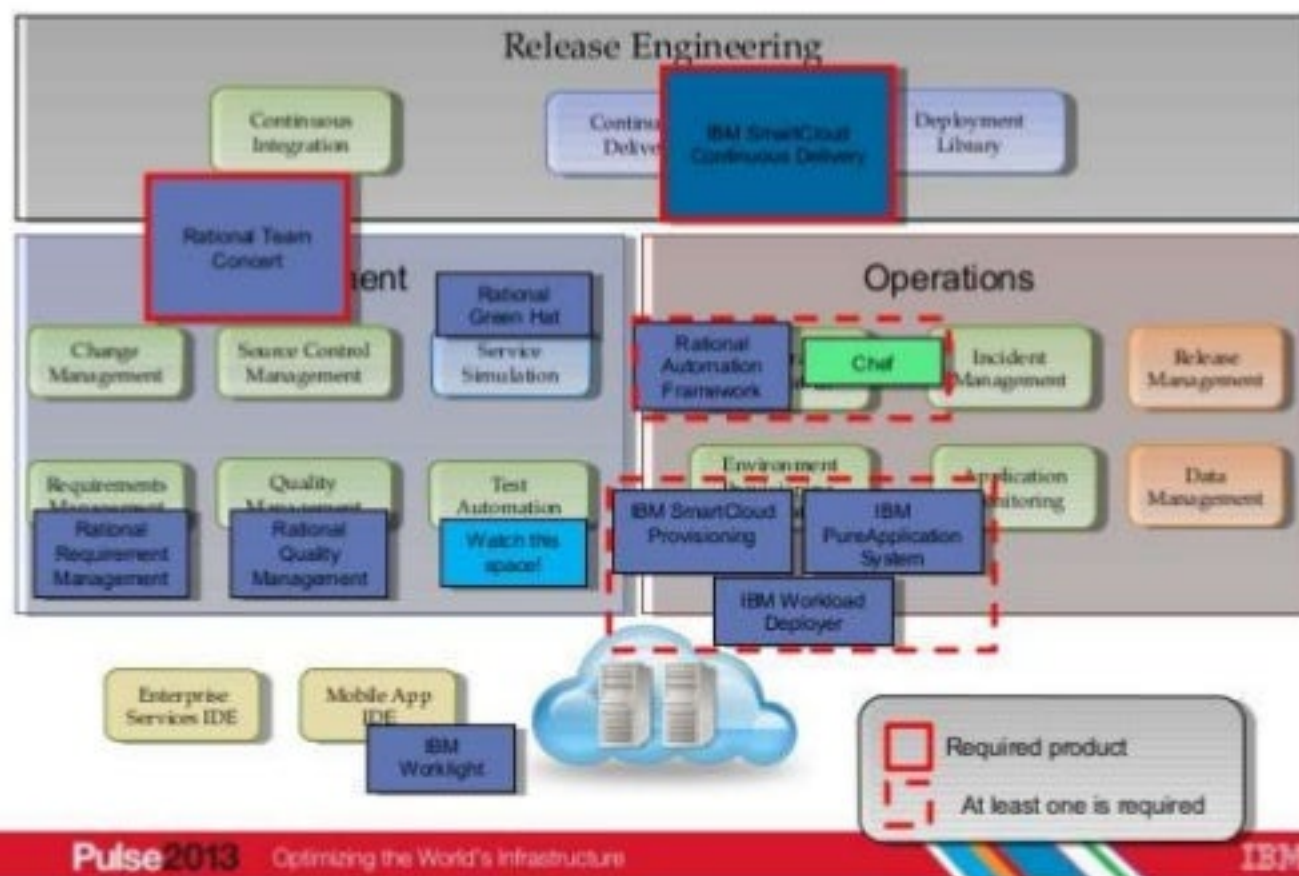
According to IBM



According to IBM



Reference Architecture: Product Implementations



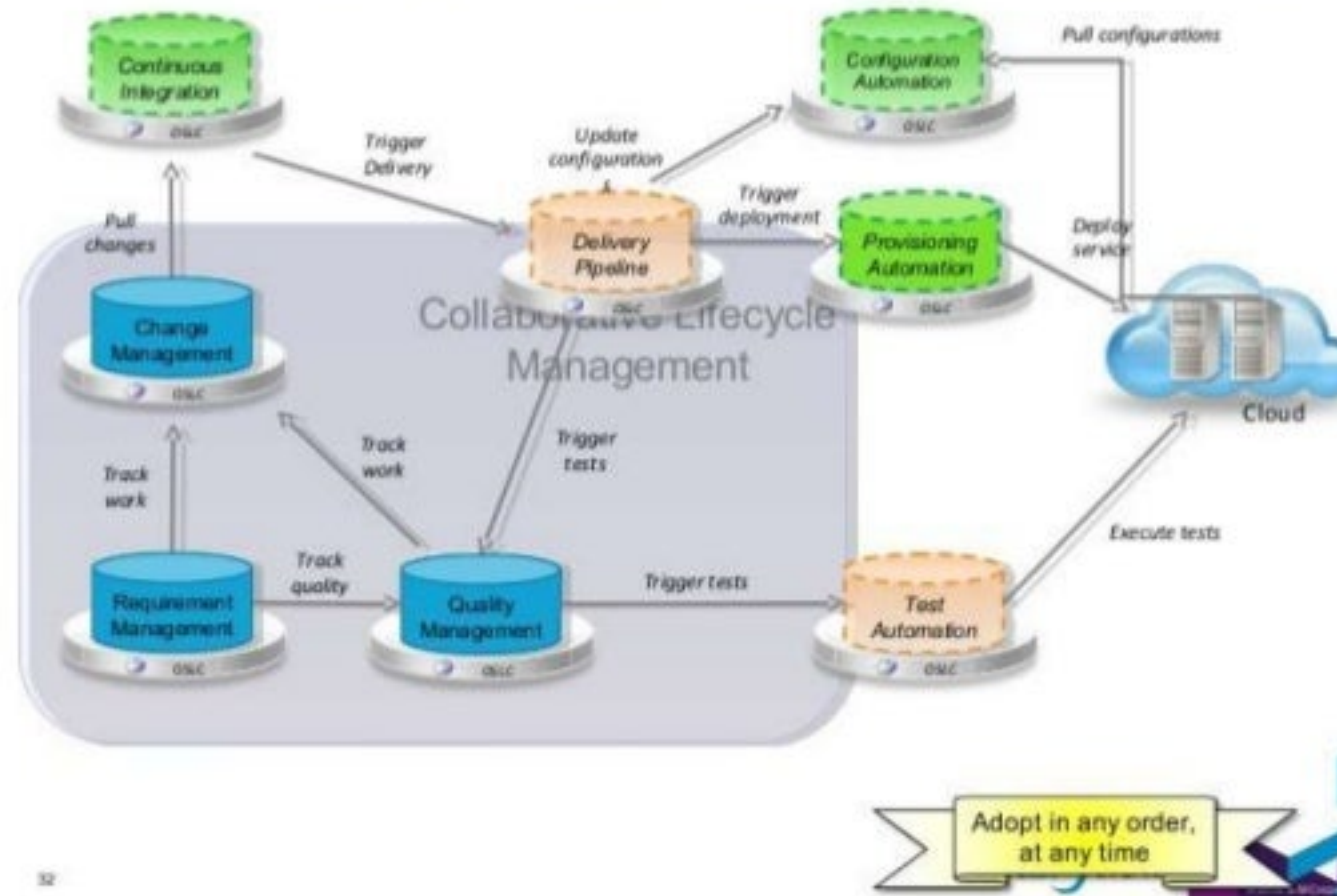
Pulse2013 Optimizing the World's Infrastructure

IBM

According to IBM

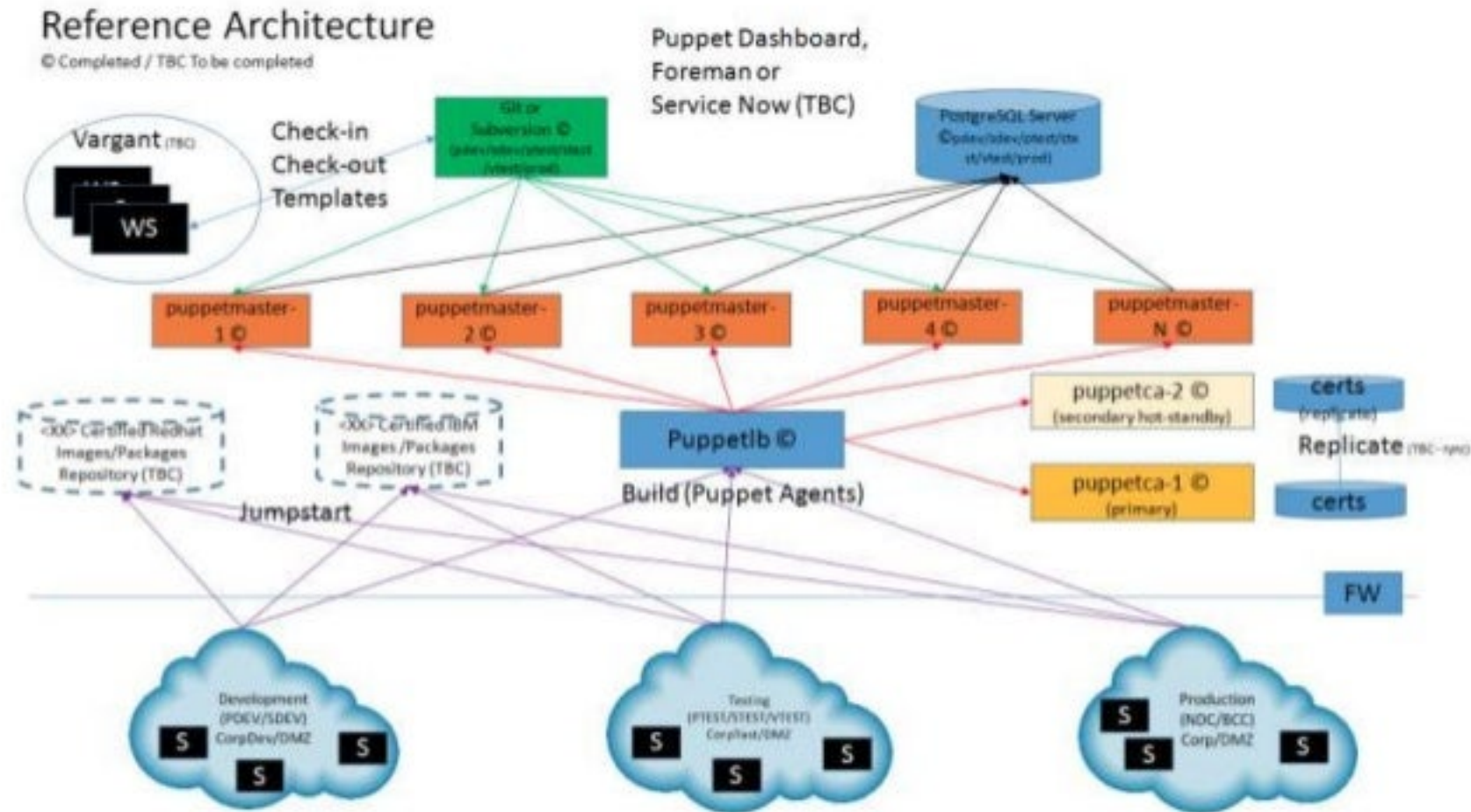


Lifecycle Management Reference Architecture

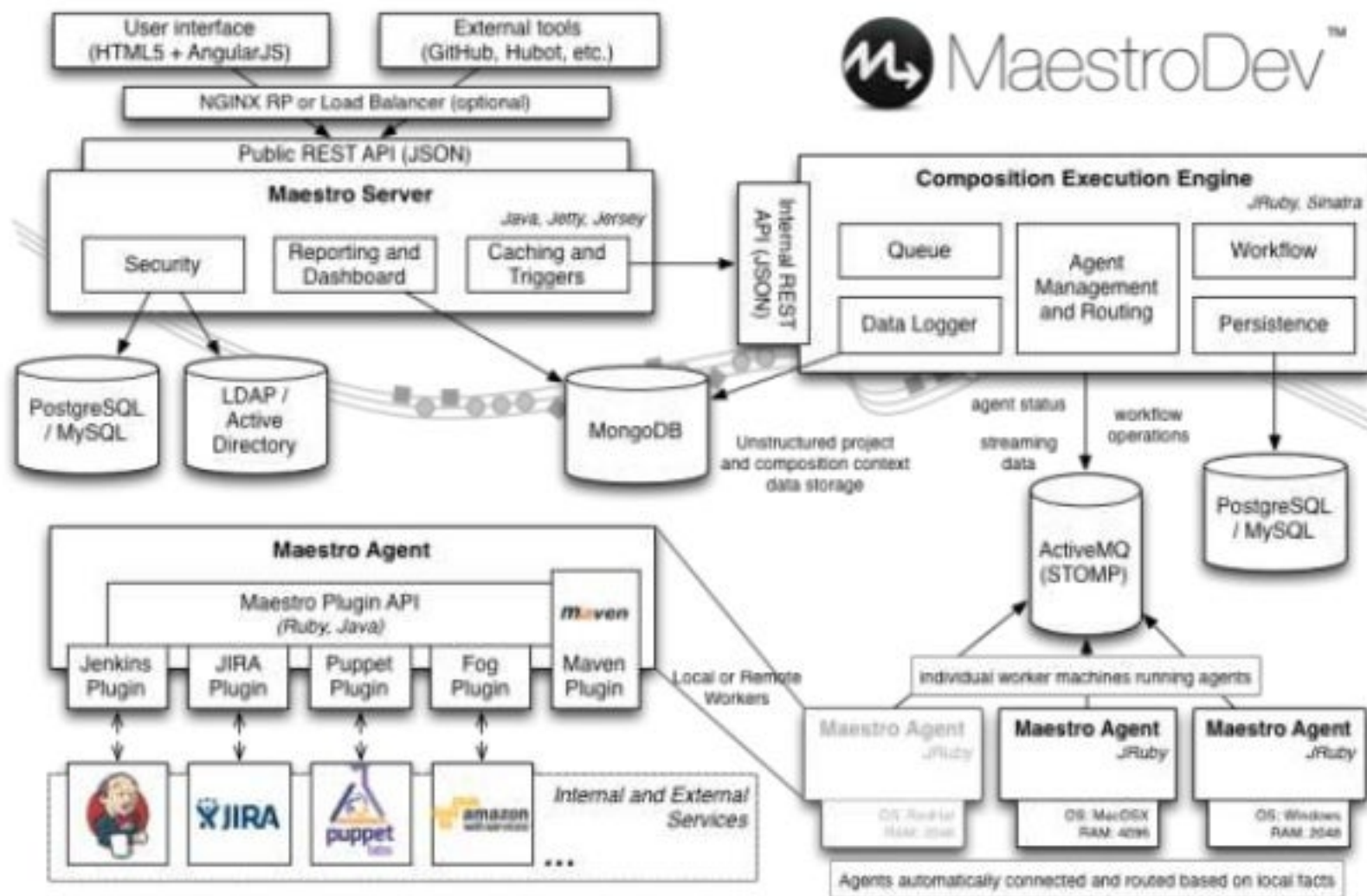


32

According to OZsofts

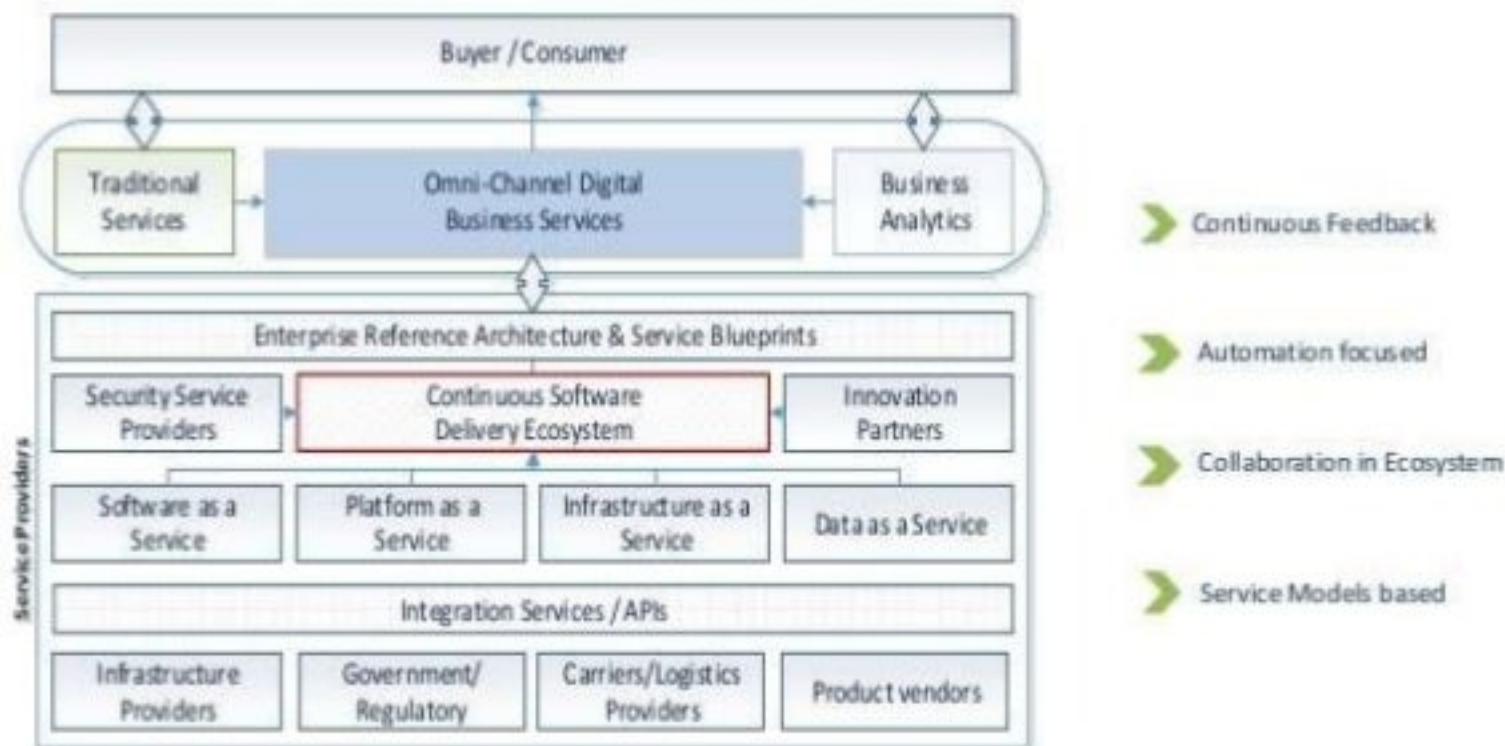


According to MaestroDev

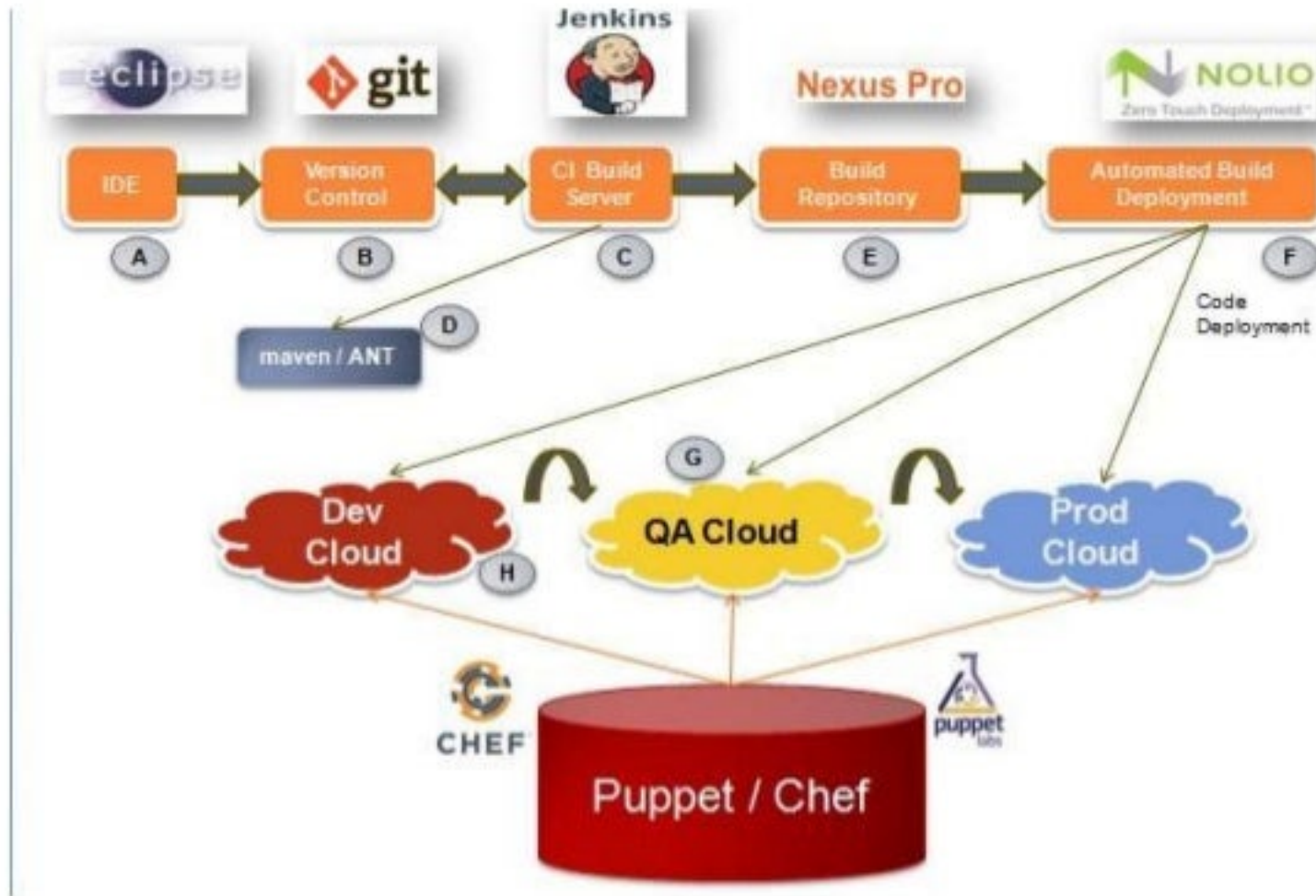


According to The Open Group

DevOps – A paradigm shift in IT to deliver Digitalization



According to AgileTrick

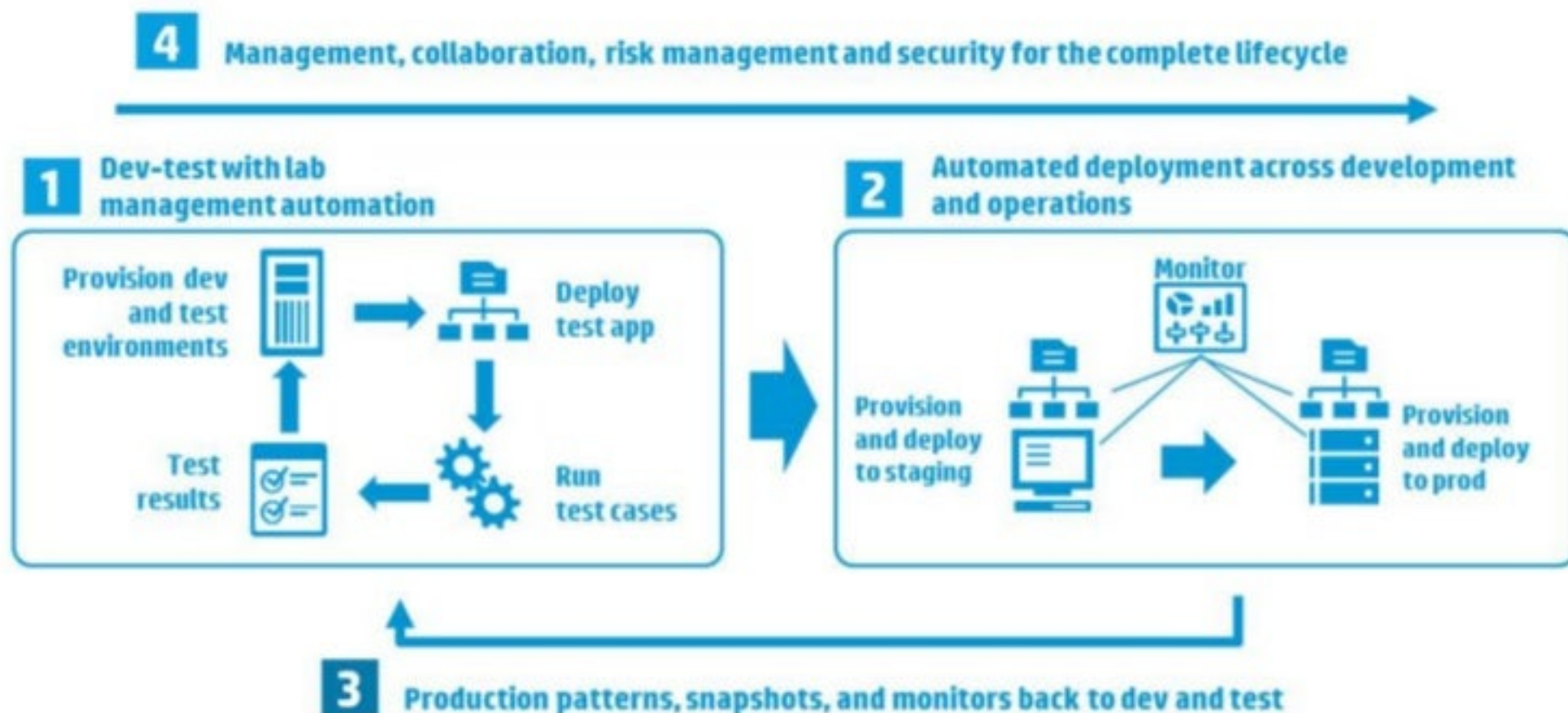


According to DevOps by Design

DEVOPS BY DESIGN

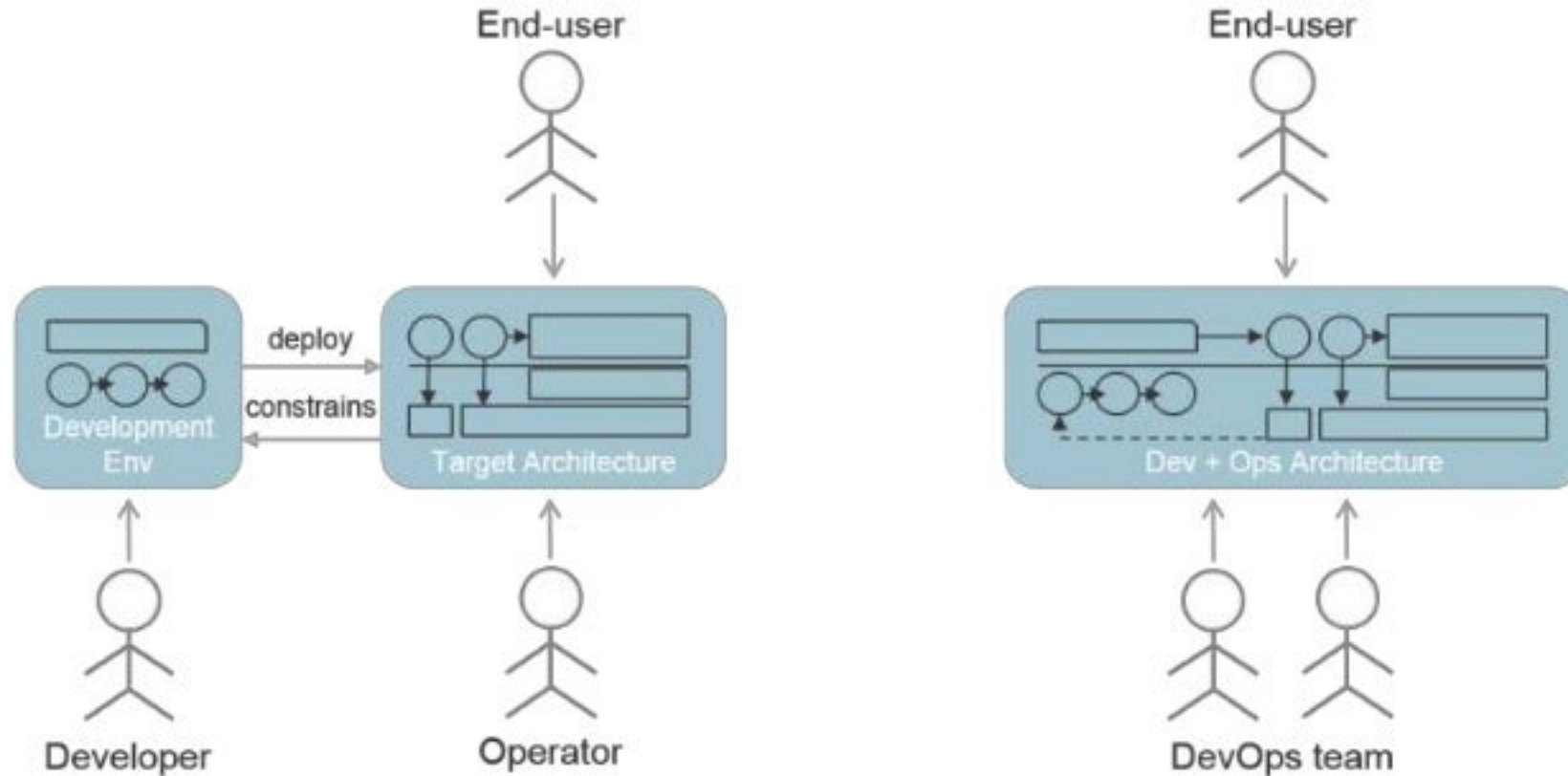


According to HP

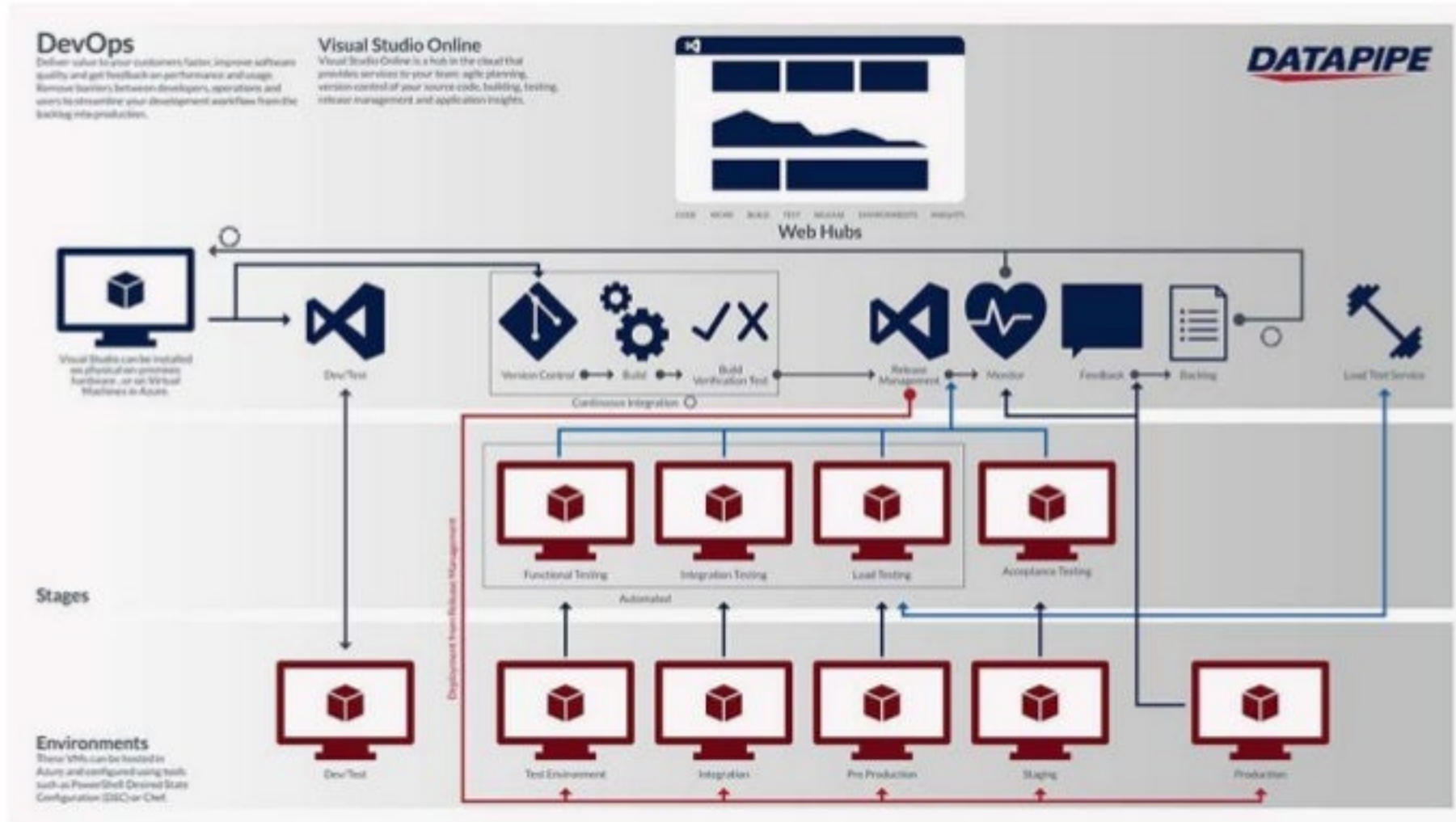


According to Eljto

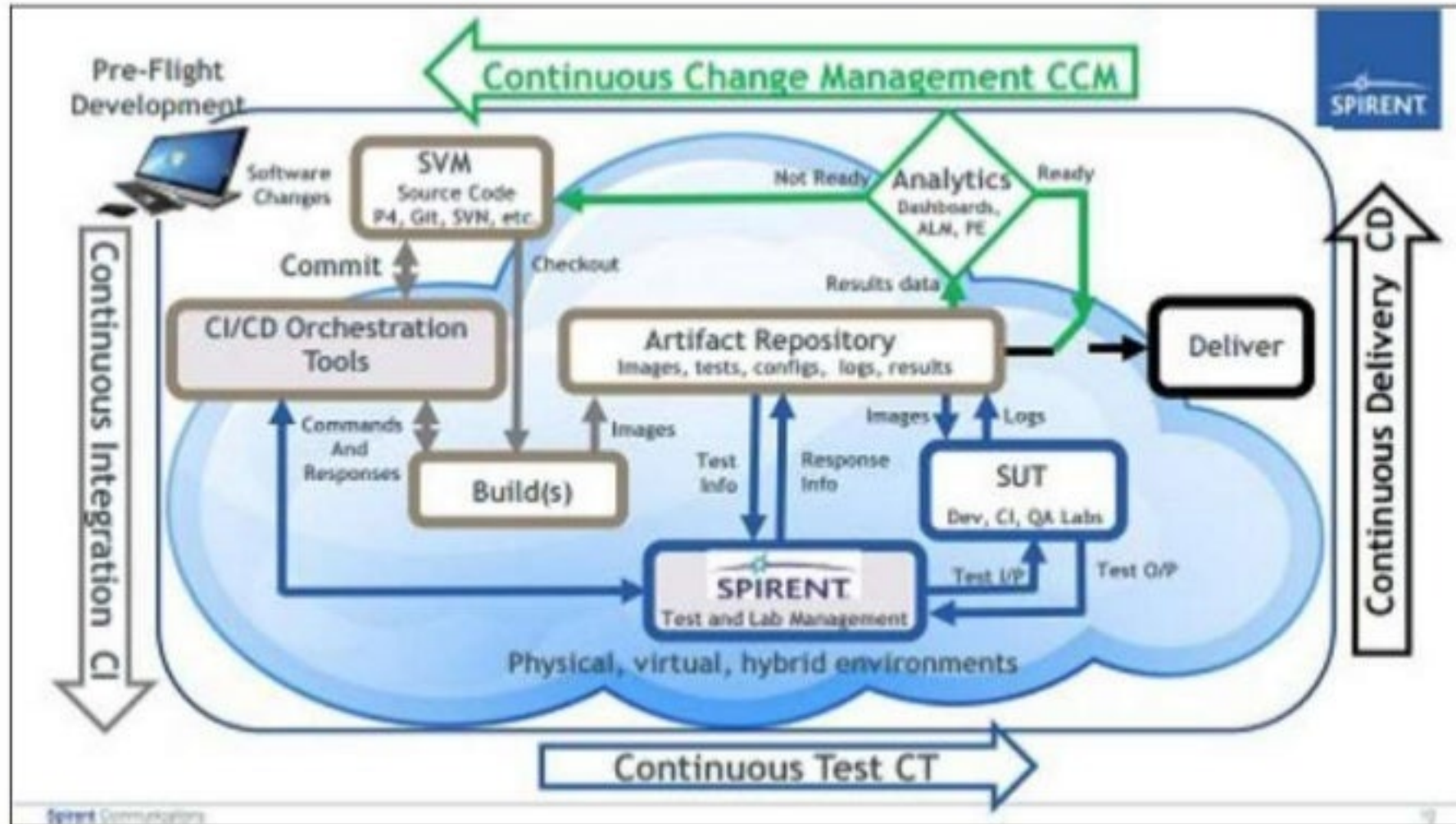
Architecture separated from Dev → Consolidated DevOps architecture



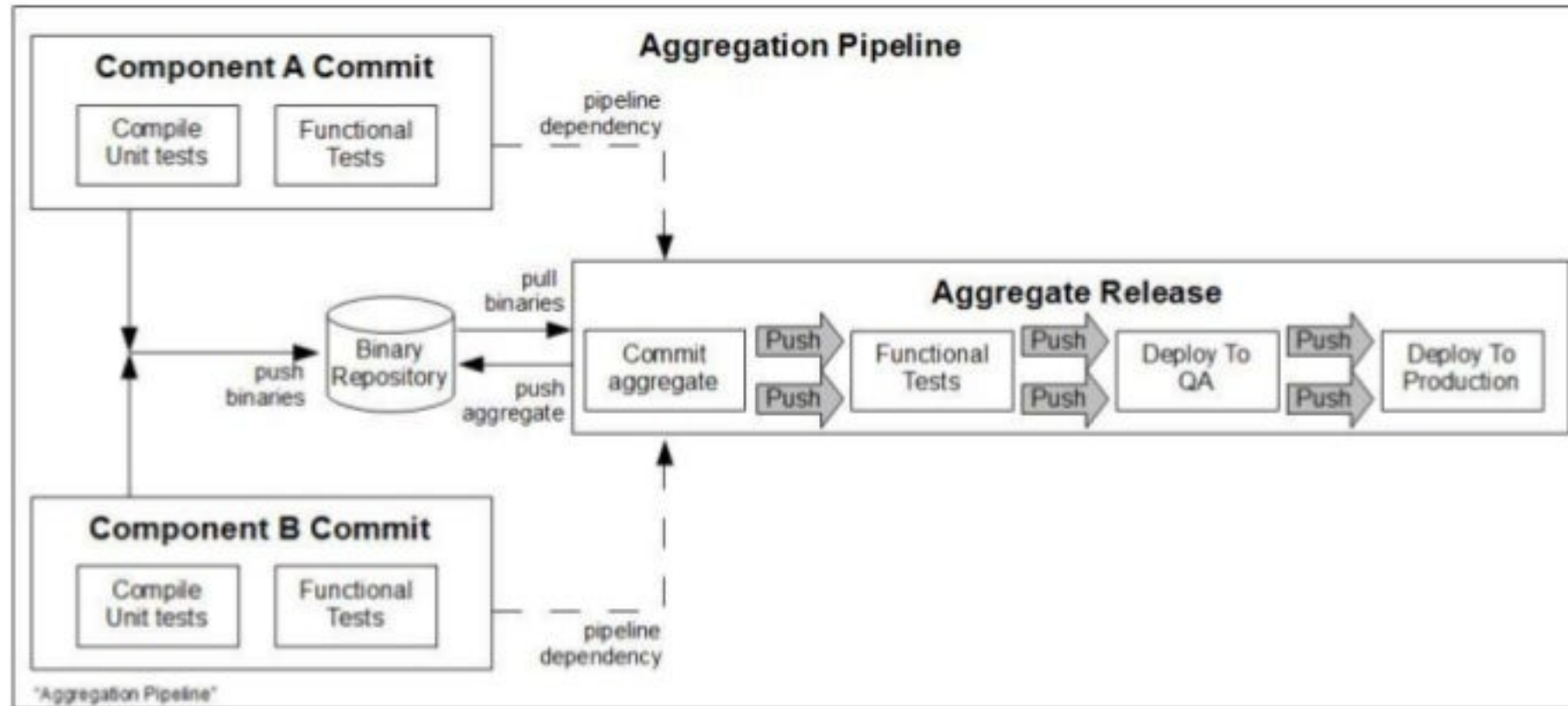
According to DATAPIPE



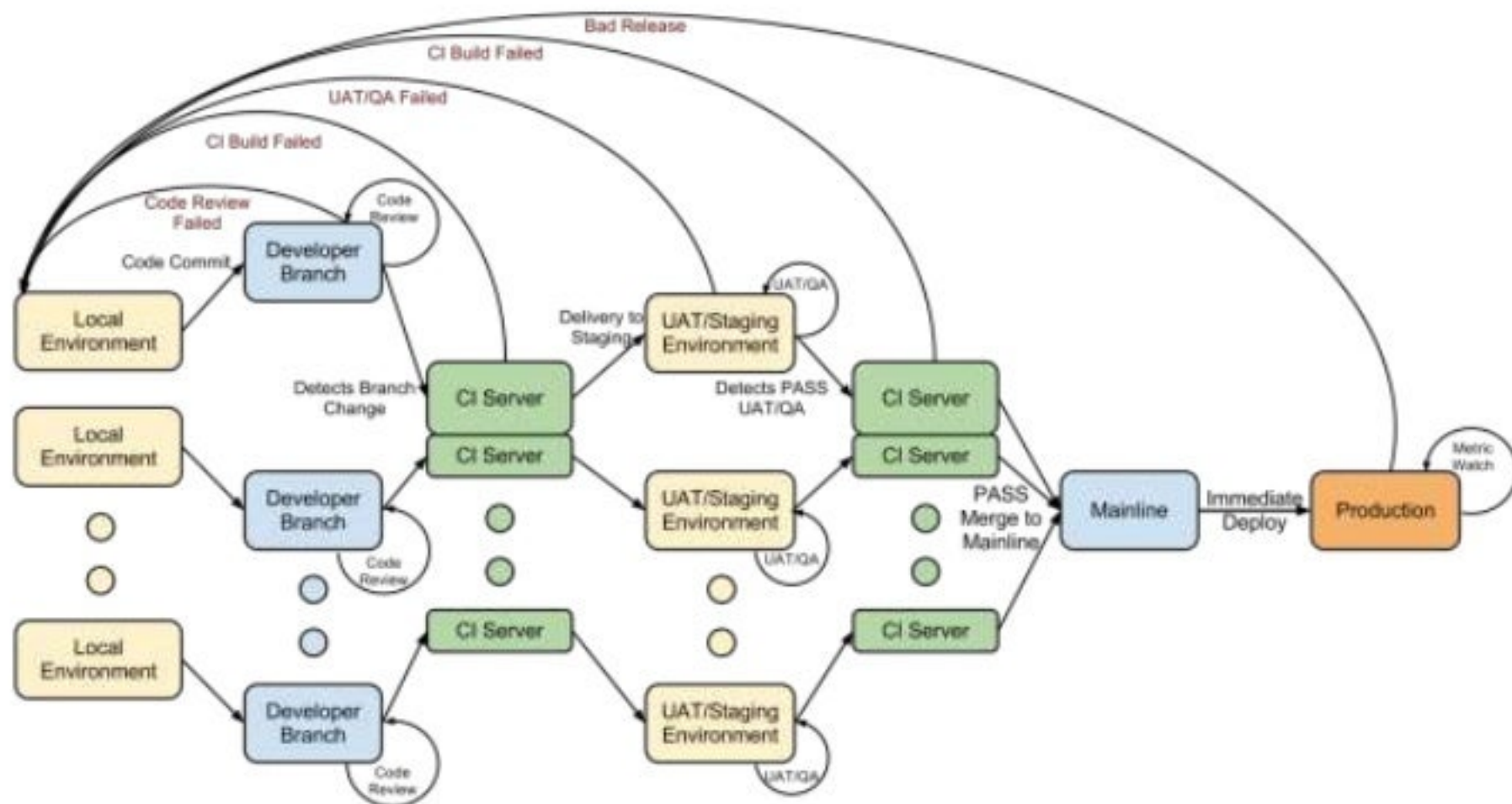
According to Spirent



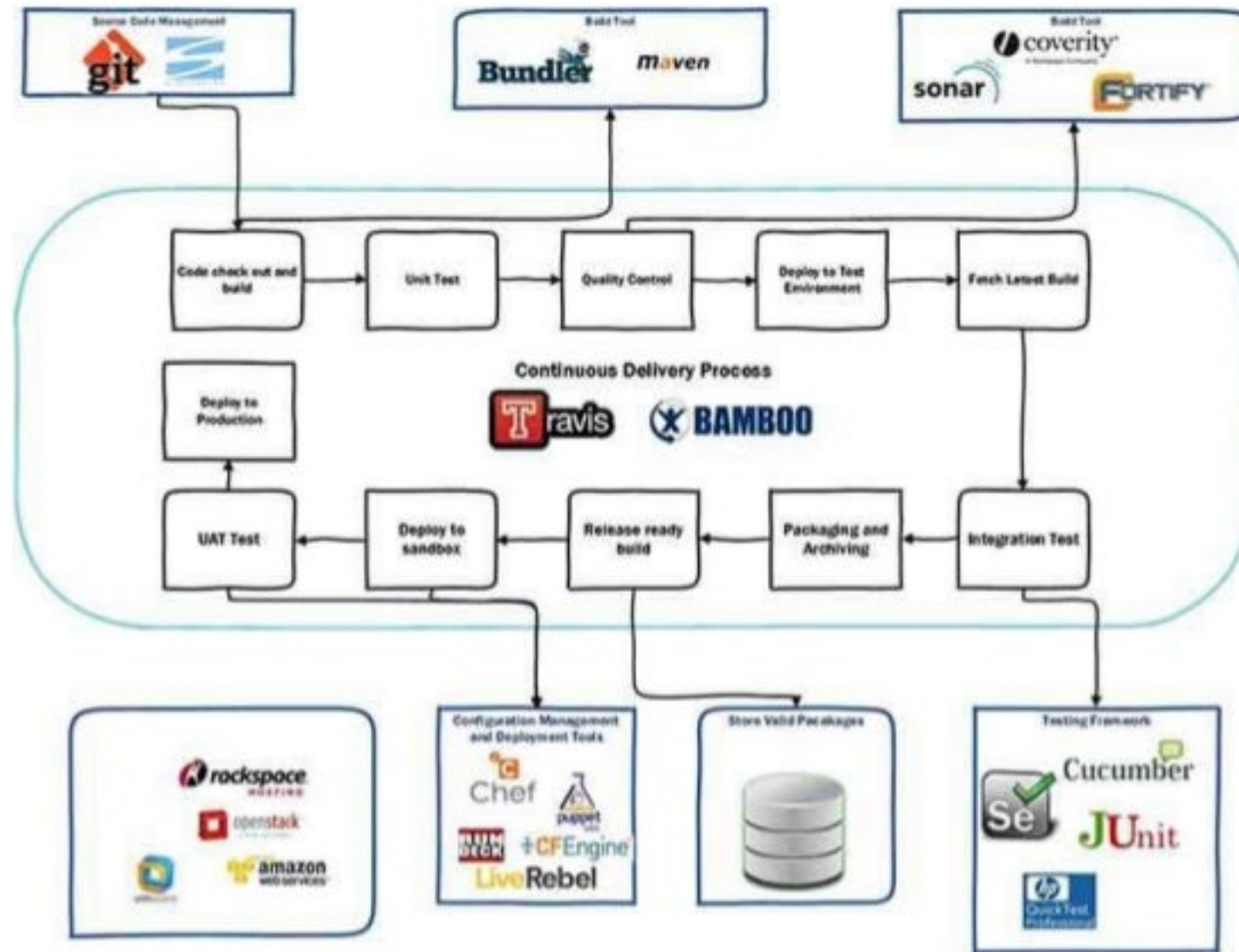
According to Always Agile



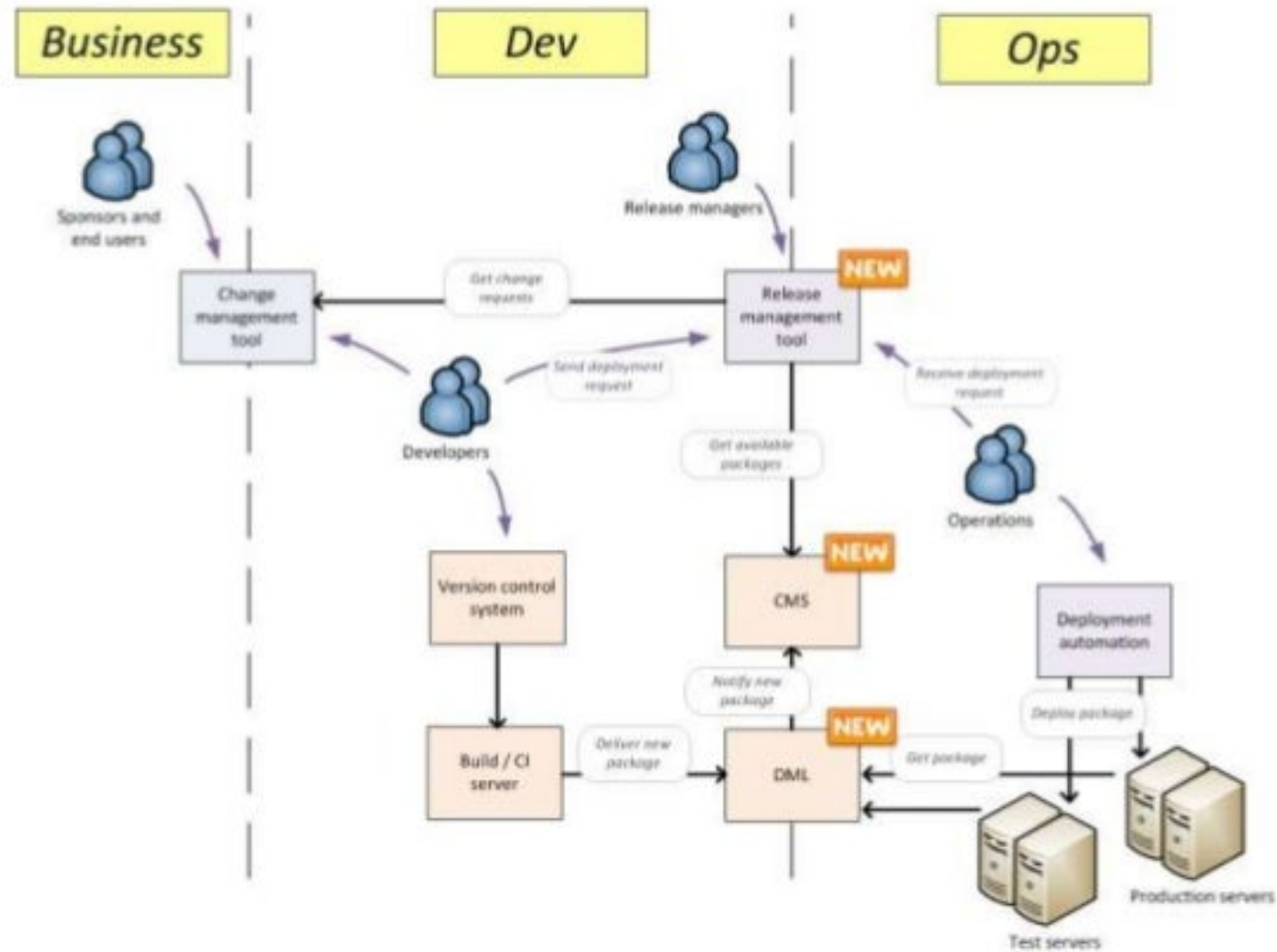
According to Assembla



According to Girikon



According to Bartholomeus





Andi Mann



Chris Swan



Damien Coraboeuf



Karthik Gokkavad



Madhu Akula



Damon Edwards



Derek Weeks



Ed Ruiz



Matt Jones



Simon Bennetts



Edward A Webb



Erlend Oftedal



Helen Beal



Zane Lackey



Shannon Lietz

96 practitioner led sessions. 24 hours.

October 24, 2017

Register for Free @ www.AllDayDevOps.com

