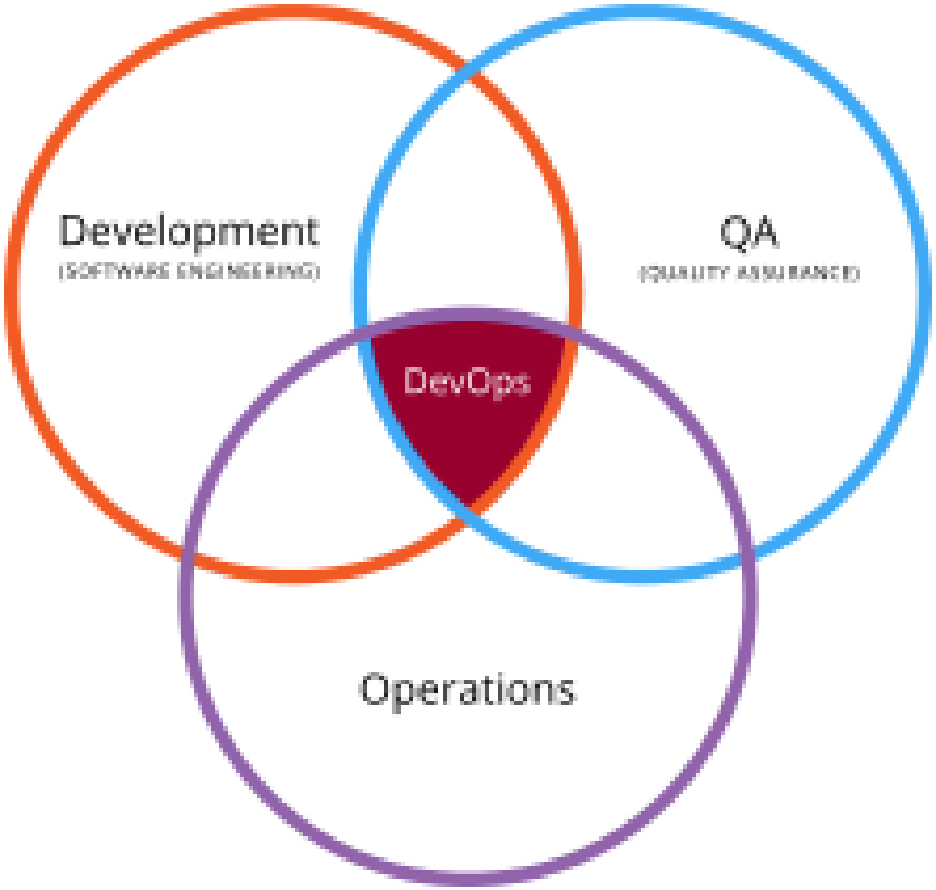


Automation Testing and the DevOps Pipeline
presented by Randy Spiess (Jan ‘18)

Learning Objectives

What is a DevOps Pipeline

DevOps Foundational Elements

Tools Used in the DevOps Pipeline

About Randy

DevOps Instructor

Agile Coach

Process & Quality Manager

Manager of Software Development

Manager of SCM and Test

What is DevOps

What Does DevOps Achieve

Highest Quality

Shortest Lead Time

Lowest Cost

Sustainability

State of DevOps 2017

High-performing organizations decisively outperform their lower-performing peers.

- They deploy 46 times more frequently.
- 440 times faster lead times
- recover 96 times faster
- have 5 times lower change failure rates.

High performers have better employee loyalty, as measured by employee Net Promoter Score (eNPS).

High-performing organizations

- spend 22 percent less time on unplanned work.
- are able to spend 29 percent more time on new work, such as new features or code.

DevOps is *not*...

A certification

A role

A set of tools

A prescriptive process

- Jez Humble

What is DevOps

“a cross-disciplinary community of practice dedicated to the study of building, evolving and operating rapidly-changing resilient systems at scale.” - Jez Humble

“A set of practices that emphasize the collaboration and communication of both software developers and information technology (IT) professionals while automating the process of software delivery and infrastructure changes. It aims at establishing a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably.” - Wikipedia

DevOps wants to achieve

Operations and development are skills, not roles. Delivery teams are composed of people with all the necessary skills.

Delivery teams run software products - not projects - that run from inception to retirement

How did DevOps get here

1948-75 Toyota Production System

1984 The Goal published

2001 Agile Manifesto

2007 Continuous Integration

2010 Continuous Delivery. First edition of DevOps newsletter

2013 The Phoenix Project. Docker initial release.

2014 Etsy doing 50 deploys/day, Amazon 1079 deploys per hour

2015 DevOps Enterprise Summit

Where is DevOps today?

DevOps Breakdown

DevOps Principles

Kaizen

Quality Culture

Automation

Sustainable

Plan/ Deliver Small

Customer Driven

Continuous Integration Key Components

Software and Infrastructure Automation: we are going to move to quickly to do things by hand in a risky manual way

Environment Management: Environments should be predictable, available and not get in the way

Application Lifecycle Management: We need to plan well and ensure traceability

Configuration Management: We need to control all of the moving parts

Tooling: We need technology

Collaboration: We can’t do this in isolated pockets, the scope is too wide

Quality comes not from monitoring, but from improvement of the production process.

It is not enough that management commit themselves to quality and productivity, they must know what it is they must do.

Such a responsibility cannot be delegated.

—W. Edwards Deming

Software House of Lean

THE GOAL: VALUE

Highest Quality, Lowest Cost, Shortest Lead Time

PILLAR 1: Respect for People

- Customer focus
- Equip your teams Build partnerships based on trust
- Develop people/teams
- Empowered teams
- Sustainability

DevOps Workflow

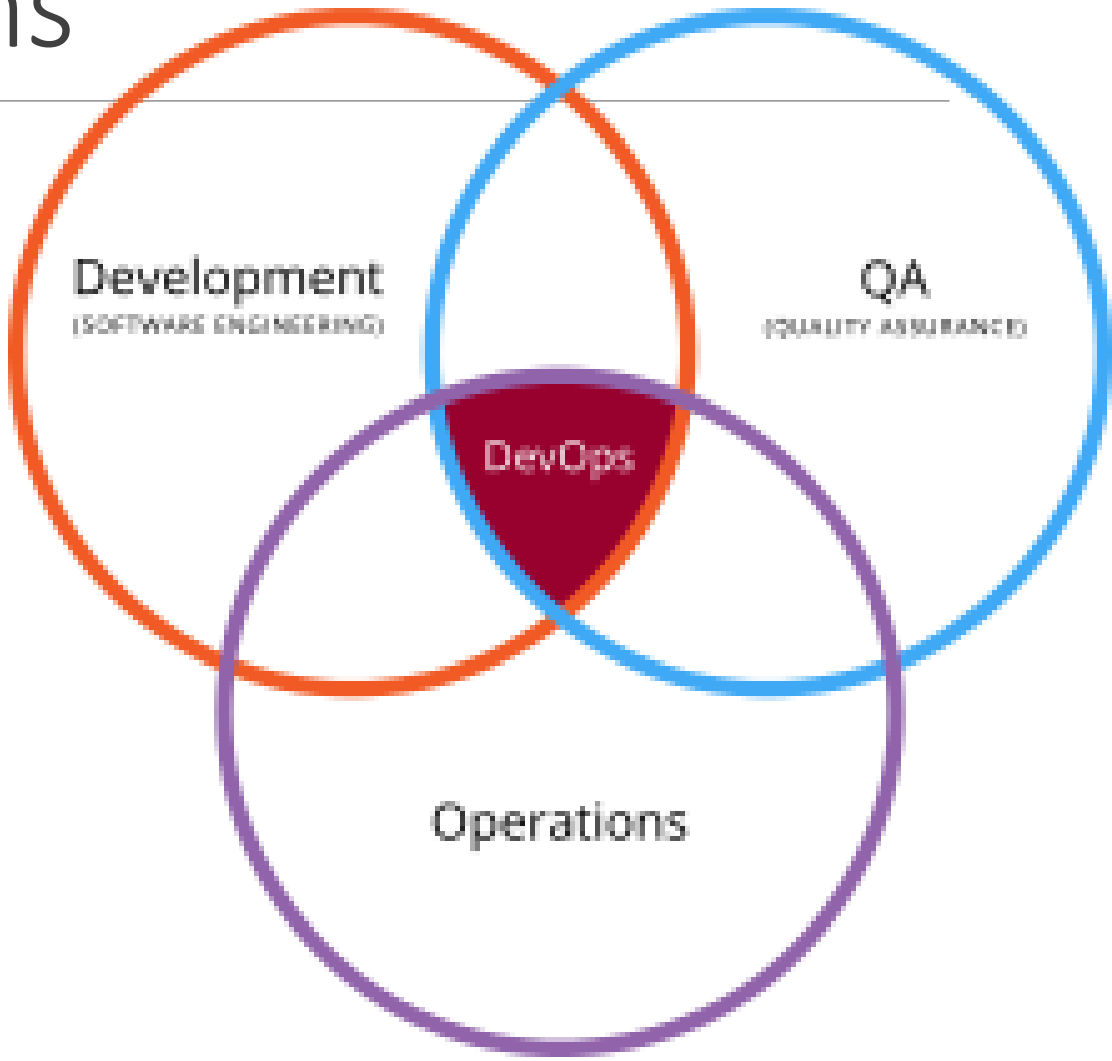
1. Actively manage queues
2. Reduce batch size
3. Apply fast feedback
4. Decentralize control

PILLAR 2: Kaizen

- Stop and notify of abnormalities
- Continuous improvement
- A constant sense of danger
- Use tools like retrospectives, root cause analysis

Configuration Management (everything), automated QA, DevOps orchestration, Low Inventory, Plan Small and Deliver Often

DevOps is a Balance of Dev, QA, Operations

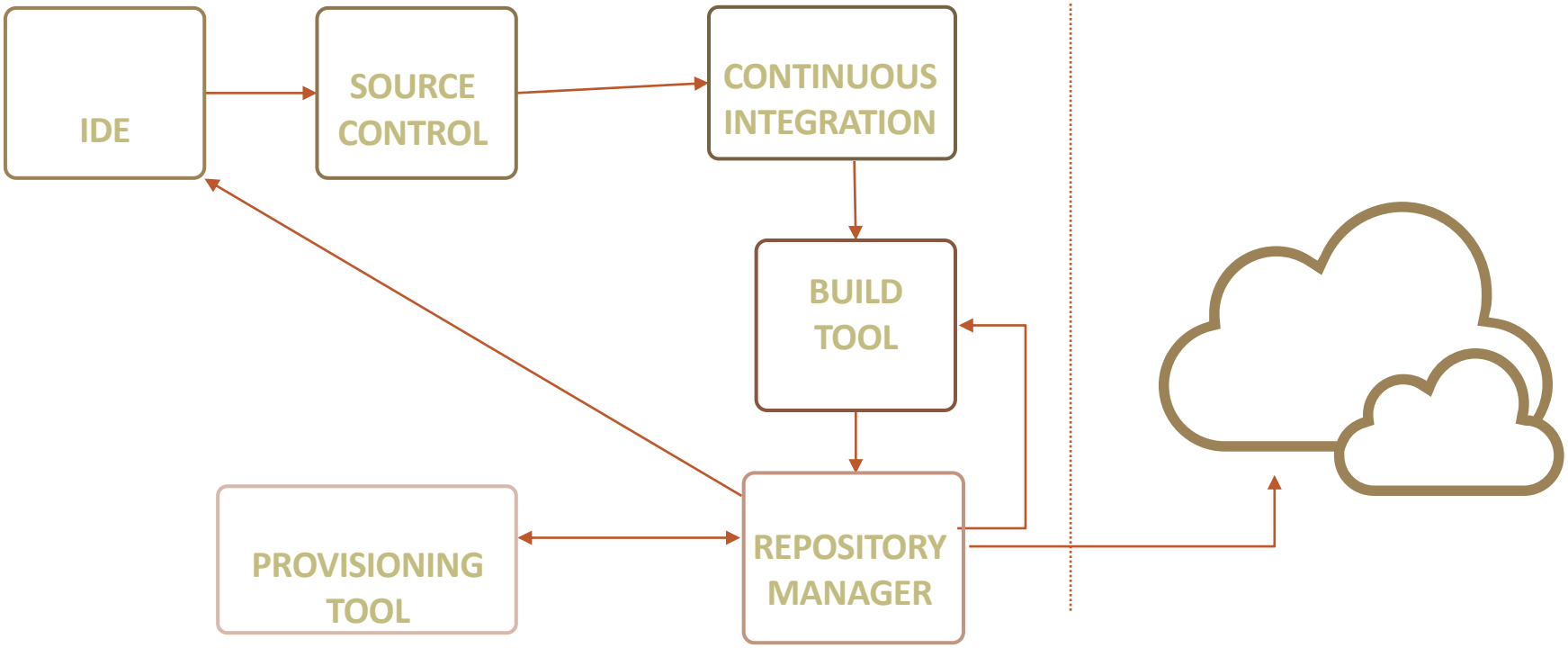


Agile and DevOps

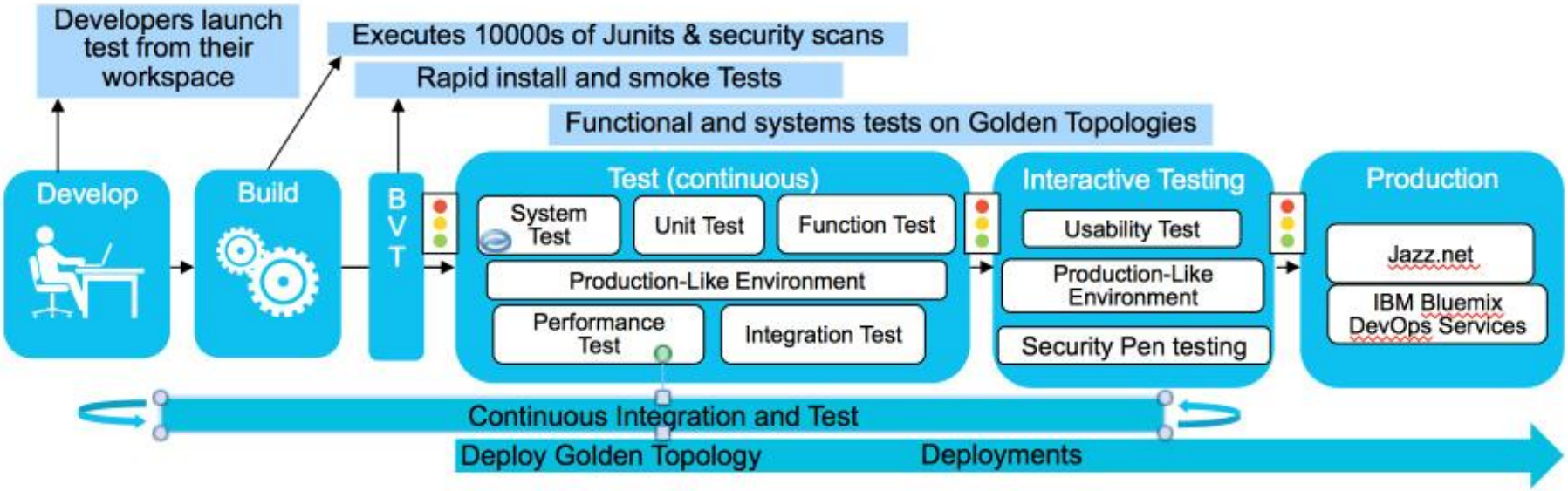
	Agile Manifesto Principle	Kaizen	Quality Culture	Automation	Sustainable	Plan/ Deliver Small	Customer Driven
1	Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.					x	x
2	Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.					x	x
3	Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.					x	
4	Business people and developers must work together daily throughout the project.	x	x				
5	Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.	x	x				
6	The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	x	x				
7	Working software is the primary measure of progress.	x	x	x	x	x	
8	Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.				x	x	
9	Continuous attention to technical excellence and good design enhances agility.	x					
10	Simplicity--the art of maximizing the amount of work not done--is essential.	x	x	x	x	x	x
11	The best architectures, requirements, and designs emerge from self-organizing teams.	x	x				
12	At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.	x	x		x		

Pipeline

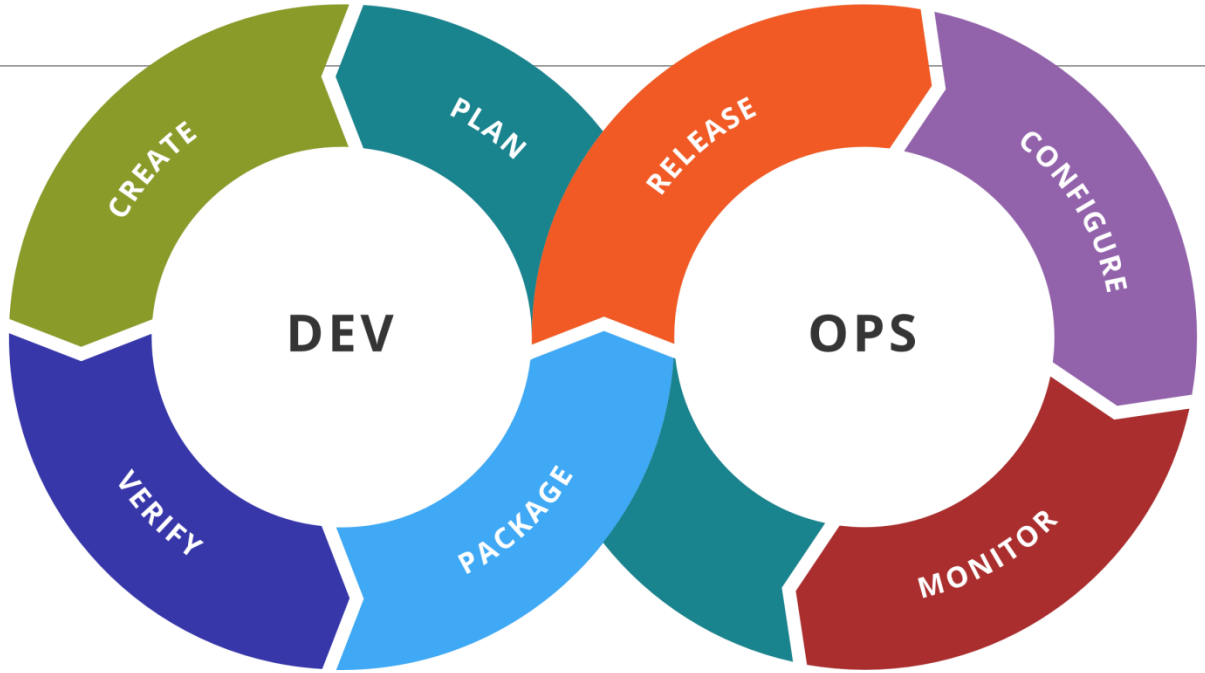
DevOps Flow/ Pipeline



DevOps Pipeline



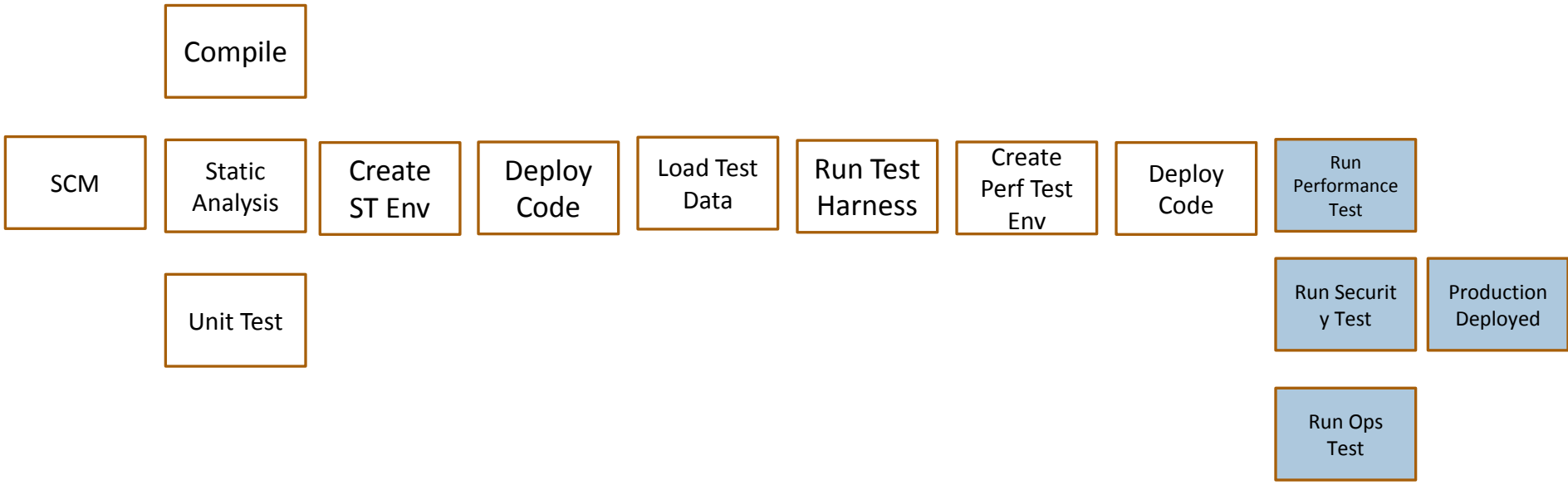
DevOps Toolchain



A **DevOps toolchain** is a set or combination of tools that aid in the delivery, development, and management of applications throughout the software development lifecycle, as coordinated by an organizations that uses DevOps practices.

Pipeline

Building a Delivery Pipeline is a highly effective way of maximizing the visibility and effectiveness of automation.



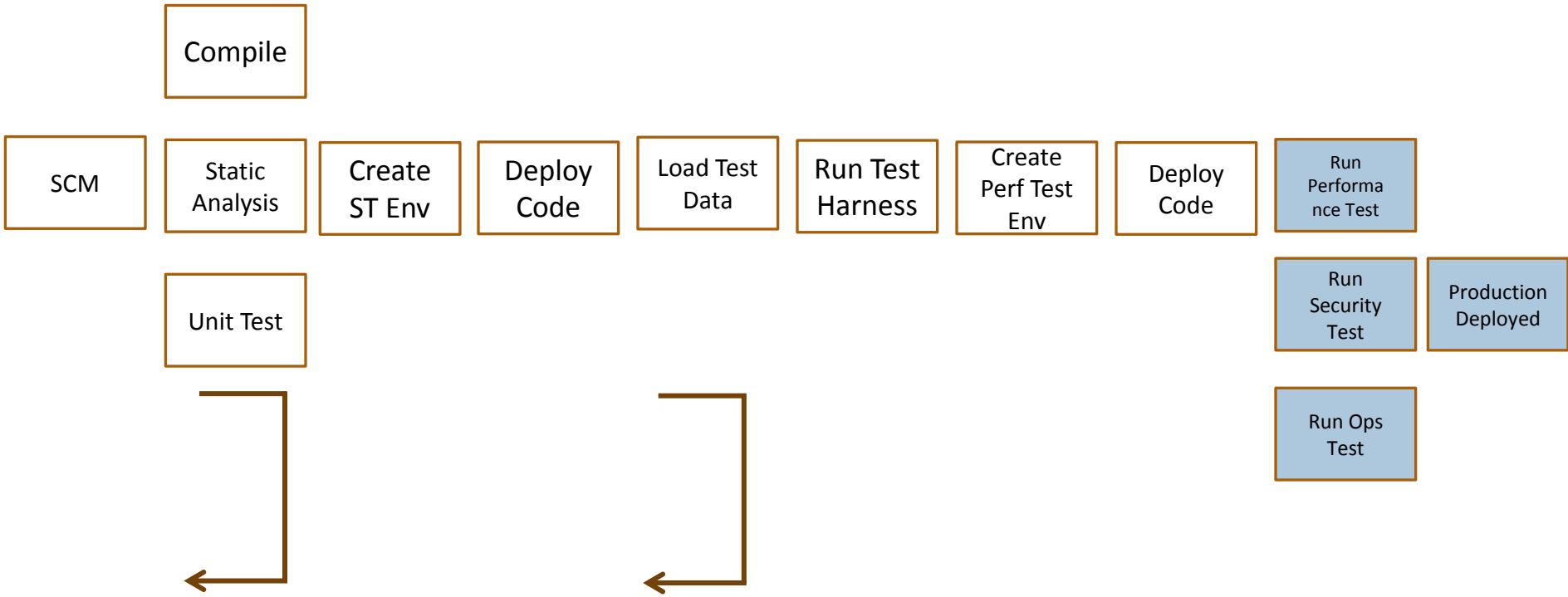
Continuous Integration

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly.

- Martin Fowler

Pipeline (CI)

Building a Delivery Pipeline is a highly effective way of maximizing the visibility and effectiveness of automation.



Continuous Delivery

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

- Agile Manifesto

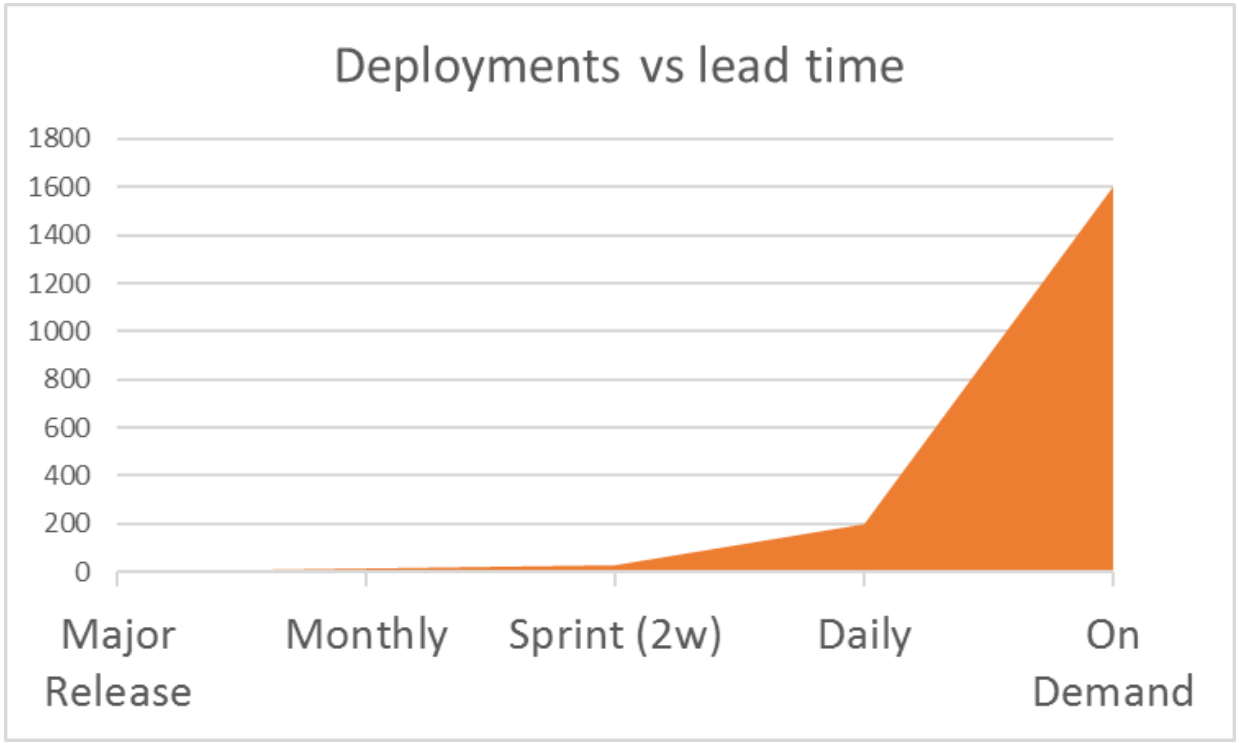
Reliable Software Releases through Build, Test and Deployment Automation.

- Jez Humble

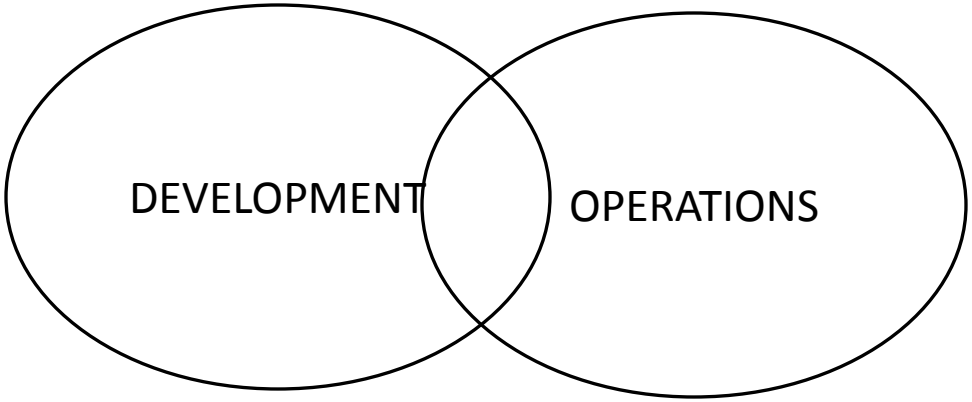
Shortening Lead Times

Doing things manually is no longer an option.

Predictability is essential.
No time for manual processes.



How Infrastructure as code fits into DevOps



Infrastructure is under CM

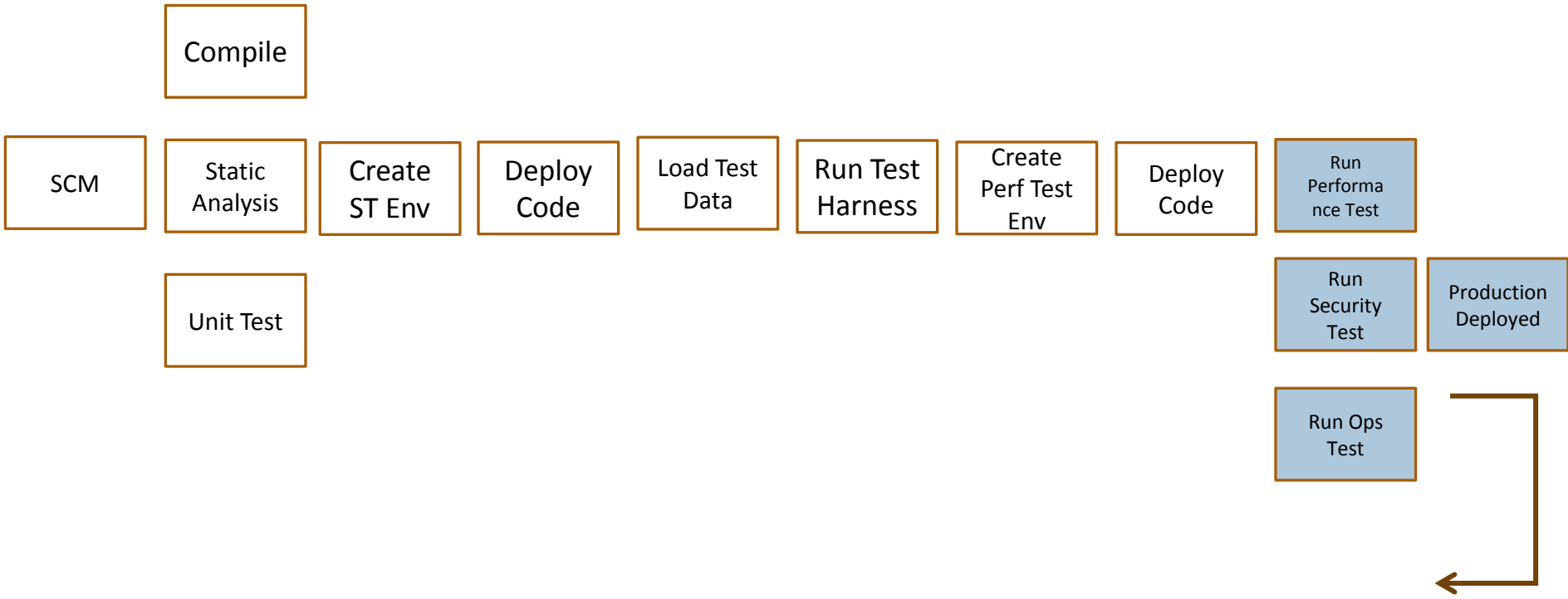
Bridges inconsistencies in Dev and Ops

Leverages automation and CM tools

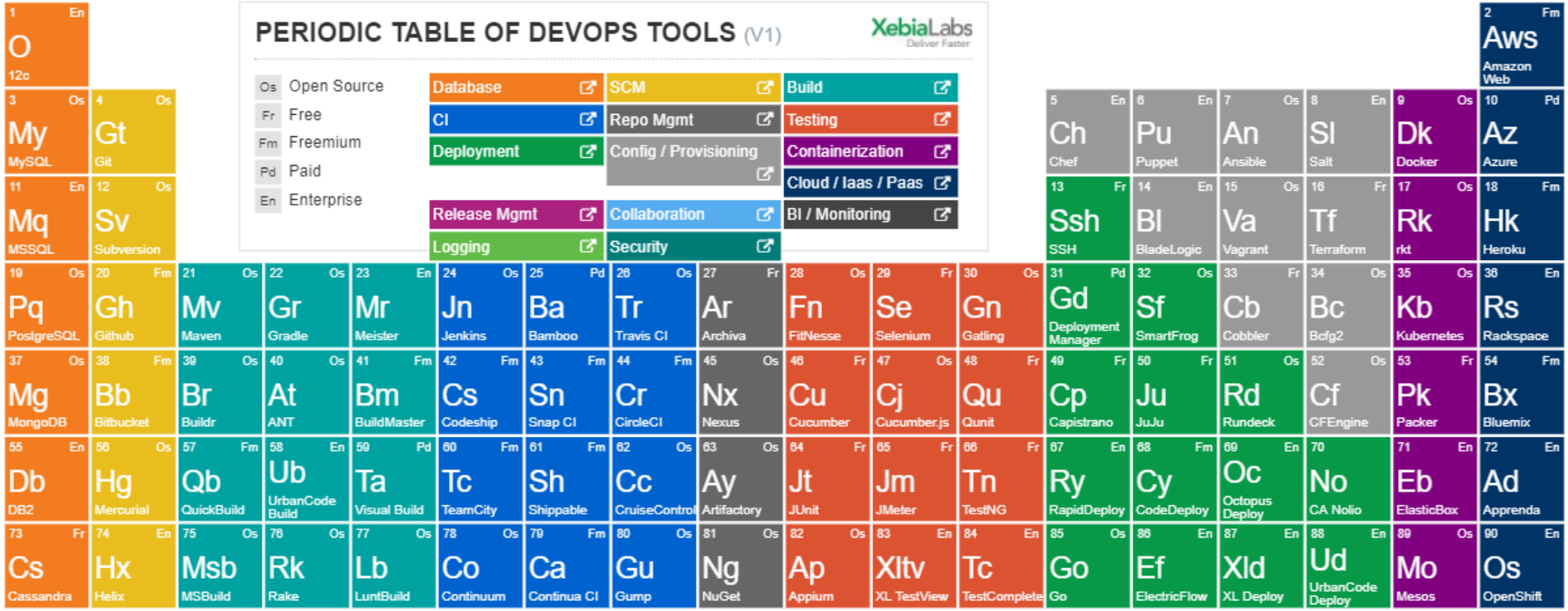
Facilitates the increased rate to demand for releases from application and business unit stakeholders

Pipeline (CI)

Building a Delivery Pipeline is a highly effective way of maximizing the visibility and effectiveness of automation.



DevOp Tools


PERIODIC TABLE OF DEVOPS TOOLS (V1)
XebiaLabs
Deliver Faster
Os Open Source
Fr Free
Fm Freemium
Pd Paid
En Enterprise
Database
SCM
Build
CI
Repo Mgmt
Testing
Deployment
Config / Provisioning
Containerization
Cloud / Iaas / Paas
Release Mgmt
Collaboration
BI / Monitoring
Logging
Security
1 En O 12c
2 Fm Aws Amazon Web
3 Os My MySQL
4 Os Gt Git
5 En Ch Chef
6 En Pu Puppet
7 Os An Ansible
8 En Sl Salt
9 Os Dk Docker
10 Pd Az Azure
11 En Mq MSSQL
12 Os Sv Subversion
13 Fr Ssh SSH
14 En Bl BladeLogic
15 Os Va Vagrant
16 Fr Tf Terraform
17 Os Rk rkt
18 Fm Hk Heroku
19 Os Pq PostgreSQL
20 Fm Gh Github
21 Os Mv Maven
22 Os Gr Gradle
23 En Mr Meister
24 Os Jn Jenkins
25 Pd Ba Bamboo
26 Os Tr Travis CI
27 Fr Ar Archiva
28 Os Fn FitNesse
29 Fr Se Selenium
30 Os Gn Gatling
31 Pd Gd Deployment Manager
32 Os Sf SmartFrog
33 Fr Cb Cobbler
34 Os Bc Bcfg2
35 Os Kb Kubernetes
36 En Rs Rackspace
37 Os Mg MongoDB
38 Fm Bb Bitbucket
39 Os Br Buildr
40 Os At ANT
41 Fm Bm BuildMaster
42 Fm Cs Codeship
43 Fm Sn Snap CI
44 Fm Cr CircleCI
45 Os Nx Nexus
46 Fr Cu Cucumber
47 Os Cj Cucumber.js
48 Fr Qu Qunit
49 Fr Cp Capistrano
50 Fr Ju JuJu
51 Os Rd Rundeck
52 Os Cf CFEngine
53 Fr Pk Packer
54 Fm Bx Bluemix
55 En Db DB2
56 Os Hg Mercurial
57 Fm Qb QuickBuild
58 En Ub UrbanCode Build
59 Pd Ta Visual Build
60 Fm Tc TeamCity
61 Fm Sh Shippable
62 Os Cc CruiseControl
63 Os Ay Artifactory
64 Fr Jt JUnit
65 Fr Jm JMeter
66 Fr Tn TestNG
67 En Ry RapidDeploy
68 Fm Cy CodeDeploy
69 En Oc Octopus Deploy
70 No CA Nolio
71 En Eb ElasticBox
72 En Ad Apprenda
73 Fr Cs Cassandra
74 En Hx Helix
75 Os Msb MSBuild
76 Os Rk Rake
77 Os Lb LuntBuild
78 Os Co Continuum
79 Fm Ca Continua CI
80 Os Gu Gump
81 Os Ng NuGet
82 Os Ap Appium
83 En Xltv XL TestView
84 En Tc TestComplete
85 Os Go Go
86 En Ef ElectricFlow
87 En Xld XL Deploy
88 En Ud UrbanCode Deploy
89 Os Mo Mesos
90 En Os OpenShift

DevOps Tools

SCM

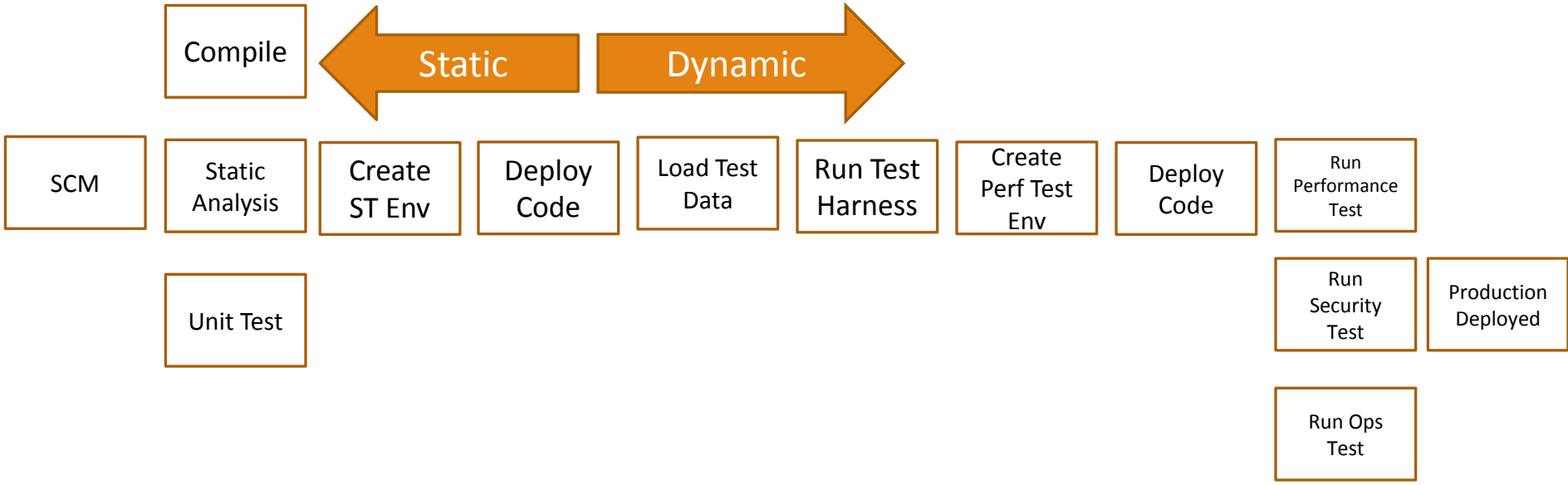
Build/ Release

QA/Test

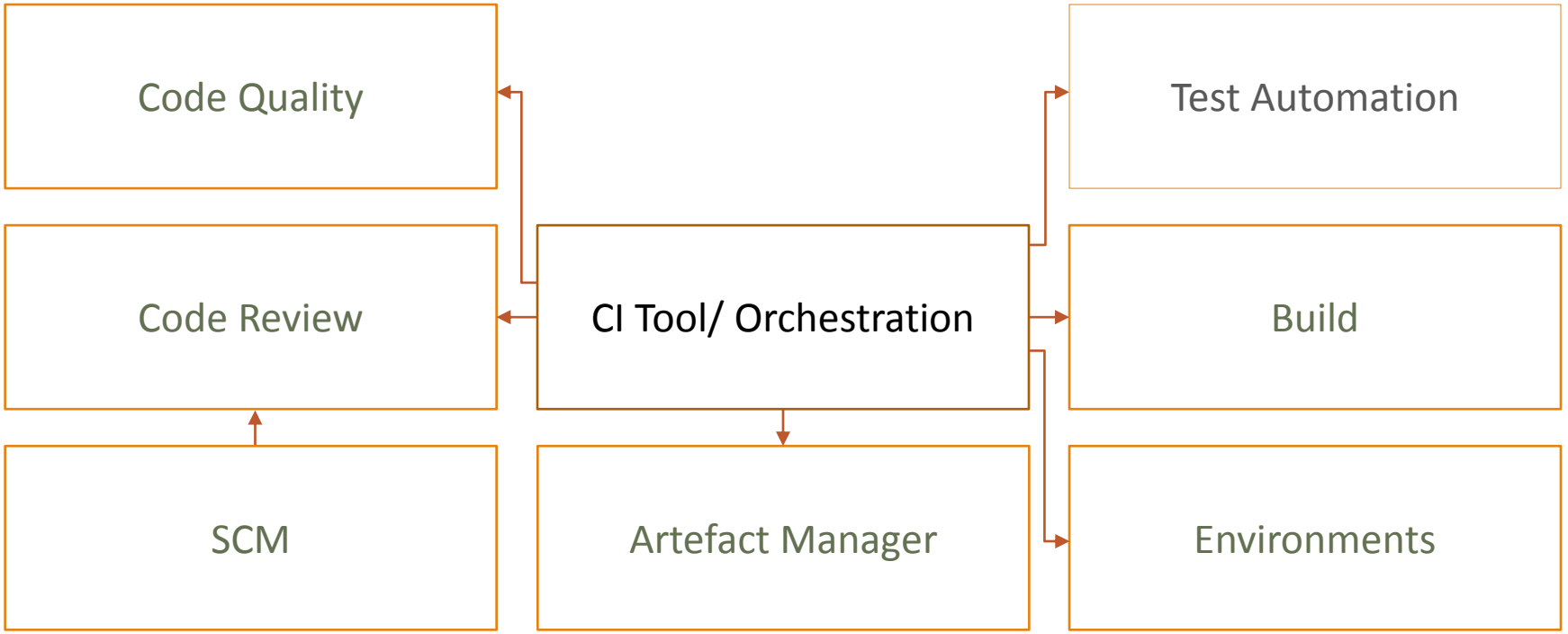
Infrastructure Configuration Managment

Pipeline

Building a Delivery Pipeline is a highly effective way of maximizing the visibility and effectiveness of automation.



Typical CI Architecture Practices



SCM Tools

SCM, a discipline comprising the tools and techniques to manage change to software assets

We need to ensure integration of code changes

We need to maintain revision history, what was update, who made updates

Ability to merge or undue changes

Git

Gerrit

subversion

Build and Release Tools

In DevOps, our software builds and deployments need to run more frequently and in less time

Continuous integration workflow

Continuous delivery enabling the ability to deploy to production environments and production like environments at any time

Jenkins

Bamboo

Automated Testing

QA/Testing Conciderations

Cost of Defects

It’s all about finding defects as early as possible

Verification skill investment

Verification tools investment

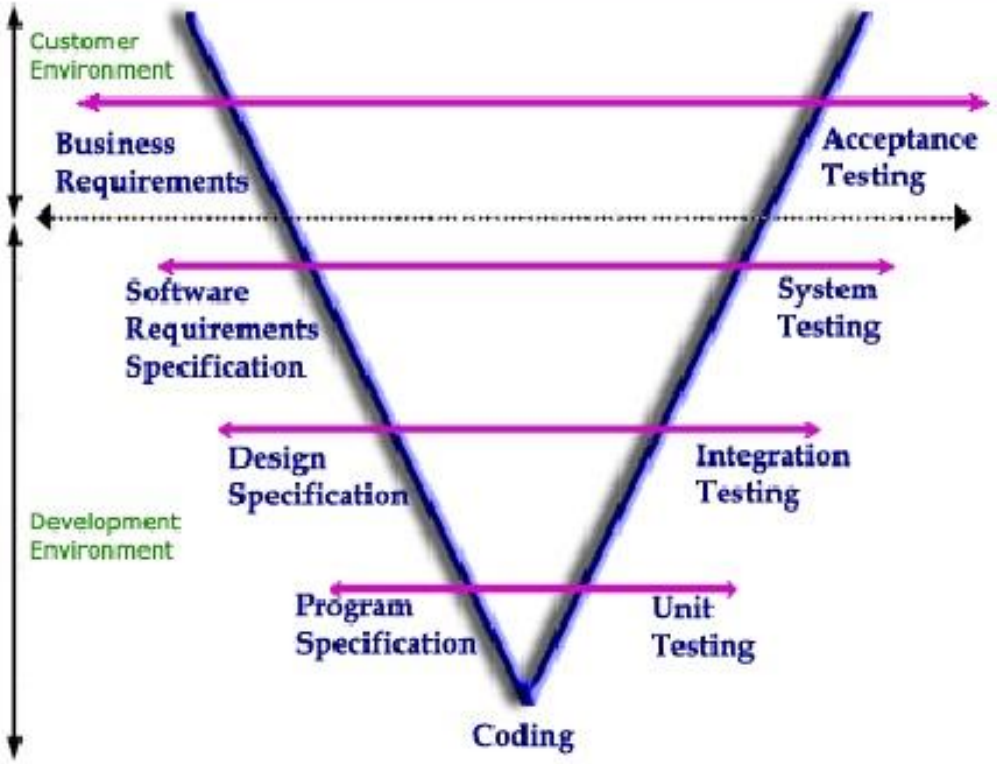
Working SW needs validation

- New functionality
- Regression (didn’t break anything)

Cost of Bugs in Release Cycle

Coding	Code Complete	Feature Complete (FC)	Release Candidate (RC)	General Availability (GA)
Cost = x	Cost = 5x	Cost = 10x	Cost = 50x	Cost = 1000x
	A bug in the code and code fix begin to impact other developers and other parts of the system.	A bug in the code and code fix now begin to impact theentire QA cycle. Each day lost because of the bug starts to push the entire schedule.	A bug in the code and code fix clearly jeopardizes the GA date. Running out of time to execute tests neccessary to ensure the integ-nity of the product after the code fix.	A bug in the code causes customer production down. Customer suffers monetary damages. Fixing code means incurring the cost of a patch release.

V-model



Orthogonal Defect Classification (ODC)

Lean way to perform root cause analysis.

IBM Research, Ram Chillarege published in 1992

ODC Trigger is the most likely place to capture an error/defect.

ODC Triggers are used to evaluate the effectiveness of testing (recent)

ODC Activity to Trigger

PEER REVIEW

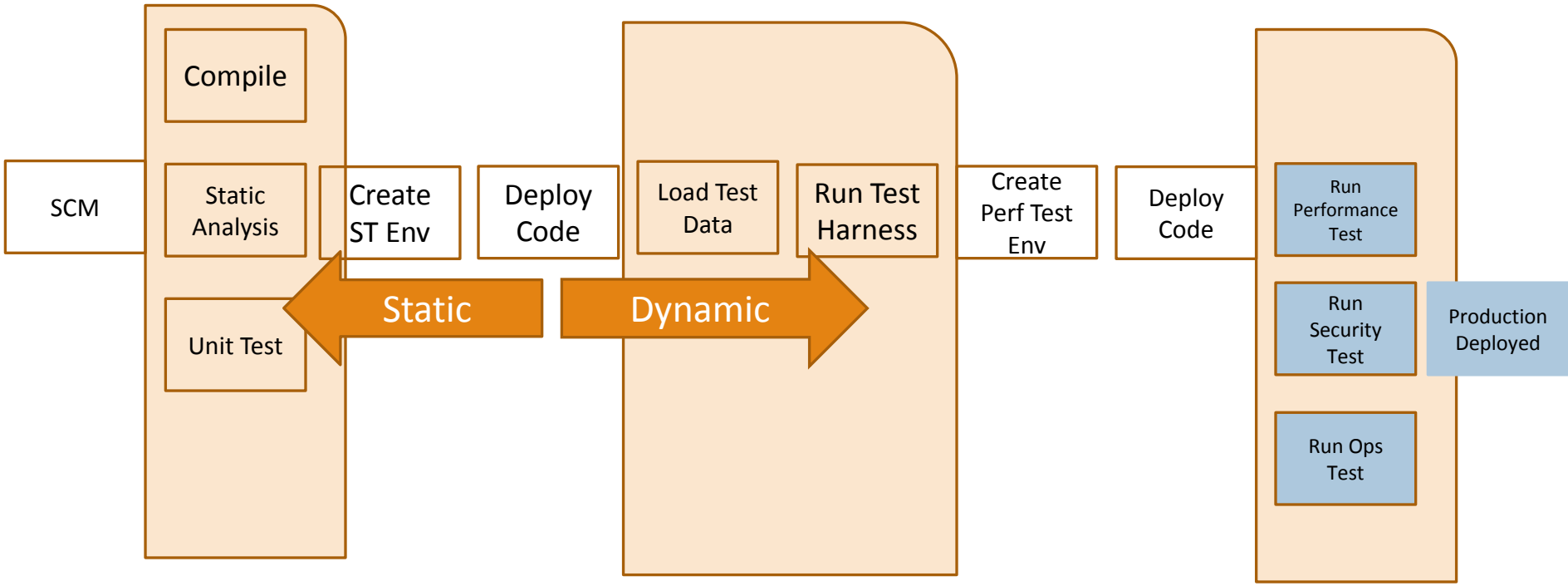
- Design Conformance
- Logic/Flow
- Backward Compatibility
- Lateral Compatibility
- Concurrency
- Internal Document
- Language Dependency
- Side Effect
- Rare Situation

QA/ VERIFICATION

- Unit Test
  - Simple Path
  - Complex Path
- Functional Test
- Test Coverage
  - Test Variation
  - Test Sequencing
  - Test Interaction
- System Test
  - Workflow/Stress
  - Recovery/Exception
  - Startup/Restart
  - HW Configuration
  - SW Config/ Blocked Test

Pipeline (Testing Stages)

Low level first progresses to the more complicated.



QA Automation (static)

Post-Commit Automated Build

- Run automated Unit Tests
- Run automated Static Code Analysis

SonarQube

CSS Lint

Food Critic

Cucumber

FitNesse

Junit

TestNG

QA Automation (dynamic)

Pre-Deployment

- Functional Suite
- Test suites against an environment
- UI or API driven
- Load Test

Selenium

Rational Robot

Sauce Labs

HP (functional)

Watir

Jmeter

Concordian

Casper.js

Watir

Autoit

QA Automation (dynamic)

Staged (last step before customer)

- Load Test
- Integration Test
- Performance and Security Testing
- Service Virtualization

Zed Attack Proxy

Snort

UpGuard

HP Test Suite

SOASTA

Appium

Tosca (service virtualization)

CA Lisa

Green Hat

Infrastructure CM

Manage the configuration of the infrastructure,

Infrastructure is under CM and is automated.

Infrastructure is imutable.

Chef

Puppet

SaltStack

Ansible

Kubernetes

Summary

Keys to DevOps

Plan small/ fail fast/ deliver quickly

Everything is under SCM

Test & Automation

People (Kaizen/ Quality Culture)

Infrastructure under CM

Time Waisted Quote

Time waste differs from material waste in that there can be no salvage. The easiest of all wastes and the hardest to correct is the waste of time, because wasted time does not litter the floor like wasted material.

- Henry Ford

Take Away Thoughts

DevOps enables SW teams to achieve a higher performance.

Small Batches and Quick Feedback

A Pipeline with Automated QA

Quality Culture/ Working Software/ Lower Costs

Continuous Improvement/ Lean

Sustainability and Sustainable Teams

Thank You