Mitchell CI/CD Journey Continues…

Richard Fong

(Raj) Rajwinder Singh Makkar

Introduction

Mitchell International

- Founded in 1946, with 70 years of experience in Property, Casualty claims, and Collision Repair industries
- 50 million transactions annually
- 300 insurance companies/claims payers
- Over 30,000 collision repair facilities
- 2,000 associates

Raj Makkar

Senior Configuration Engineer @ Mitchell

Developing solutions around software release automation, dependency management and continuous integration for various software products in Mitchell.

Currently promoting continuous integration and continuous delivery solutions to increase efficiency within Mitchell's software delivery pipeline.

Before Mitchell use to work for GE.

Richard Fong

CICD enthusiast with experience creating service oriented infrastructure to efficiently and rapidly build, integrate, and deliver quality software product for large enterprise.

Richard’s has worked with many known companies such as Yahoo, Intuit and Qualcomm in architecting their full CICD tool stacks and evangelized CICD for those organization.

Journey

Keep in mind it’s a journey …

7 Years Ago

- Componentized
- **Developer** and **SCM Team** maintain the Build
- **Developer** maintain the Build of Materials
- **Developer** decide when to Release
- Central **QA** Team run Test Manually

Issues

- **Dependency management** is a pain
- **Custom deployment** for each component
- Multi days planning just for **release coordination**

The

“BIG BUILD”

4 Years Ago

- Big Build
- **System** Team maintains the Build
- **System** Team maintain the Build of Materials
- **System** Team decide when to Release
- Central **QA** Team try to automate test

Issues

- The Build is **slow**
- Dependency and Deployment are **bottleneck** by a single team
- Developer **cannot run local** big build

“Left Is the New Right”

Today

- Componentized
- **Developer** maintain the Build
- **Developer** maintain the Build of Materials
- **Developer** decide when to Release
- Central **QA** Team run a mix of Manual and Automated Tests

Benefits

- Componentized with **Standard Packaging**
- Component is **Developer's responsibility**
- Build, Deploy, Test on check-in with **Immediate Feedback**

How the Journey Begin

	Componentized	Who control build dependencies	Who maintain packaging	Who decide on what to release	State of QA
7 years ago	Yes	Dev+scm team	N/A	Dev	Central QA team
4 years ago	Big build and no component build	System team	System team	System team	Being Agile ;-) + partial workflow QA team
Today	Yes	Dev	Dev	System team	Central QA team

mitchell

“Build Highway, Not Traffic Light”

Lots of Terms

- CICD
- DevOps
- TDD
- BDD
- Don’t break the build
- Unit Test
- Pair programming

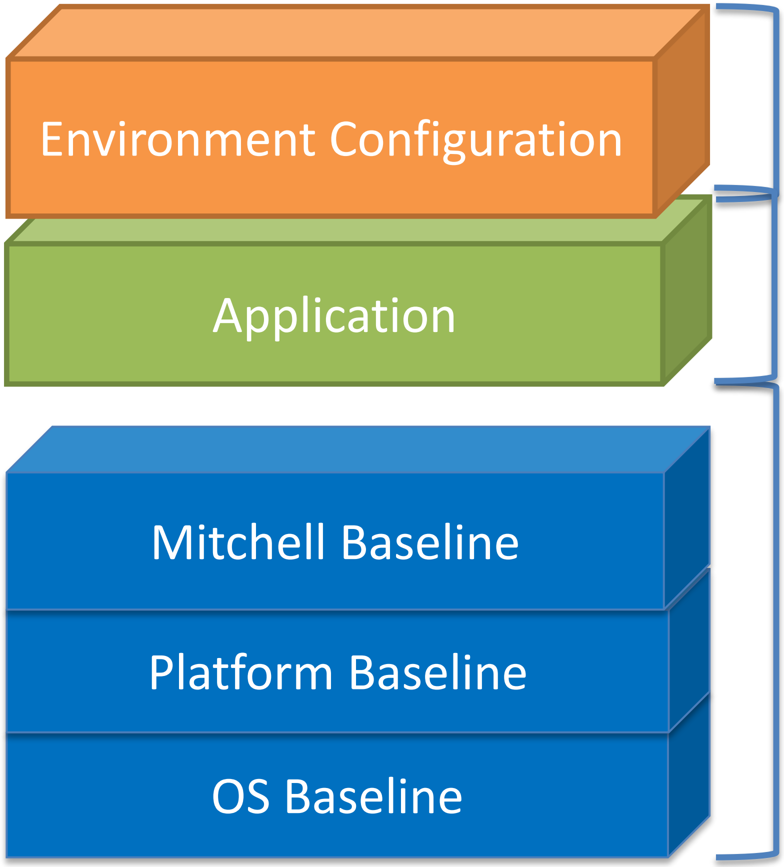
The Mission!!!

Is not about the terminology, but our Mission

to “Deliver Quality Product ***FAST***”

Tools

VM Image Layer

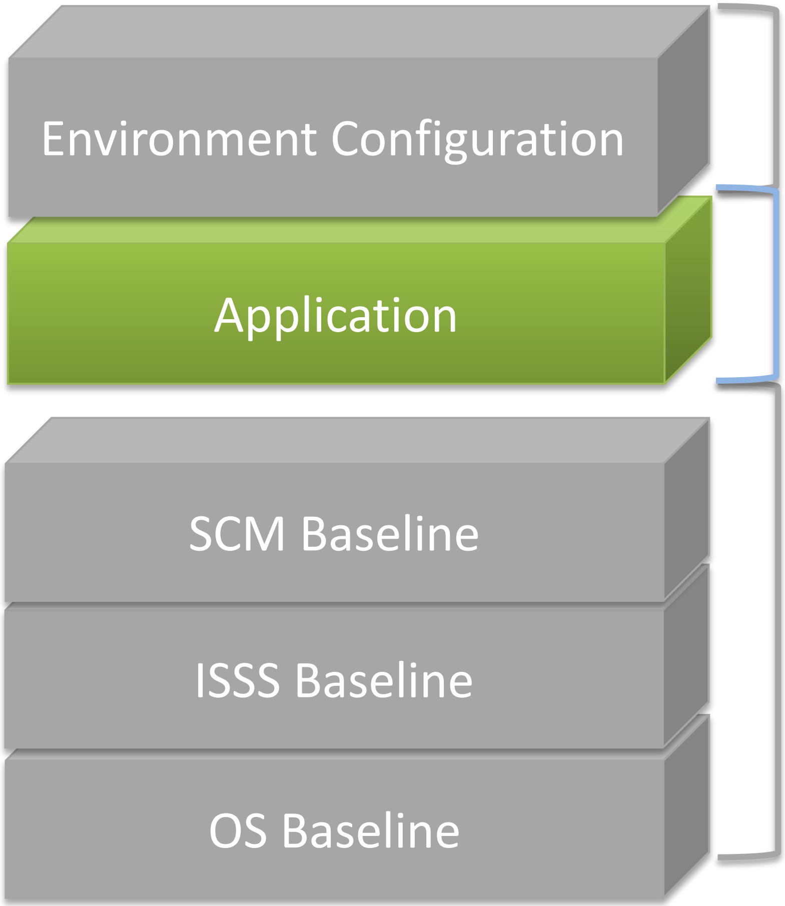


Environment Configuration

Application Deployable
Artifacts

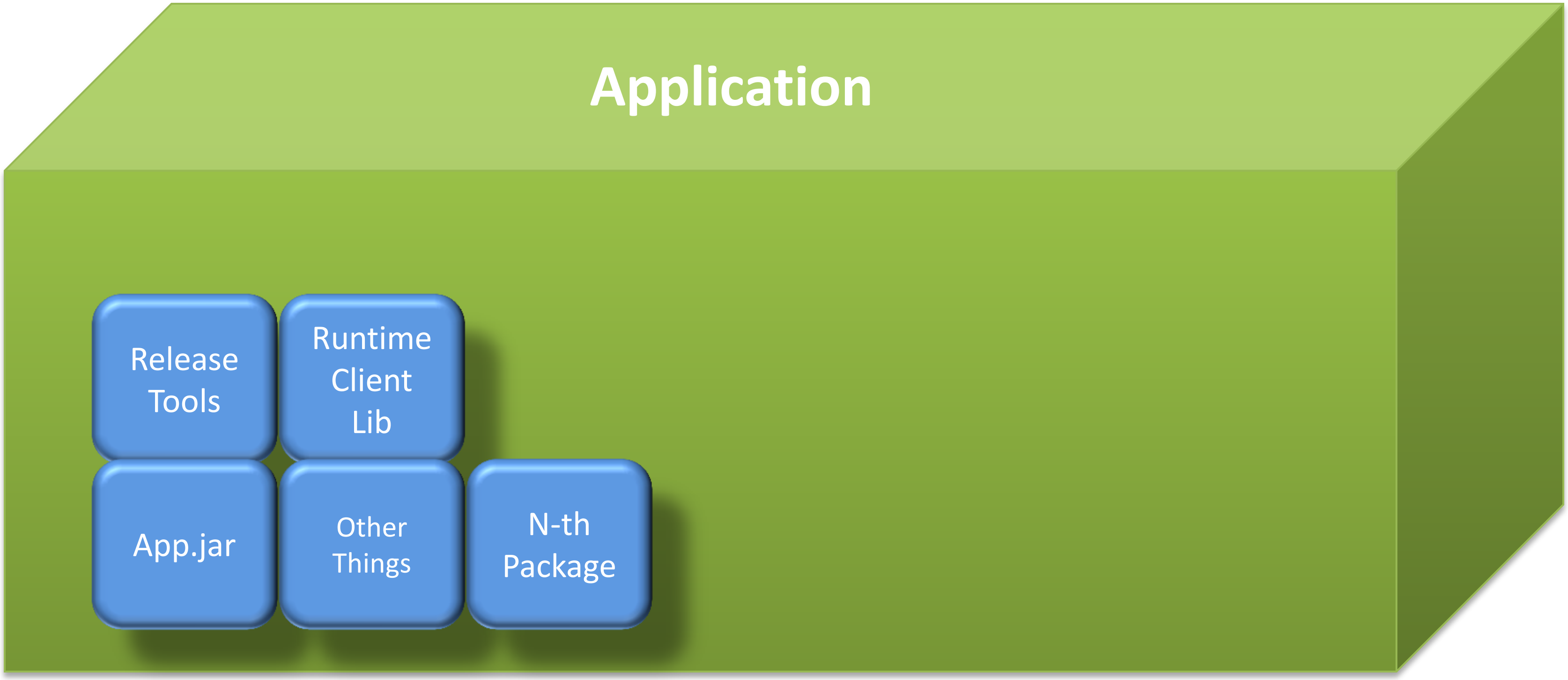
Image Baseline

Application Layer

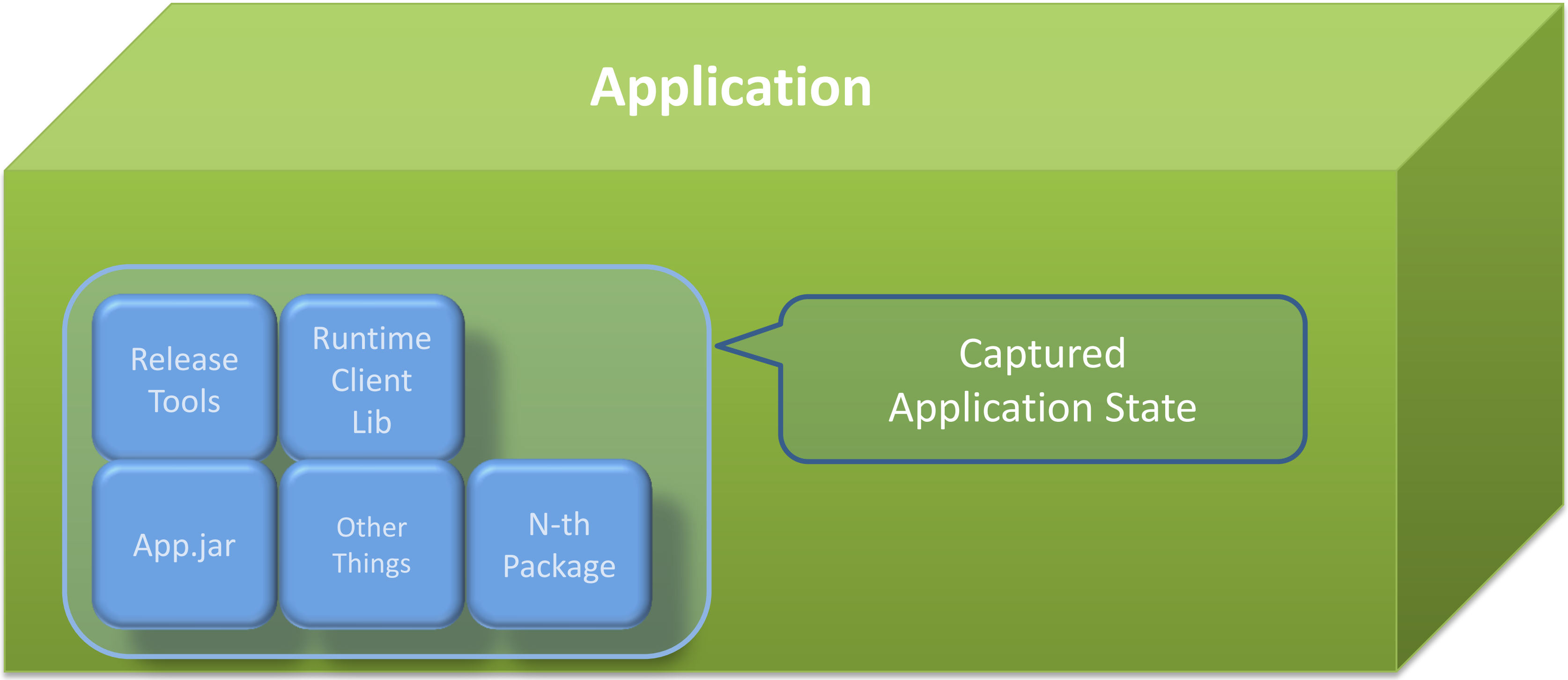


- **The Delta**
- **Reproducible**
- **Flexible**
- **Simple to use**

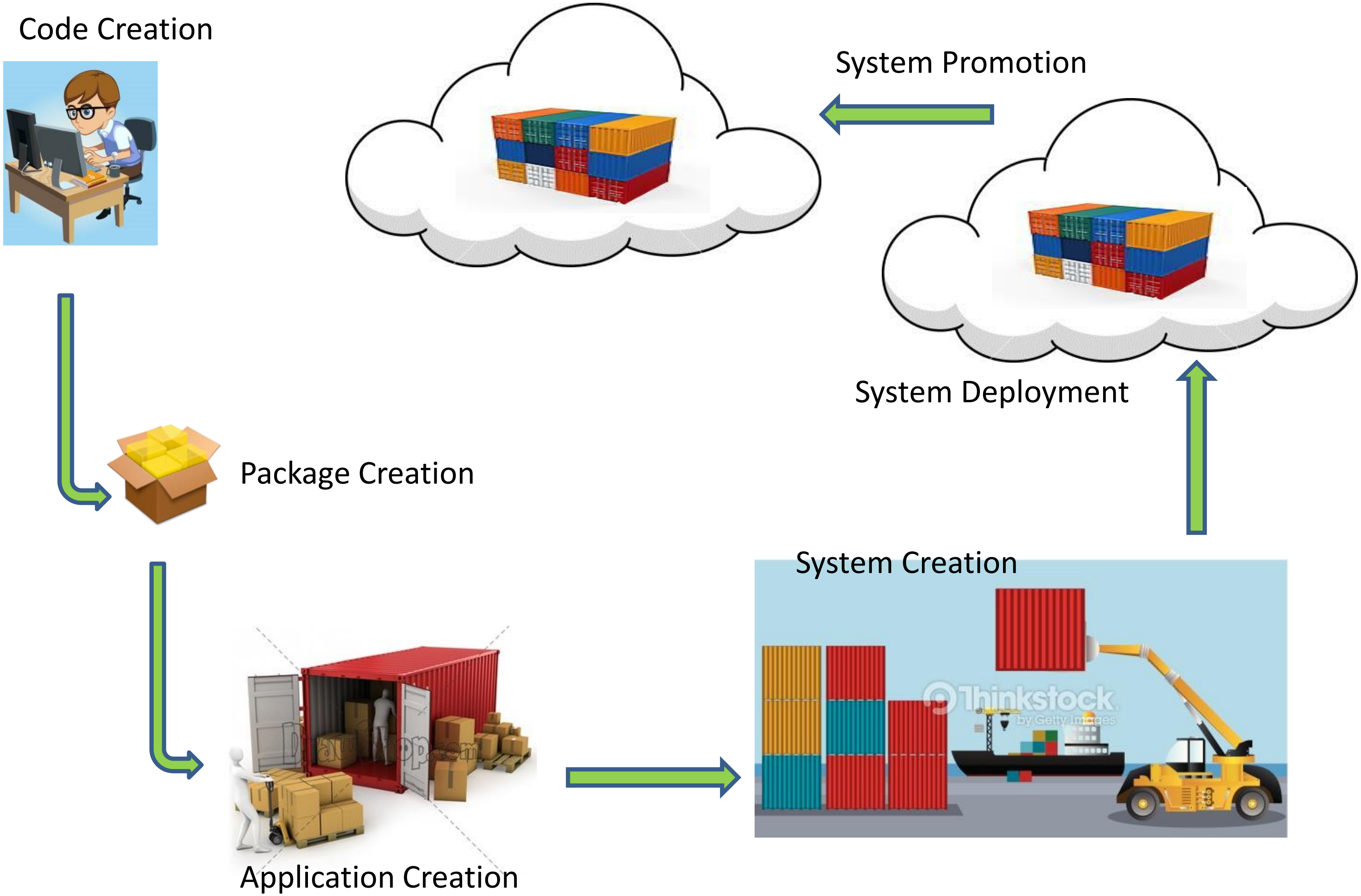
Packages


Application
Release Tools
Runtime Client Lib
App.jar
Other Things
N-th Package

State


Application
Release Tools
Runtime Client Lib
App.jar
Other Things
N-th Package
Captured Application State

Vision on Shipping Enterprise Application



Code Creation

Package Creation

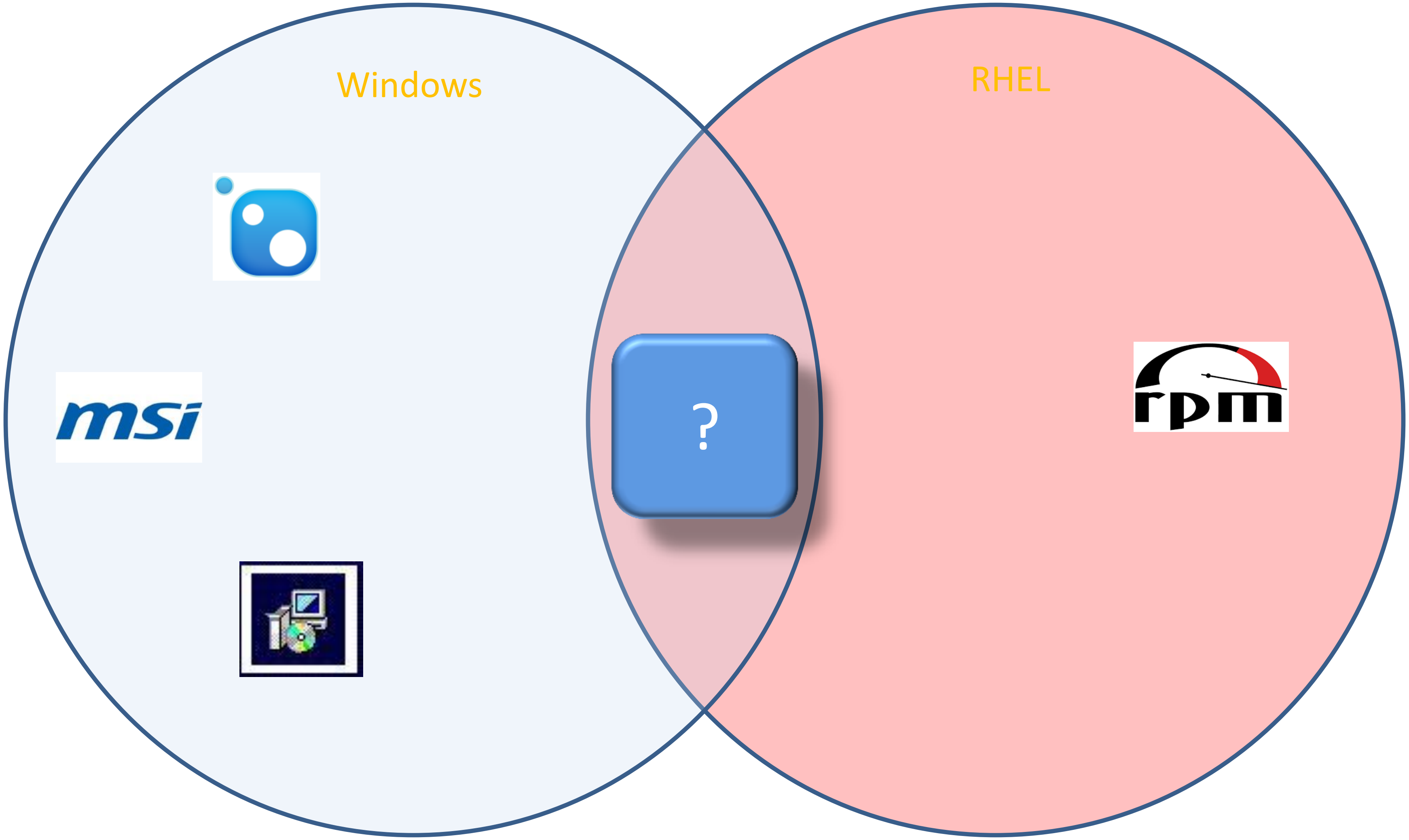
Application Creation

System Creation

System Deployment

System Promotion

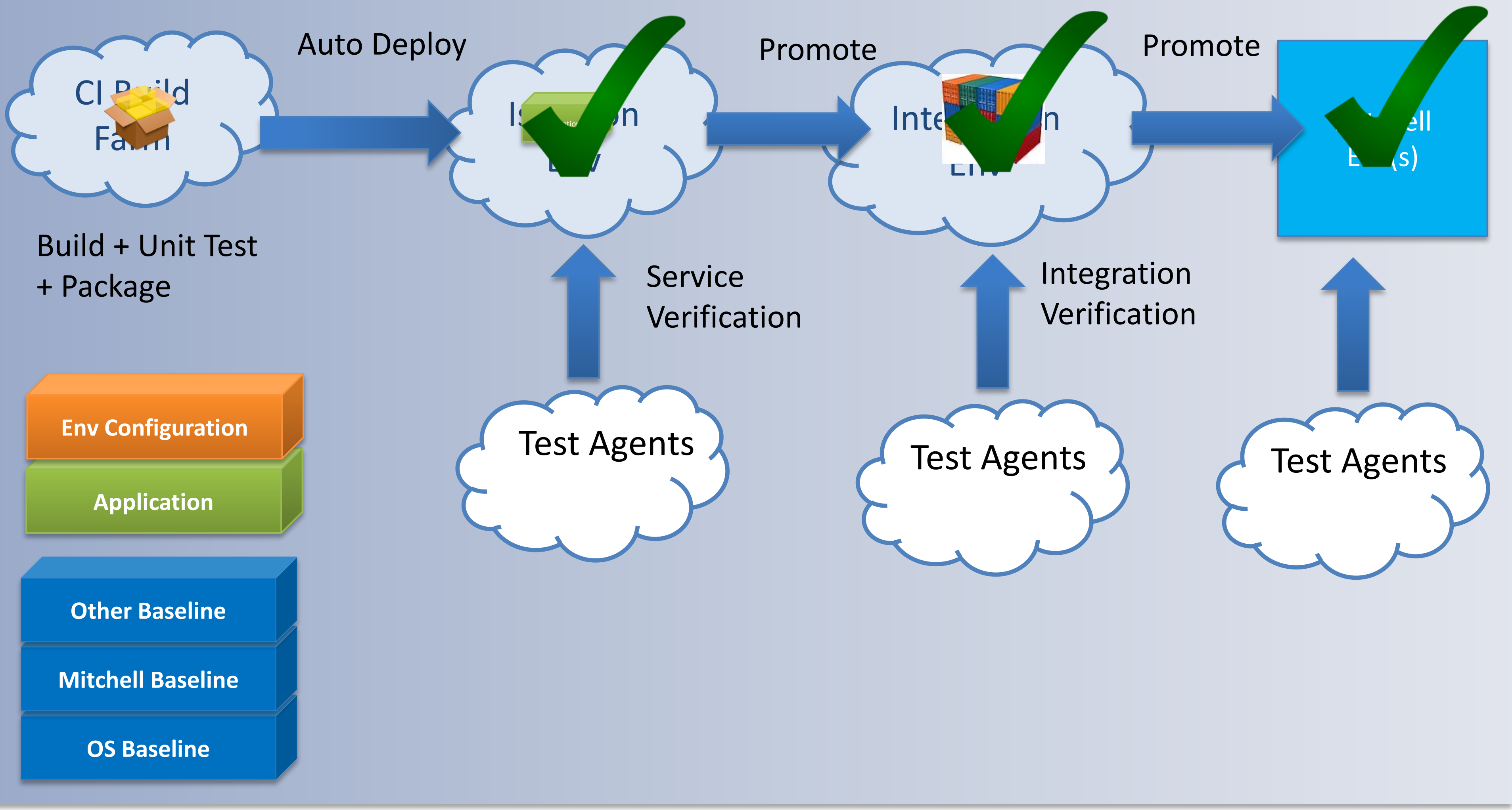
One Tool to Rule Them All



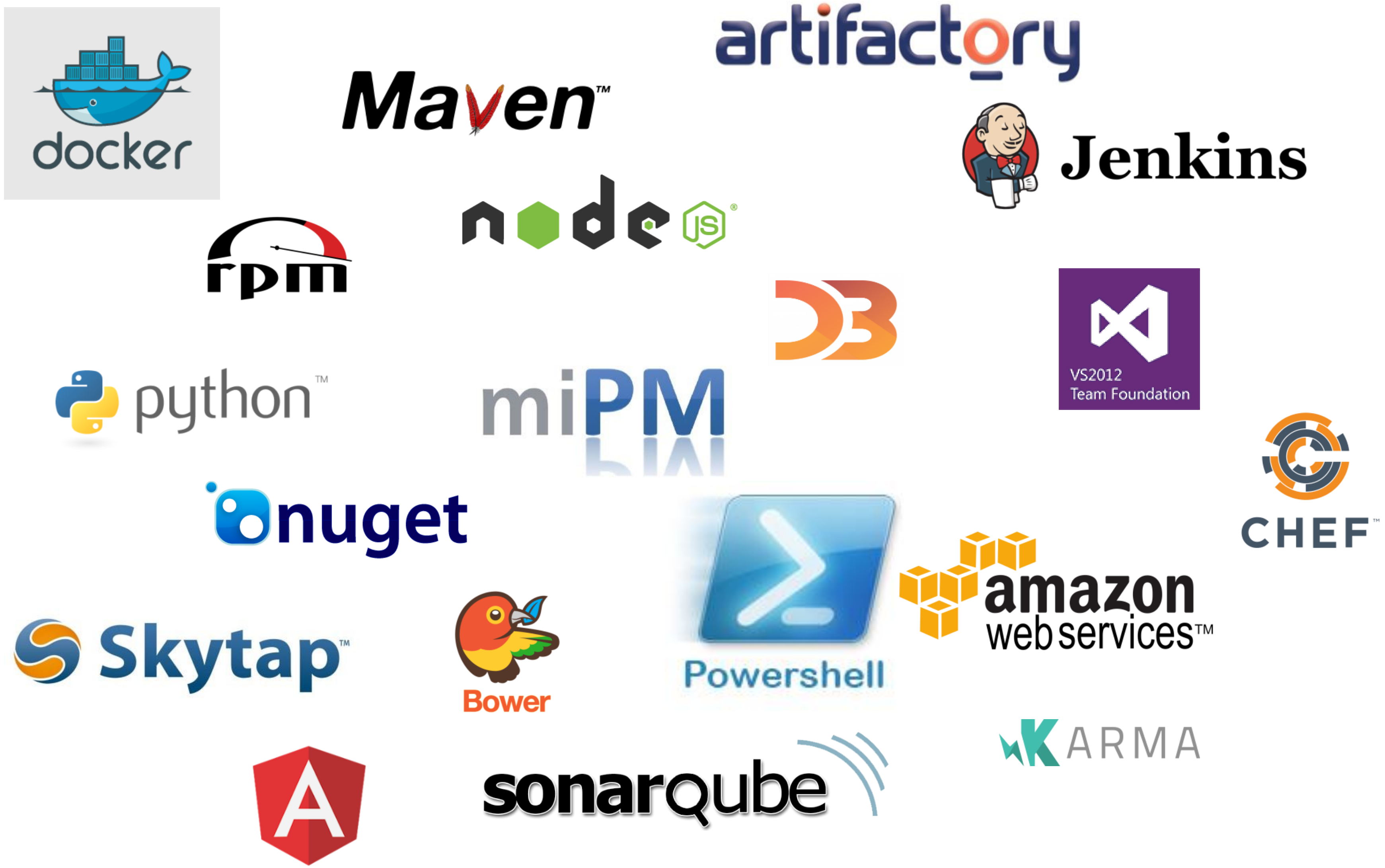
Deployment Standard

- Package to manage files
- State to manage packages
- System to manage States

Continuous Delivery Pipeline



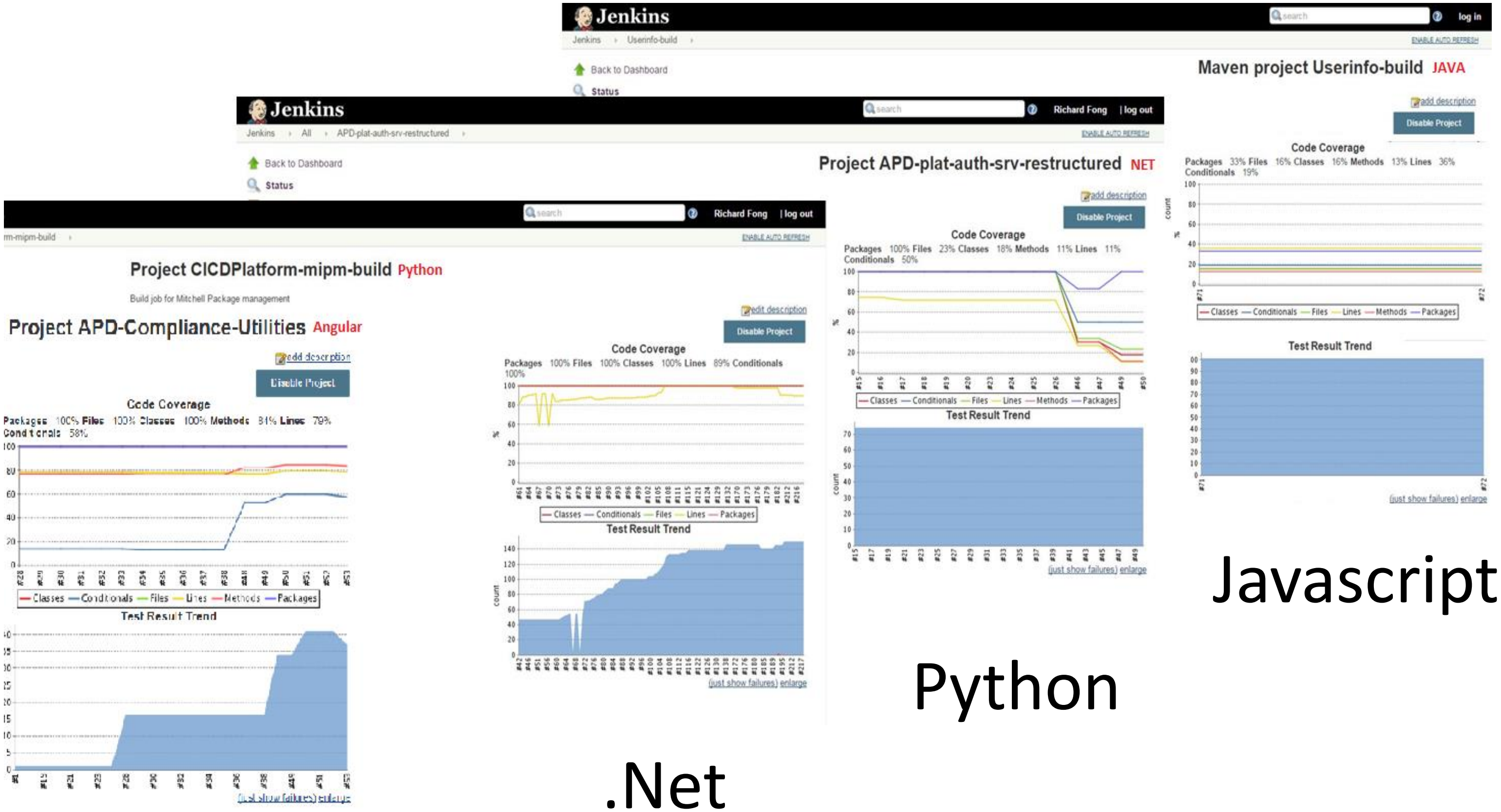
Many Technologies are Used



Use Jenkins/Artifactory

- From 1 Big Build to 1000 Component Build, Deploy, Test jobs
- Do about 1400 job runs / day
- Average life of slave is 12 hours
- Use of Docker and AWS to scale Jenkins Slaves
- Use of push model rather then polling for check-ins (loosely coupled)
- New version of artifact for each check-in
- Artifactory serve as the centralized artifact repository
- 800+ GB in about 12 months

Same Data for Reporting


Jenkins
Maven project Userinfo-build JAVA
Jenkins
Project APD-plat-auth-srv-restructured NET
Project CICDPlatform-mipm-build Python
Build job for Mitchell Package management
Project APD-Compliance-Utilities Angular
Back to Dashboard
Status
Richard Fong | log out
log in
add description
edit description
Disable Project
Code Coverage
Packages 100% Files 100% Classes 100% Methods 81% Lines 79% Conditionals 58%
Packages 100% Files 100% Classes 100% Lines 89% Conditionals 100%
Packages 100% Files 23% Classes 18% Methods 11% Lines 11% Conditionals 50%
Packages 33% Files 16% Classes 16% Methods 13% Lines 36% Conditionals 19%
Classes — Conditionals — Files — Lines — Methods — Packages
Test Result Trend
count
(just show failures) enlarge

Javascript

Python

.Net

Java

Pipeline as Code

- Too many builds to manage manually
- Pipeline defined as JSON configuration
- Let Developer maintain their own pipelines
- Ability to create a Pipeline for Branching
- Job template to allow
    - Re-usable for different projects
    - Flexible in adopting new technology

Processes

Category of CI/CD Capabilities

Build

Continuous building software and validate new source code

Deploy

Continuous deploying software and validate new compiled code

Test

Continuous integration testing for software cohesive functionalities

Promote

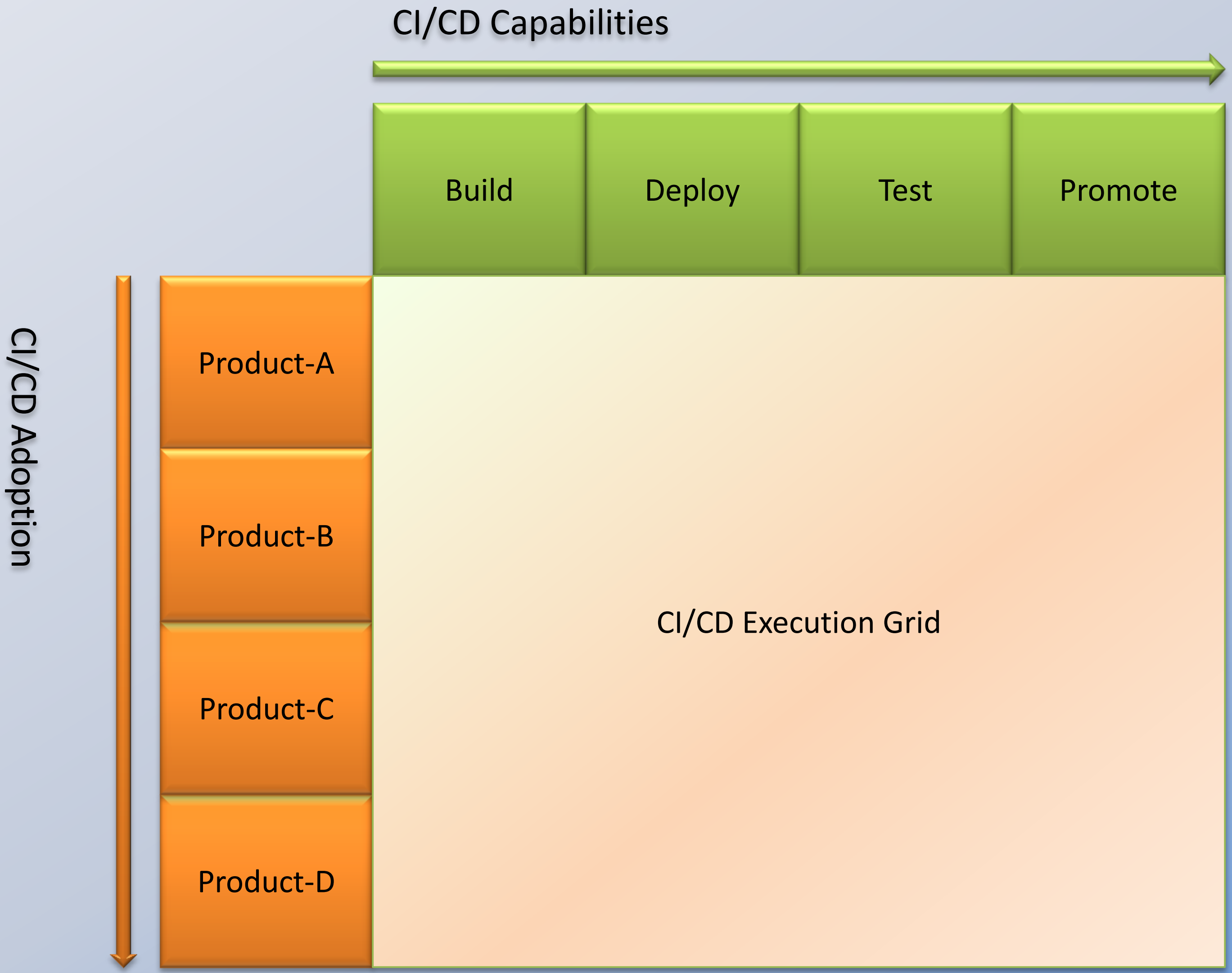
Continuous promotion of software for feature availability to market

Product Taxonomy

Product

- Systems
- Services
- Components
- Libraries
- Ownerships

Capabilities and Adoption



CI/CD Capabilities

Build

Deploy

Test

Promote

CI/CD Adoption

Product-A

Product-B

Product-C

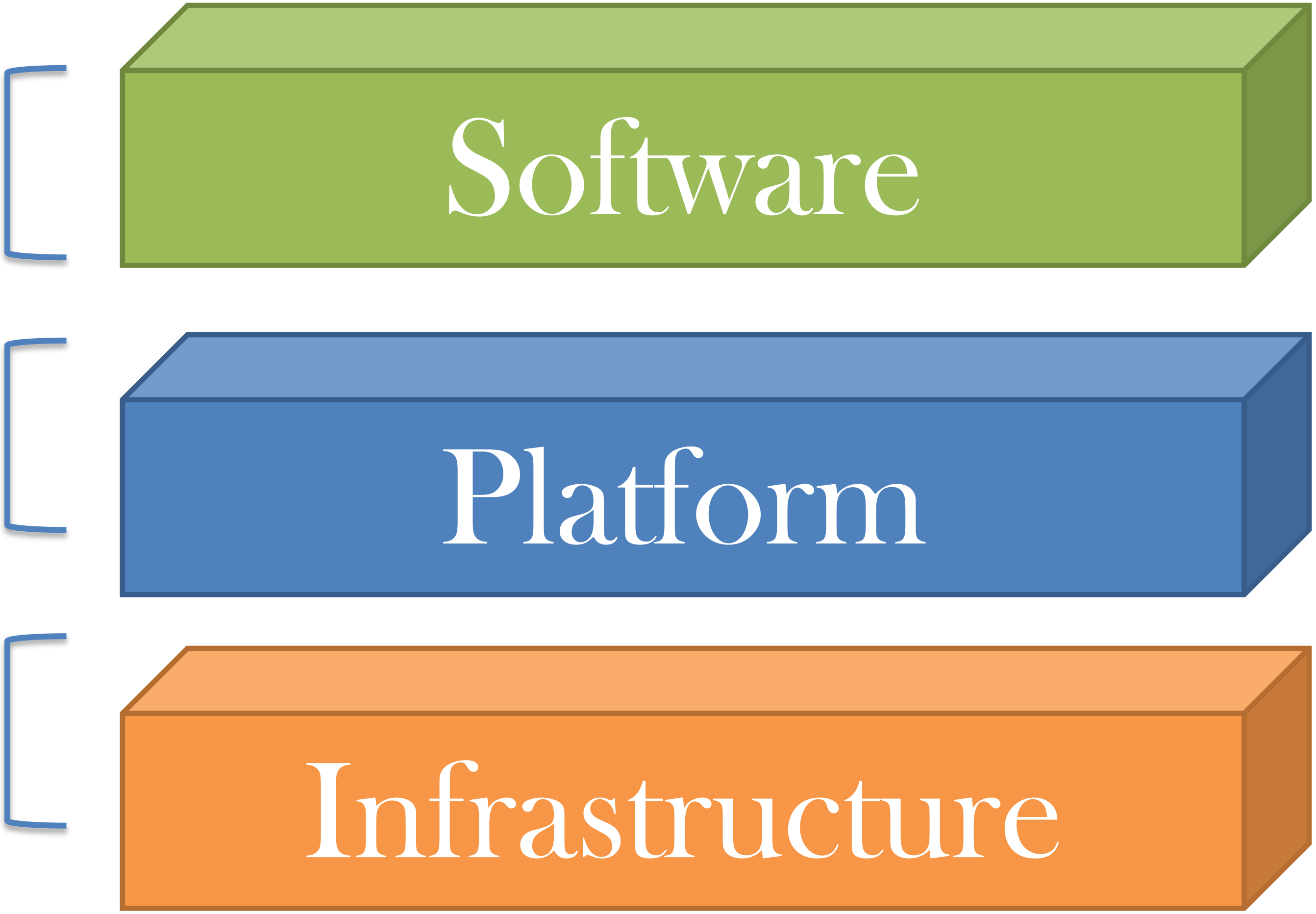
Product-D

CI/CD Execution Grid

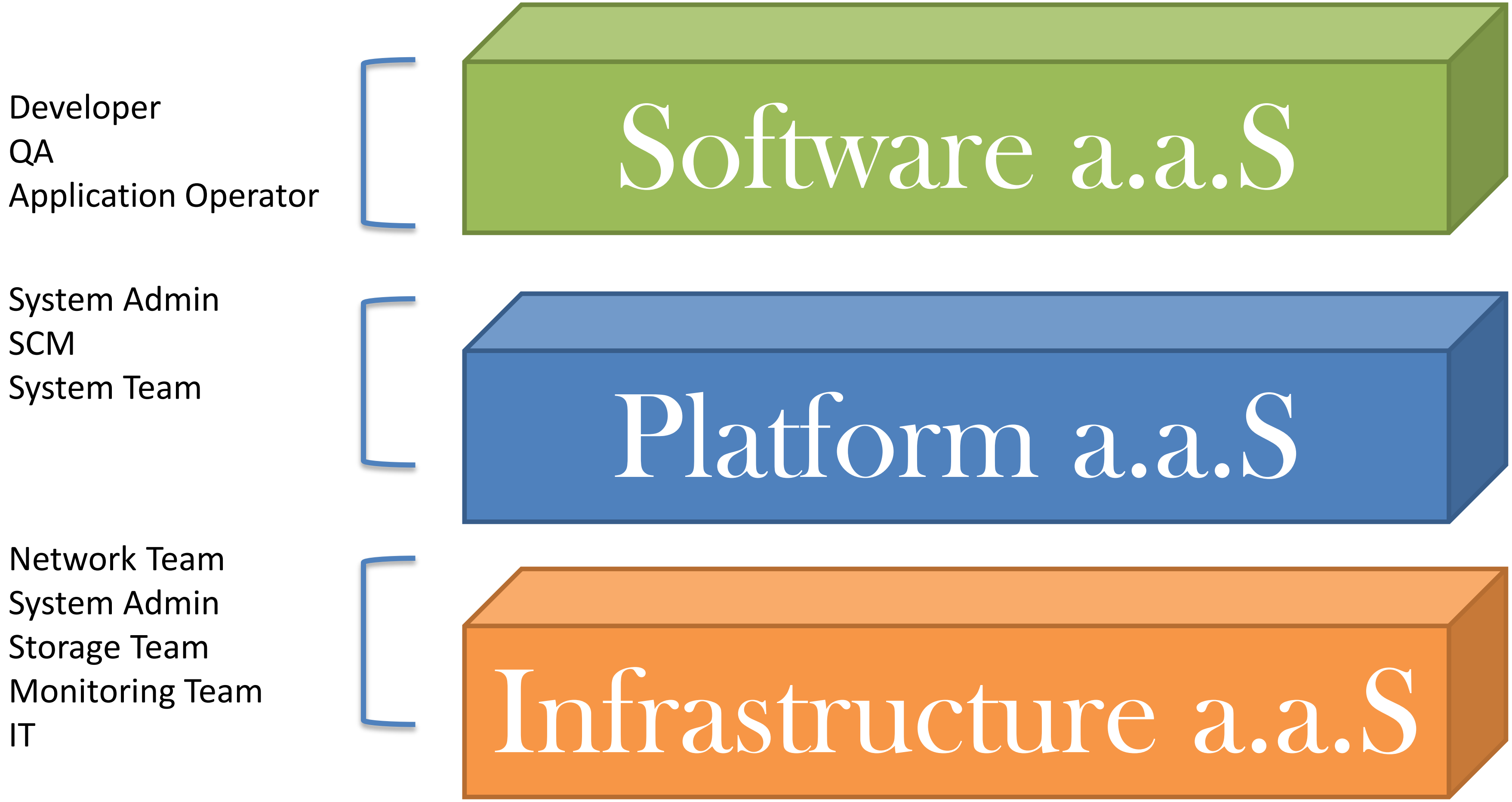
Technology Not Yet as a Service

JBoss
IIS
Apache
Oracle
SQL Server

Network
System or VM
Storage
Backup
Monitoring



Whose "aaS" do you kick?



Developer
QA
Application Operator

Software a.a.S

System Admin
SCM
System Team

Platform a.a.S

Network Team
System Admin
Storage Team
Monitoring Team
IT

Infrastructure a.a.S

Dev, QA, Ops, SCM, IT – Fragmented pipeline with walls

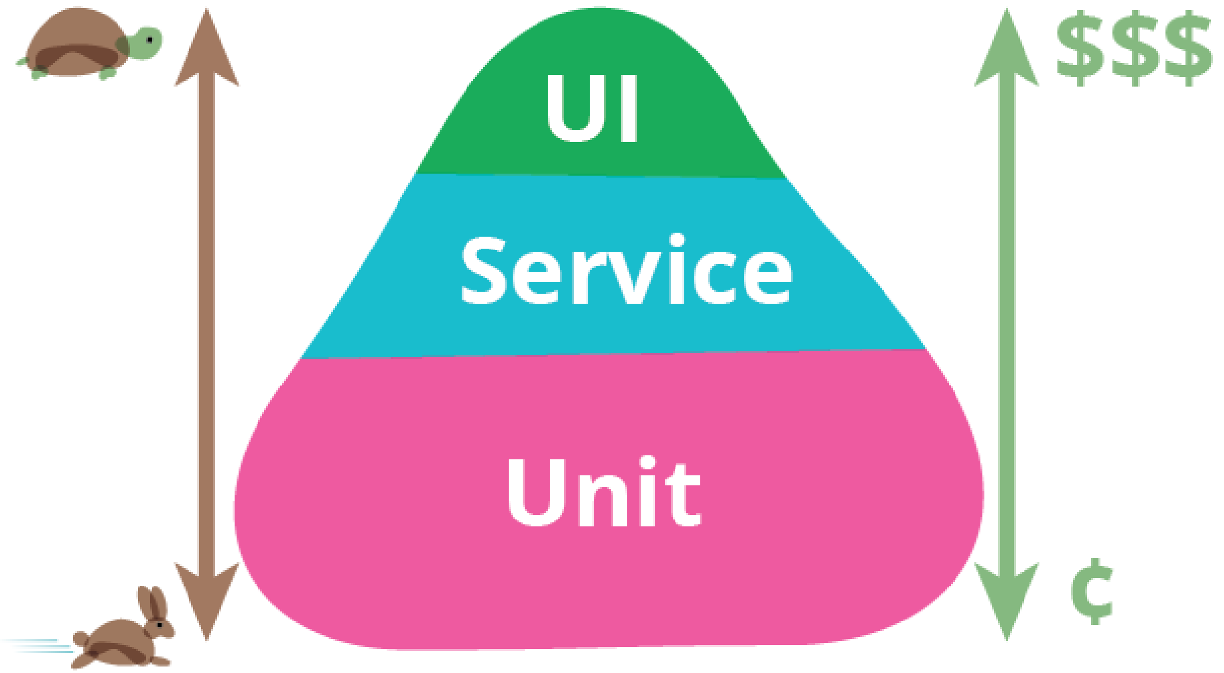
Dev	- We completed our story and task given by PO
QA	- We tested what we think we need to deliver to customer
OPS	- I hope Dev did well and QA tested it well
SCM	- Code is in version control , we can track it.
IT	- We need to keep hardware up

Left Is the New Right

- Dev -> QA -> Ops -> SCM -> IT
- Pushing ownership to Developer
- You build it, you own

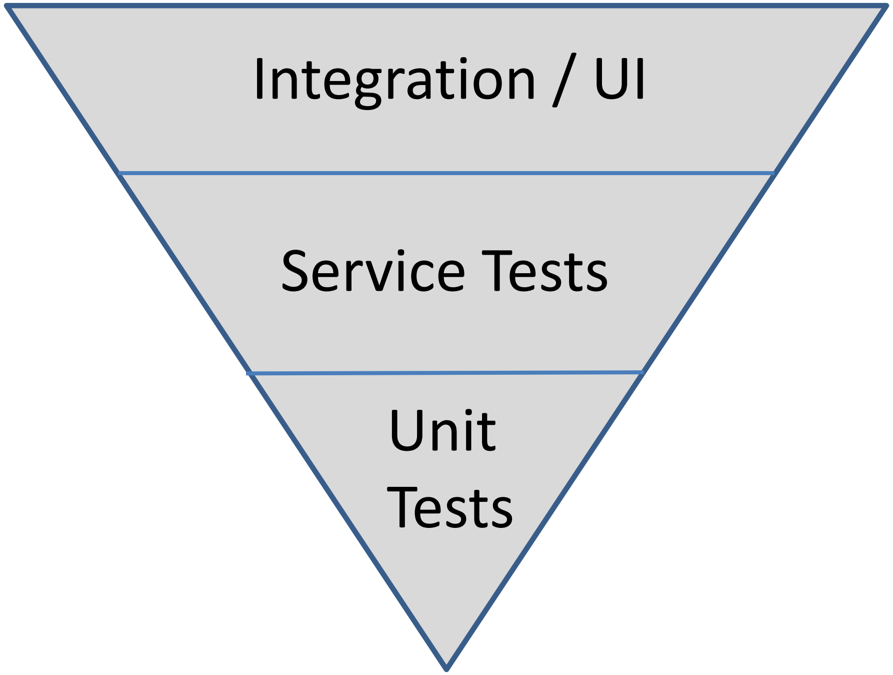
Test Pyramid

Martin Fowler: Test Pyramid

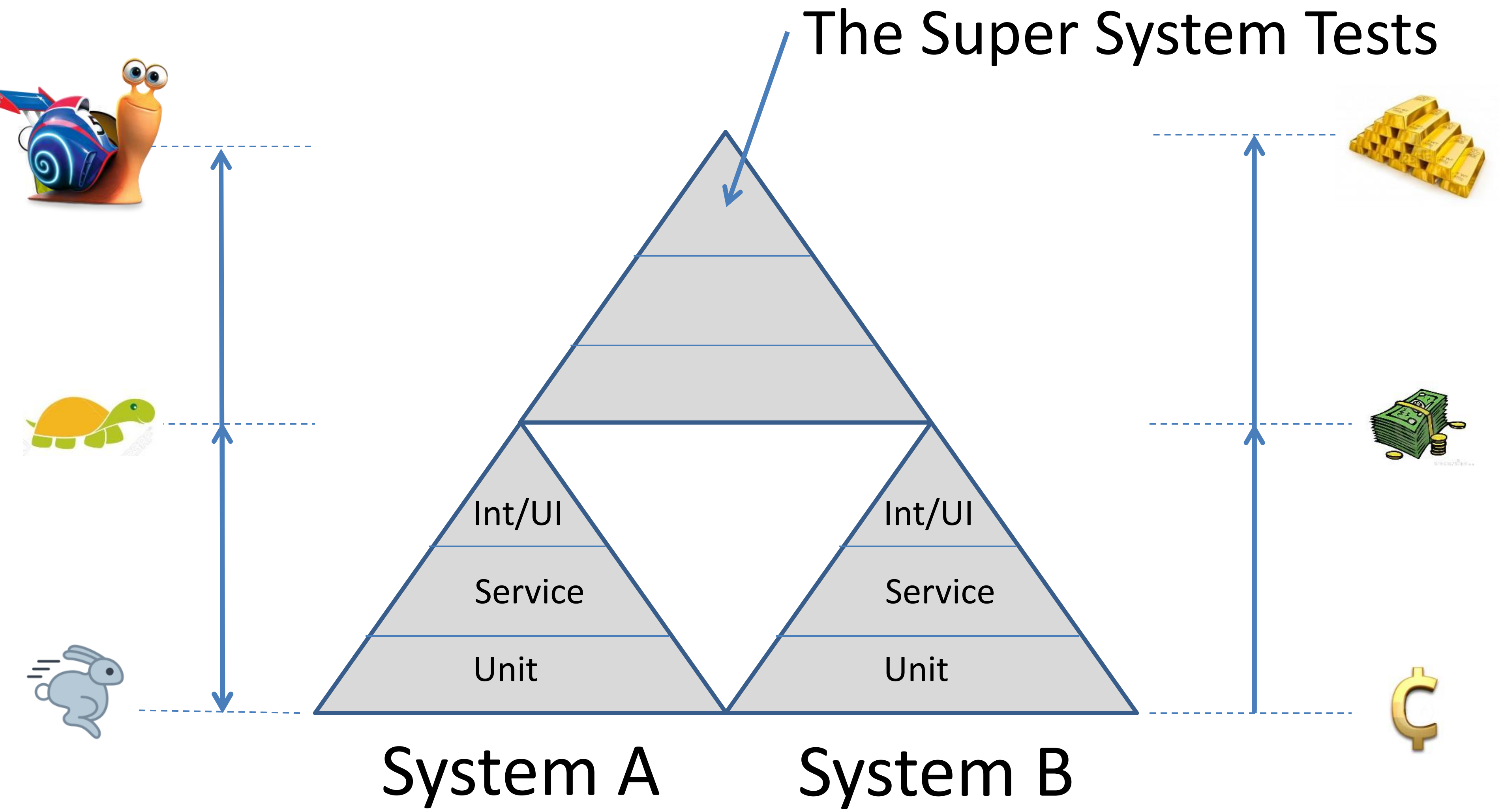


Reference: http://martinfowler.com/bliki/TestPyramid.html

Upside-Down Pyramid



Pyramid on Top of Pyramid



The Super System Tests

Int/UI

Service

Unit

Int/UI

Service

Unit

System A

System B

Software Quality Anti-Pattern

- High level integration test is better than no test
- Manual QA transitioning to Automation does not work
- No emphasis on Unit Test
- Big Build cannot be run on local machine

Software Quality Done Right

- Emphasis on Unit Test
- BDD to connect feature scenario sign off and test
- Service testing to eliminate external dependency
- Leverage local machine and VMs for functional validation
- Test Category
- Run all your tests

“Pipeline, Bucket of Sewage Water without Test”

Where Developer Spent Their Time

- Development Work
- Build fixes
- Deployment
- Support
- Meetings
- Design
- Test Support

We Don’t Support BS

- Do the right thing, not just automating current process
- Is the process event right
- People focus on current process and forgot about the reason

Culture

Some aspects …

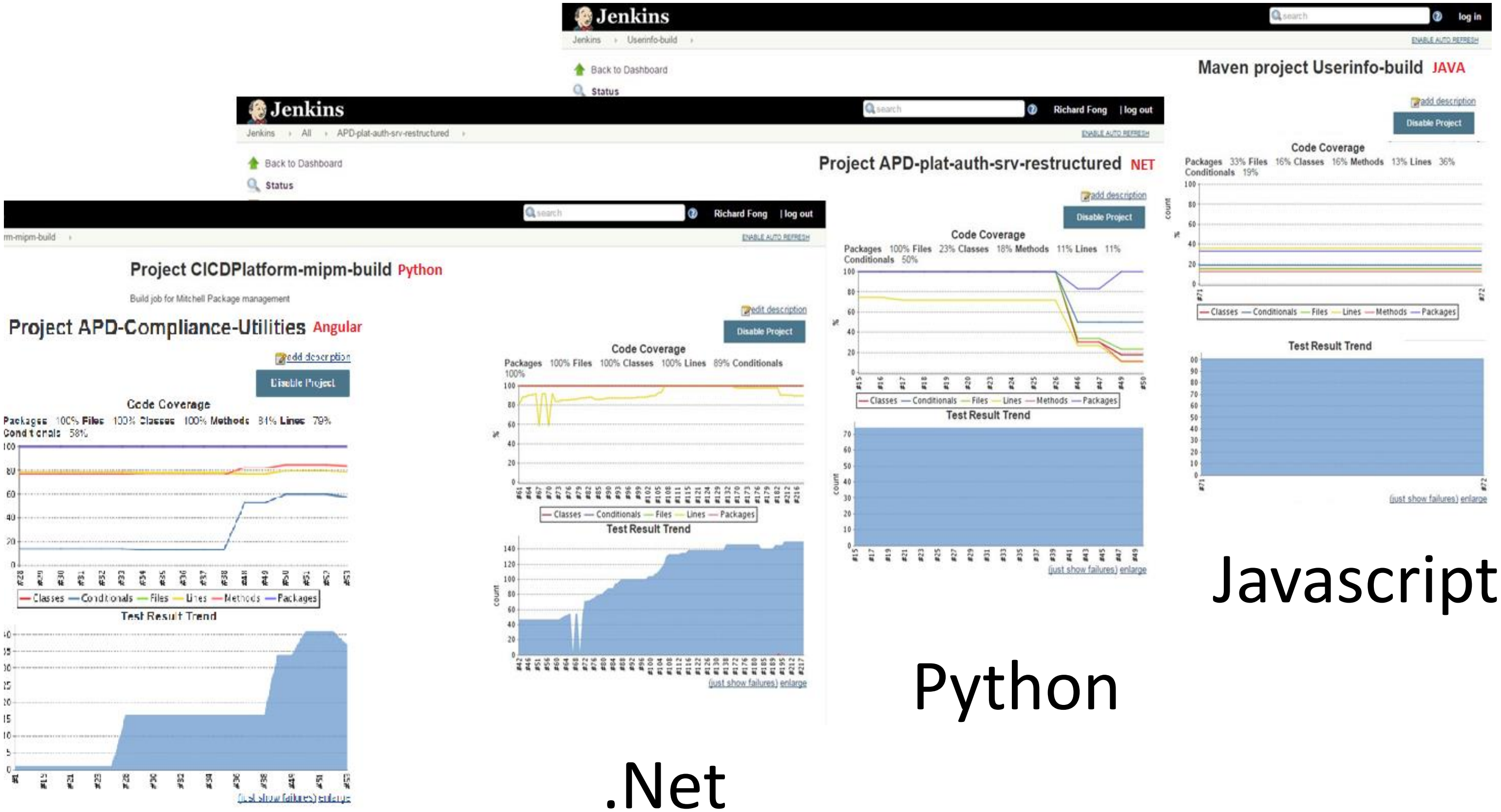
CICD and Quality is CEO’s Priority

Dealing with Merger and Acquisition

Acceleration to cloud related efforts

No big blue print designs before starting

Speak the same language across BUs



Jenkins | search | log in

Jenkins › Userinfo-build ›

ENABLE AUTO REFRESH

Back to Dashboard

Status

Maven project Userinfo-build JAVA

add description

Disable Project

Code Coverage

Packages 33% Files 16% Classes 16% Methods 13% Lines 36% Conditionals 19%

Test Result Trend

(just show failures) enlarge

Jenkins | search | Richard Fong | log out

Jenkins › All › APD-plat-auth-srv-restructured ›

ENABLE AUTO REFRESH

Back to Dashboard

Status

Project APD-plat-auth-srv-restructured NET

add description

Disable Project

Code Coverage

Packages 100% Files 23% Classes 18% Methods 11% Lines 11% Conditionals 50%

Test Result Trend

(just show failures) enlarge

search | Richard Fong | log out

rm-mipm-build ›

ENABLE AUTO REFRESH

Project CICDPlatform-mipm-build Python

Build job for Mitchell Package management

edit description

Disable Project

Code Coverage

Packages 100% Files 100% Classes 100% Lines 89% Conditionals 100%

Test Result Trend

(just show failures) enlarge

Project APD-Compliance-Utilities Angular

add description

Disable Project

Code Coverage

Packages 100% Files 100% Classes 100% Methods 81% Lines 79% Conditionals 58%

Test Result Trend

(just show failures) enlarge

%
count
Classes — Conditionals — Files — Lines — Methods — Packages
Classes — Conditionals — Files — Lines — Packages

Java

.Net

Python

Javascript

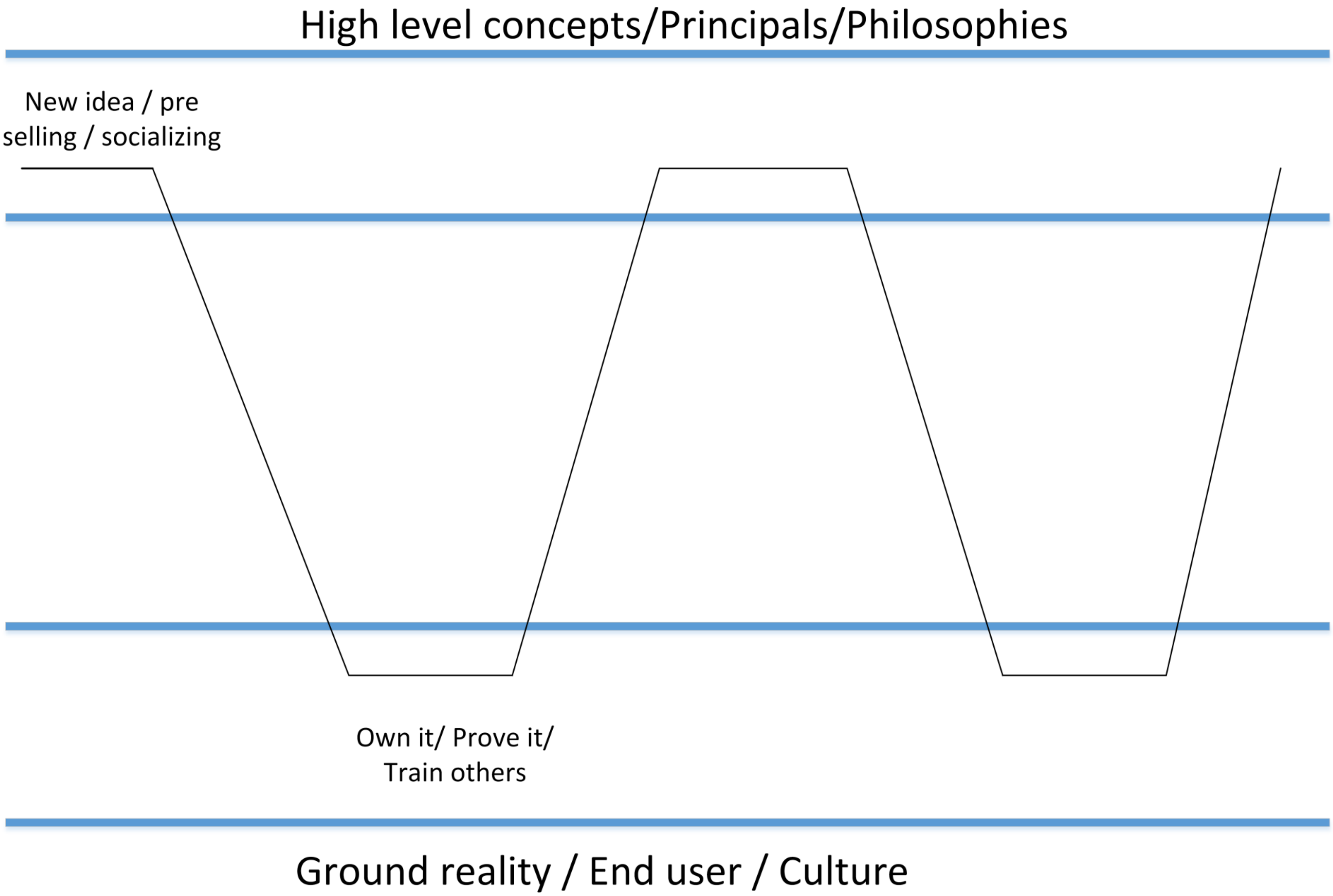
Should We Break the Build

- Builds are broken, and no one is looking at it
- Don’t break the build
- With TDD, break the build, then fix the build

Hiring Challenges

- Hiring DevOps? You want a Developer? Or an Operation Expert?
- Developer with no Ops/System/Network experience
- System Admin with no Development experience
- SCM in the middle but weak on both ends
- DevOps is a Culture and it is in all of us

Influence Changes Graph


High level concepts/Principals/Philosophies
New idea / pre selling / socializing
Own it/ Prove it/ Train others
Ground reality / End user / Culture

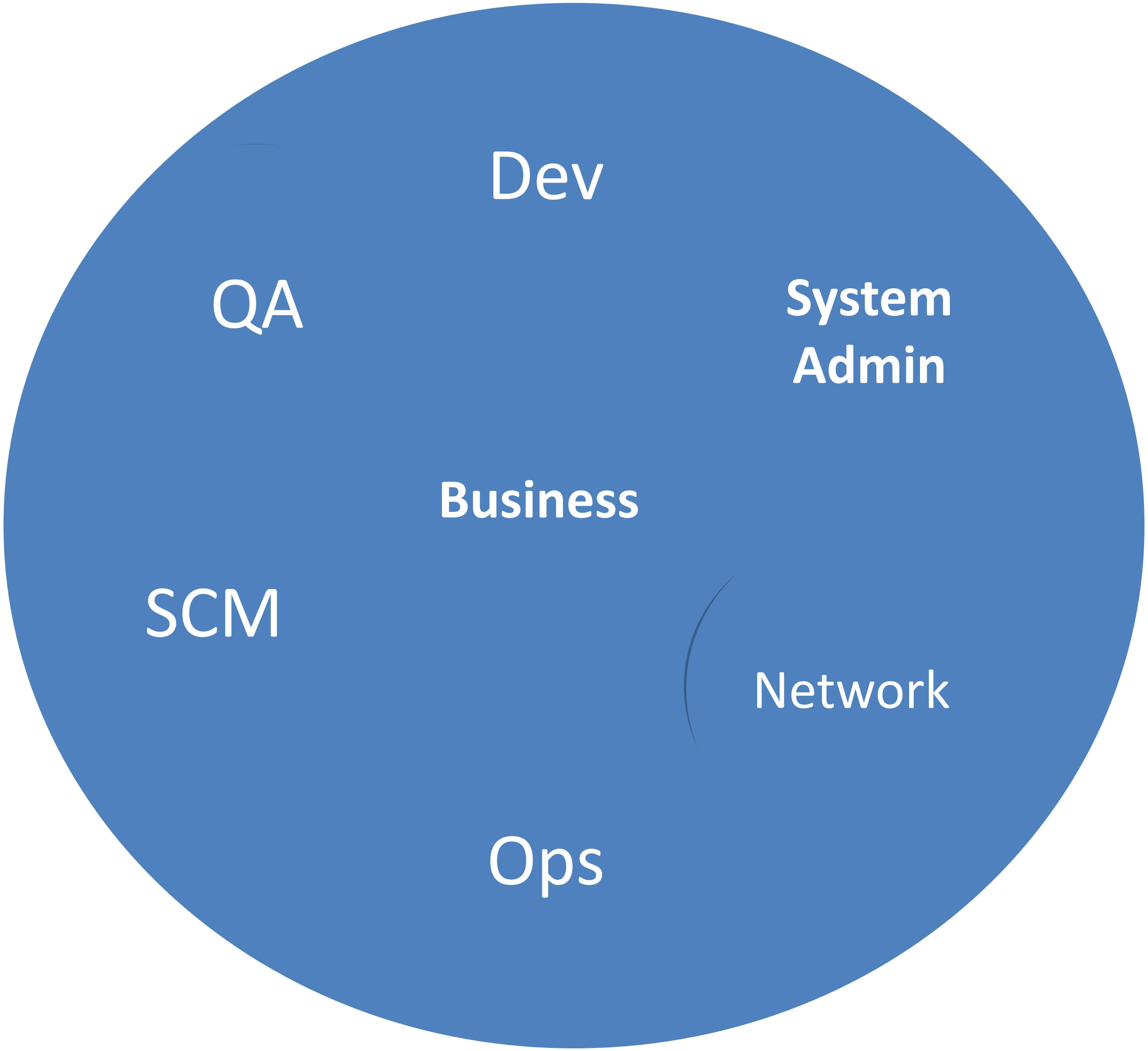
Everything is continuous … including challenges ;-)

- Exploring capabilities of TFS in diff areas
- Multiple platforms support for development teams
- How cloud fit in this whole picture
- Exploring options for information radiators
- Exploring options end to end orchestration
- Promote “culture of reuse” for our chef recipes

Training Within

- Cross pollination (other wants to learn)
- Internal Operational Manuals for other teams
- AWS Study Group and Certification

One Team



mitchell

- **Raj Makkar**
  - raj.makkar@mitchell.com
  - LinkedIn: https://www.linkedin.com/in/rajwinder-singh-83166718

- **Richard Fong**
  - richard.fong@mitchell.com
  - LinkedIn: https://www.linkedin.com/in/richardfong1