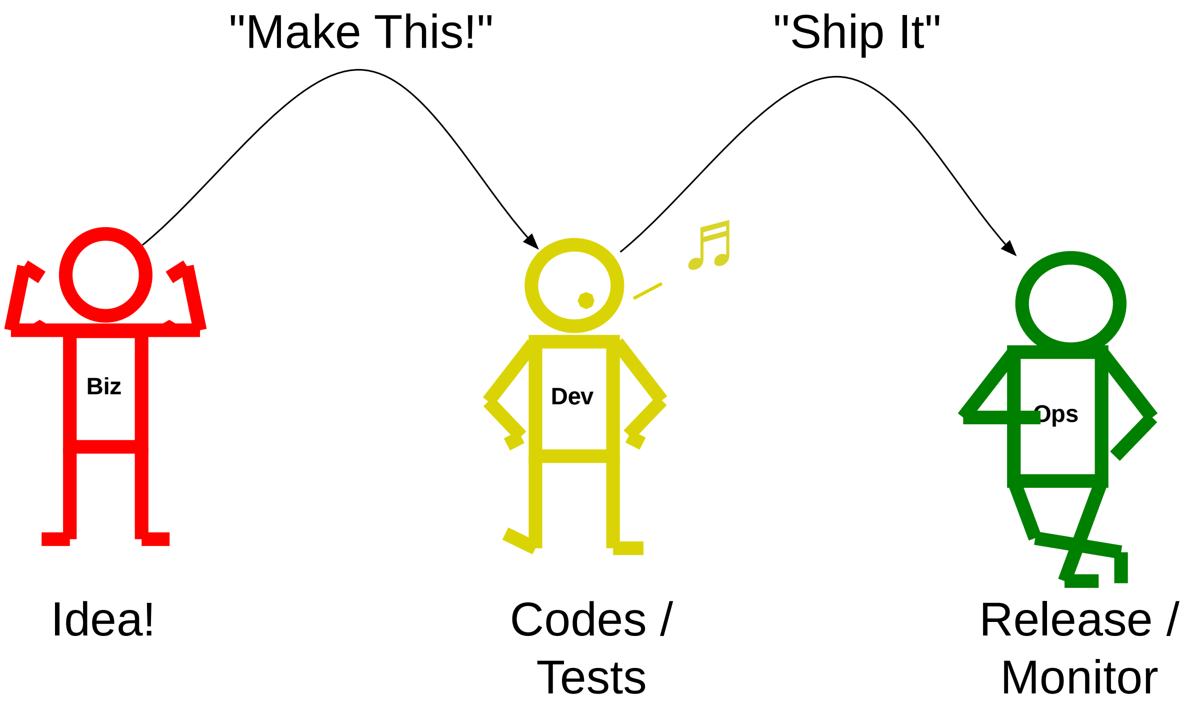
IBM Software

A DevOps Toolchain

There and back again

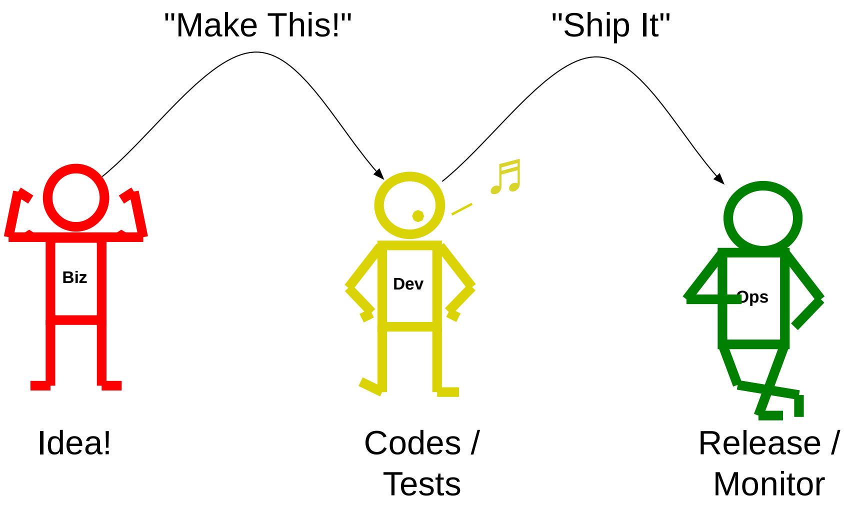
Eric Minick
eminick@us.ibm.com
@EricMinick

Software Delivery in a Nutshell

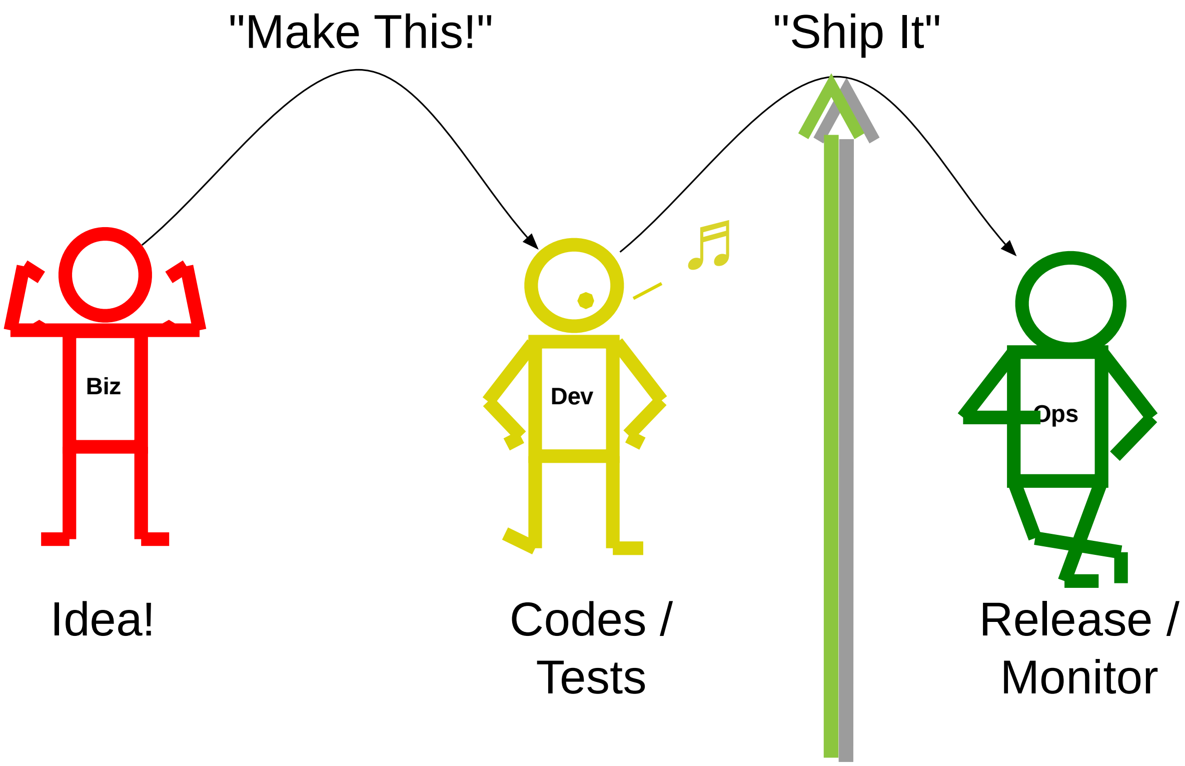


Apps: More Important / More Complex

- More frequent changes
- Lower tolerance for outages
- More complex applications
- More complex deployments



Software Delivery in a Nutshell



"Make This!"

"Ship It"

Biz

Dev

Ops

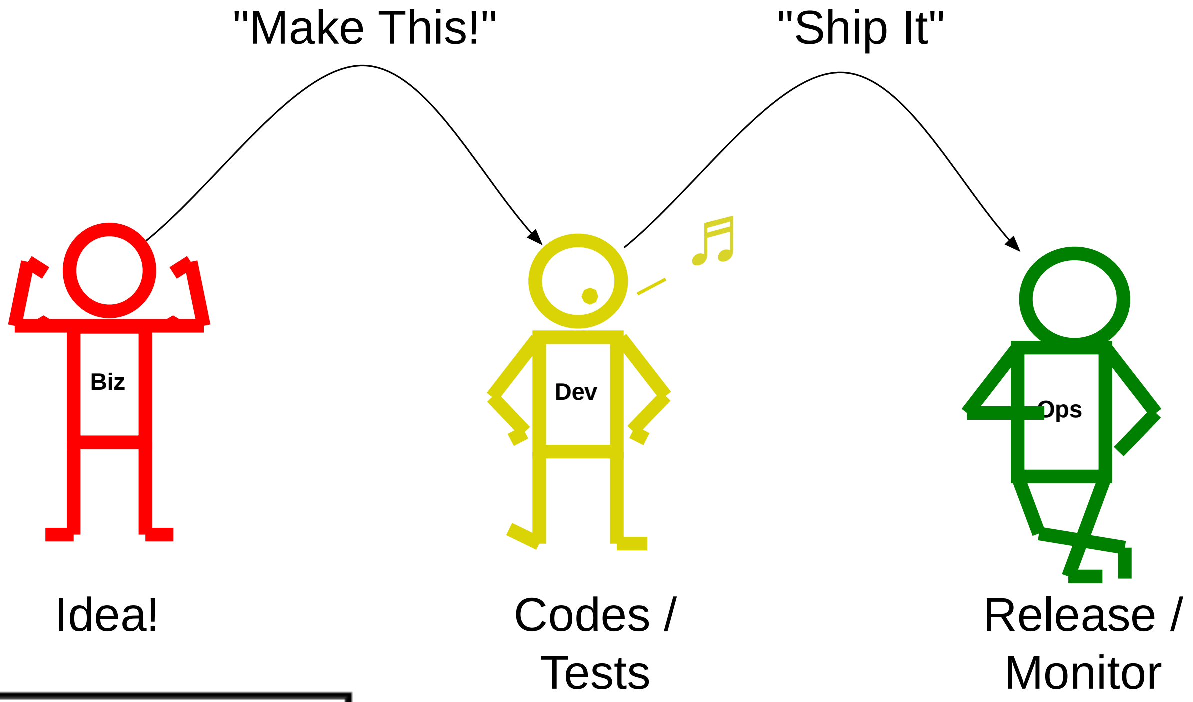
Idea!

Codes / Tests

Release / Monitor

This is broken

Software Delivery in a Nutshell



Are we making more money?

How's the performance?

Apps: More Important / More Complex

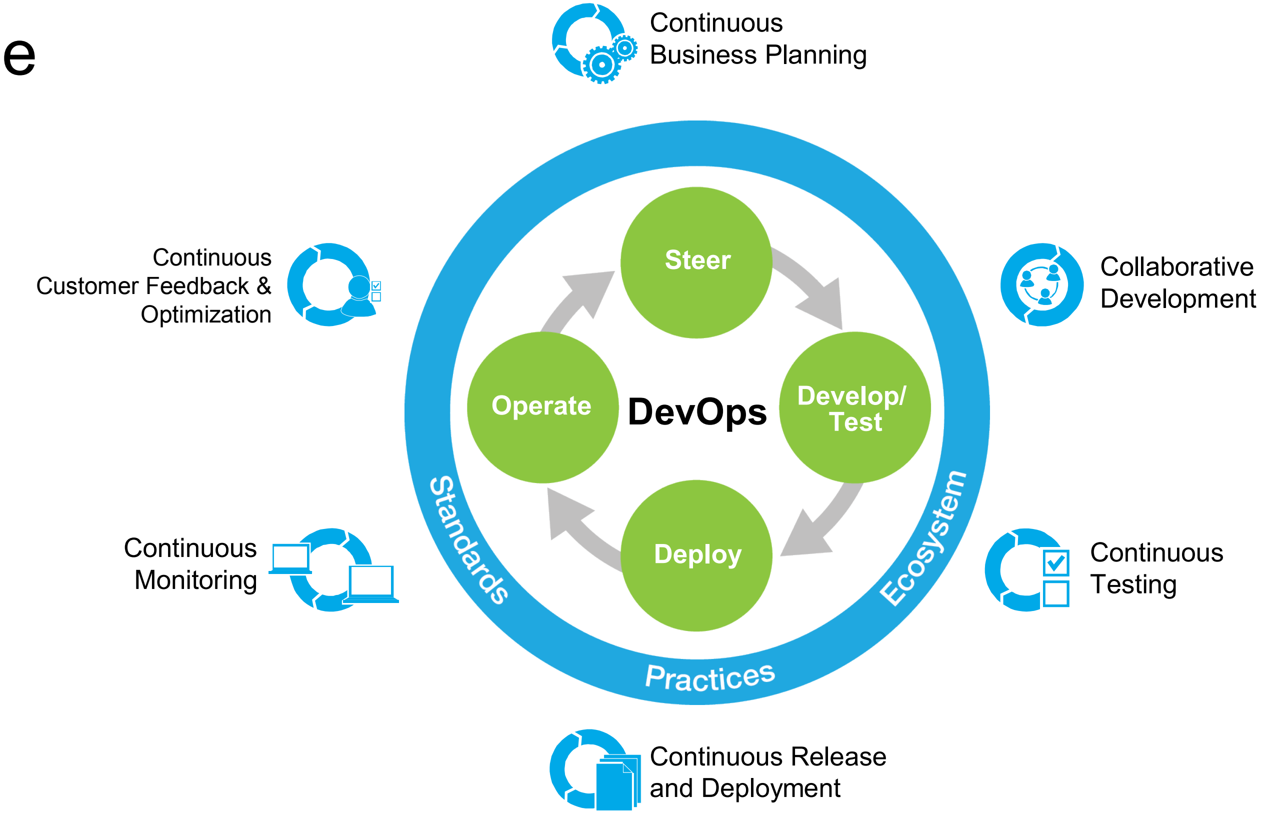
- More frequent changes
- Lower tolerance for outages
- More complex applications
- More complex deployments

- **Need:** Fix the Dev to Ops hand-off for speed / reliability, and deliver better visibility into production performance.

That's DevOps

Enterprise capability for continuous software delivery that enables organizations to seize market opportunities and reduce time to customer feedback

- ✓ Embrace Change
- ✓ Manage Risk
- ✓ Collaborate
- ✓ Automate


Continuous Business Planning
Continuous Customer Feedback & Optimization
Collaborative Development
Steer
Operate
DevOps
Develop/ Test
Deploy
Standards
Practices
Ecosystem
Continuous Monitoring
Continuous Testing
Continuous Release and Deployment

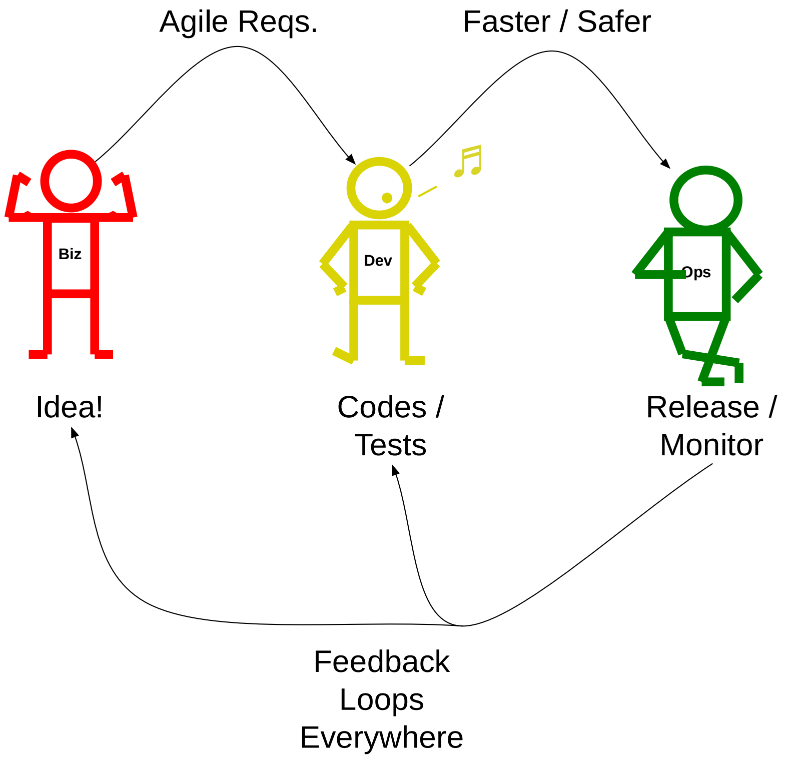
Themes in DevOps Tooling

Shared tools

Lots of automation

Self service *aaS

Version everything



Example impact on tools:
Issue Trackers

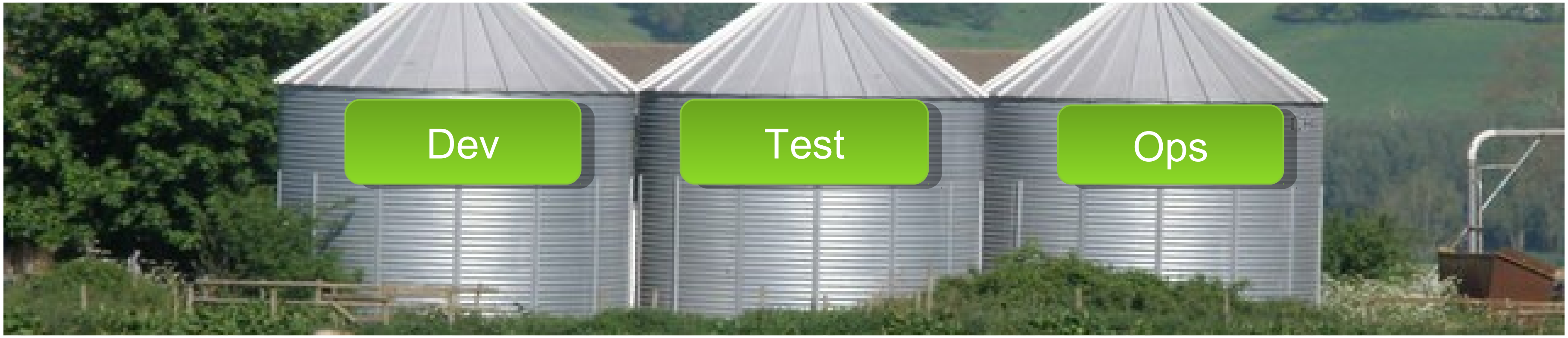
Example impact on tools: Issue Trackers

- **Shared Tools**: Within a project, have a common backlog for development and infrastructure tasks.

- **Automation:** Make sure you have APIs available to automatically create / migrate tickets based on tests, monitoring, deployments.

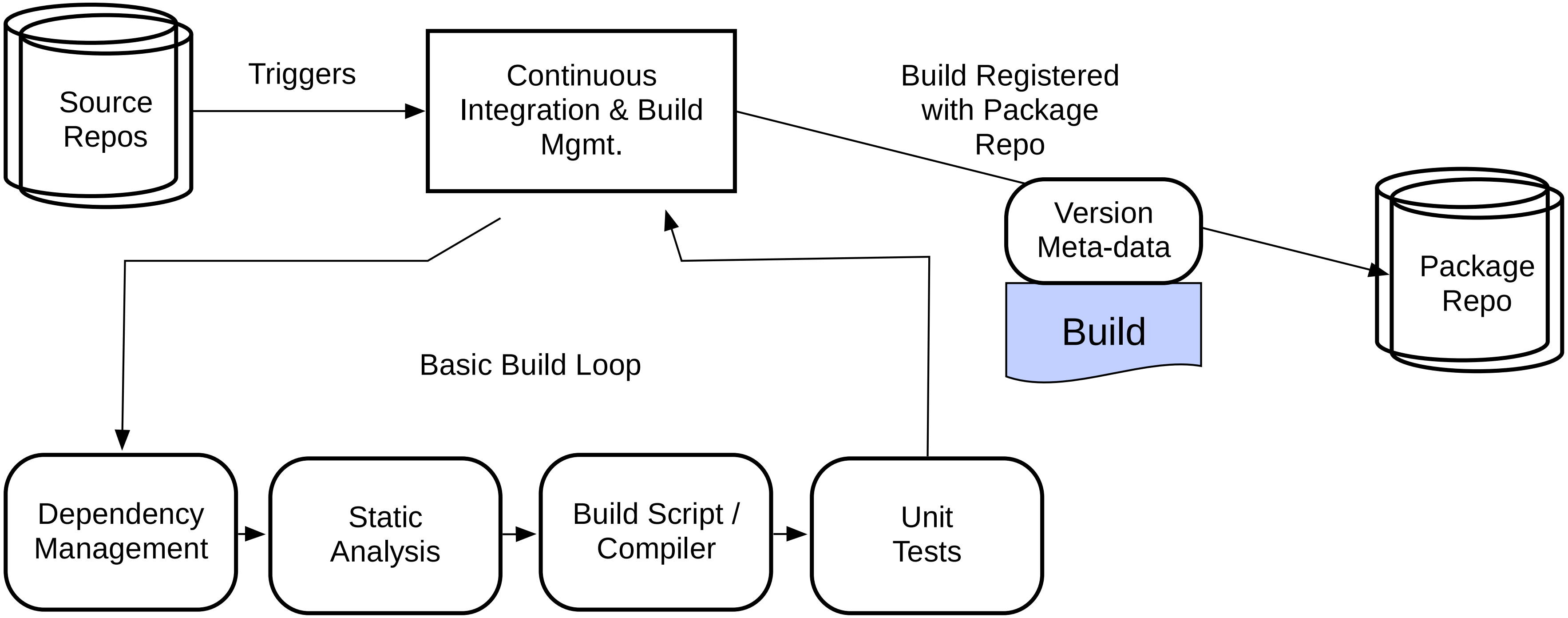
DevOps Toolchains

Span silos
Integrate a collection of a task tool-chains

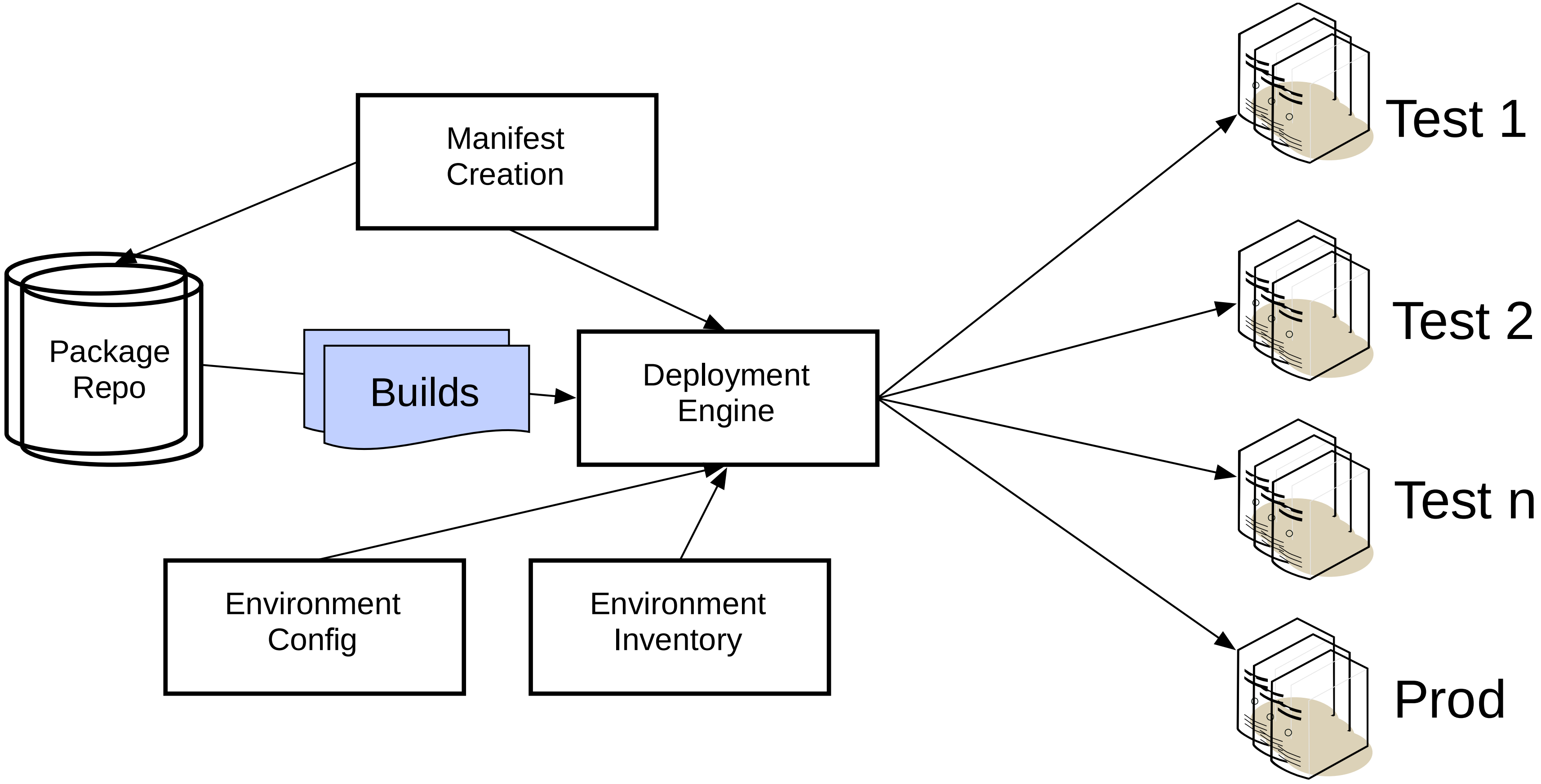


Source http://www.geograph.org.uk/profile/19979

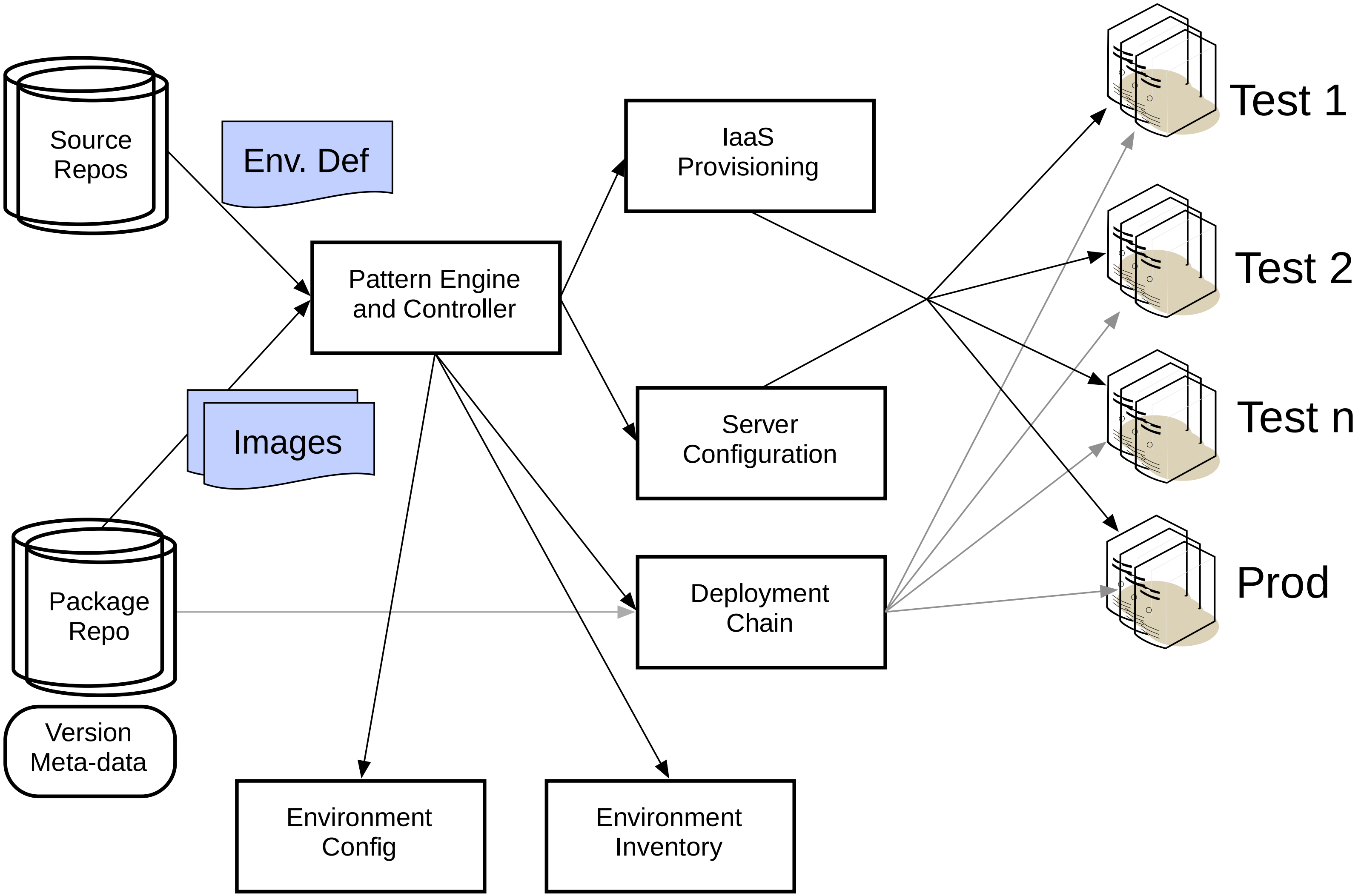
Toolchain 1: Build



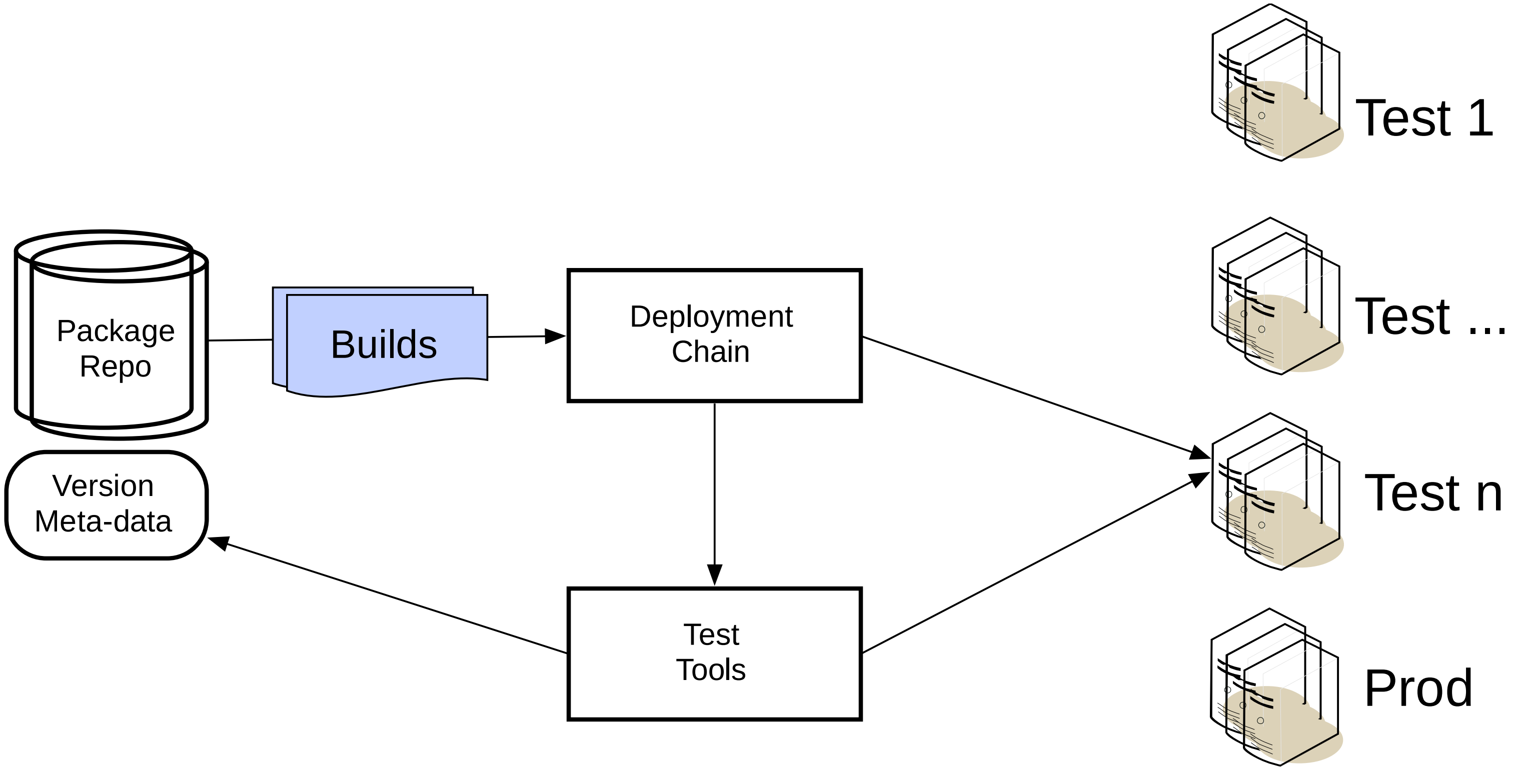
Toolchain 2: Deployment



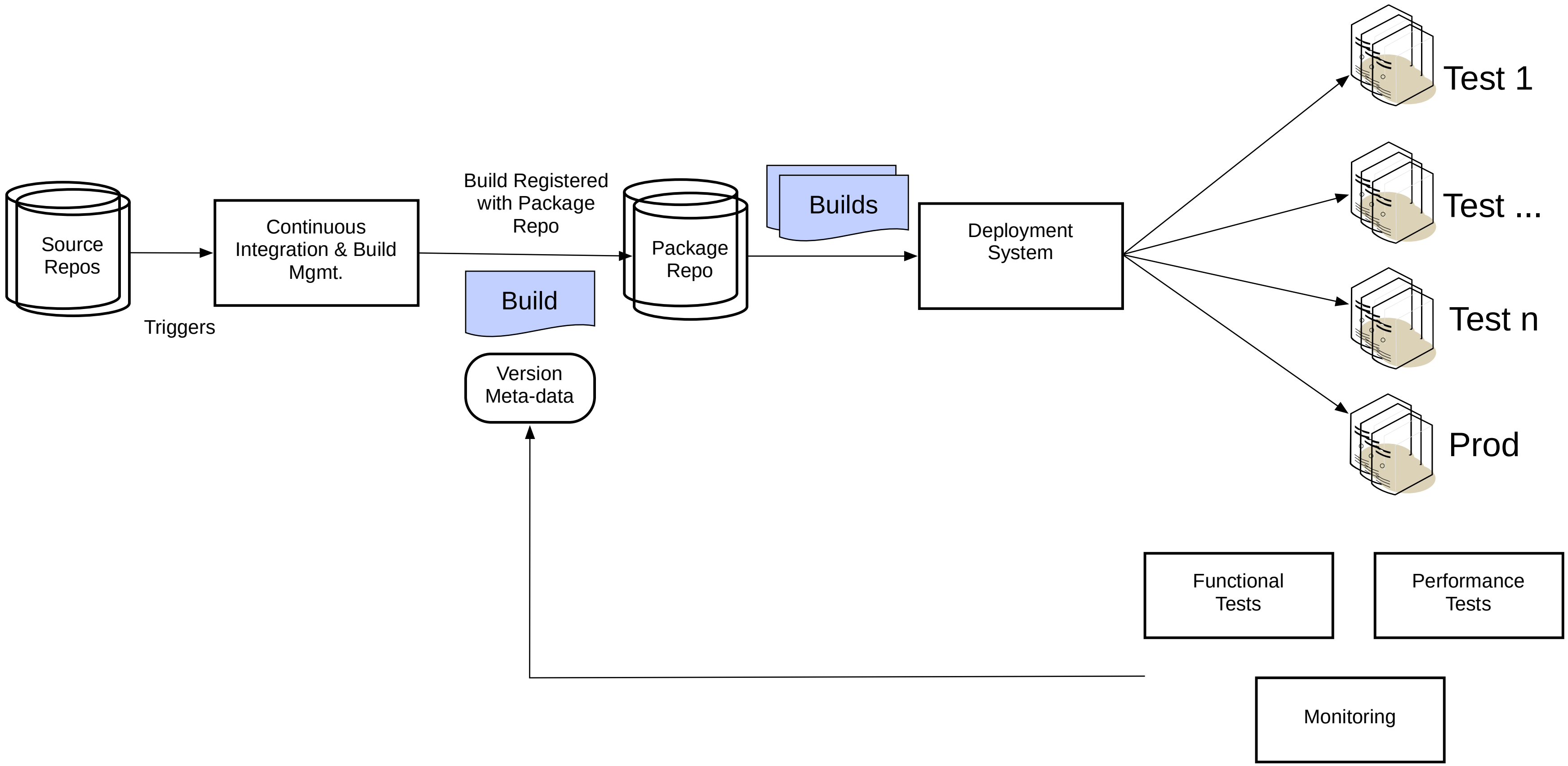
Toolchain 3: Platform as a Service



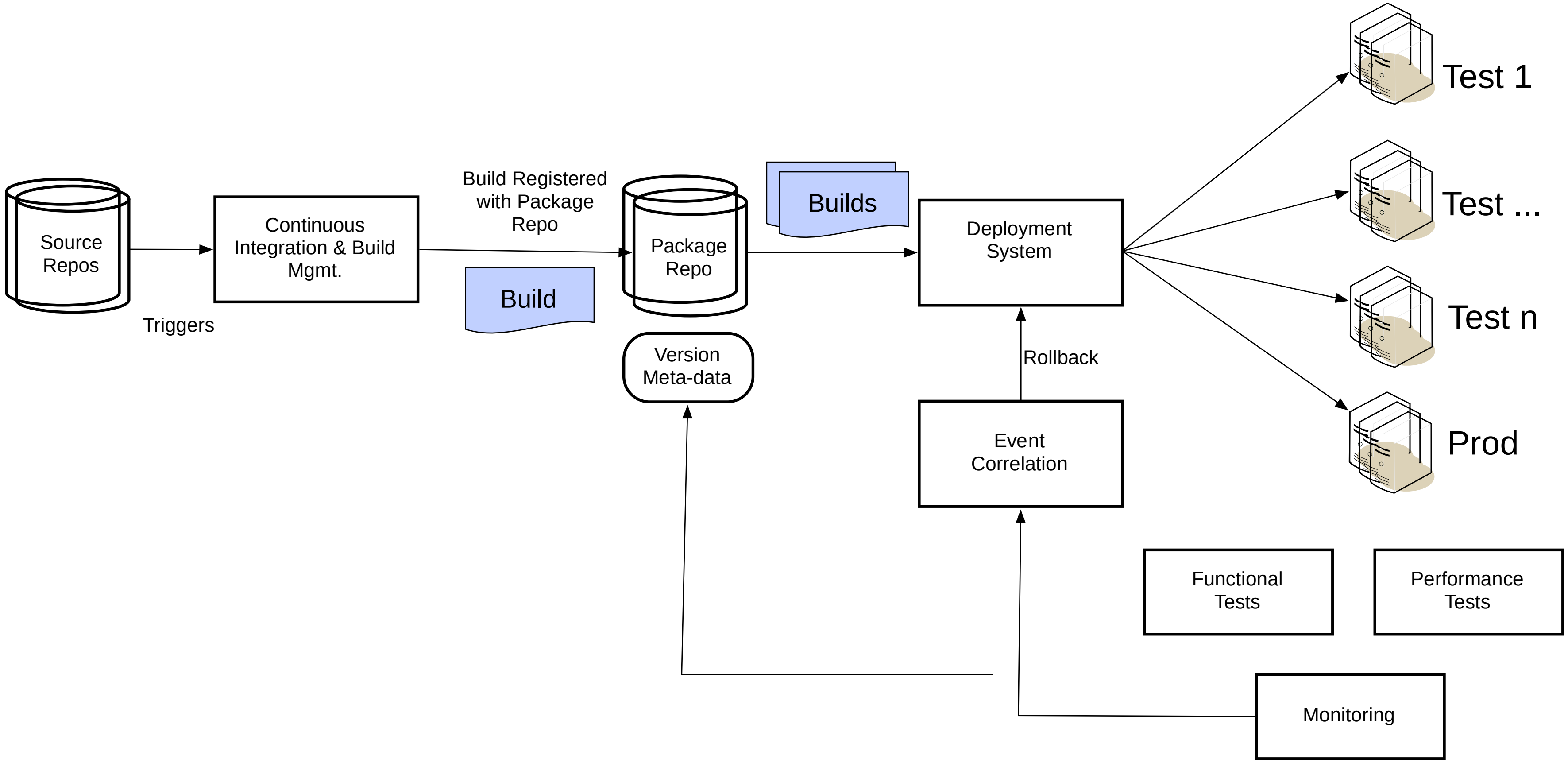
Toolchain 4: Test on Deployment



Toolchain 5: Continuous Delivery / Deploy



Toolchain 6 – CD with Rollback



Where does this get us?

Efficiency

Less manual effort

We avoid rework

Information is easier to find

Less dependence on tribal knowledge

Where does this get us?

Consistency

Tasks required to deliver tracked in one place

CI and “official builds” unified

Same deployment tool for all environments

Same infrastructure tool in all environments

Where does this get us?

Safety

Error prone manual processes minimized

Deployments of known versions

Where does this get us?

Visibility

Deployments of known versions

Inventory of what is where

Audit is easy

Where to start?

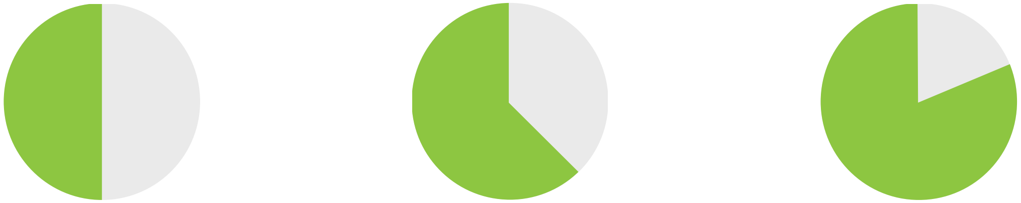
IBM DevOps adoption model

New practices, tools and services to plan and execute a staged adoption of DevOps to improve business outcomes

	Inefficient	Leaner	Leaner and Smarter
Steer	Process-based	Product-based	Optimizing
Develop/Test	Process-heavy	Agile	More Predictable
Deploy	Manual	Automated	More Transparent
Operate	Silo-ed	Collaborative	More Continuous

Efficiency

🟩 Productive ⬜ Waste



Feedback cycles

Audit what you have

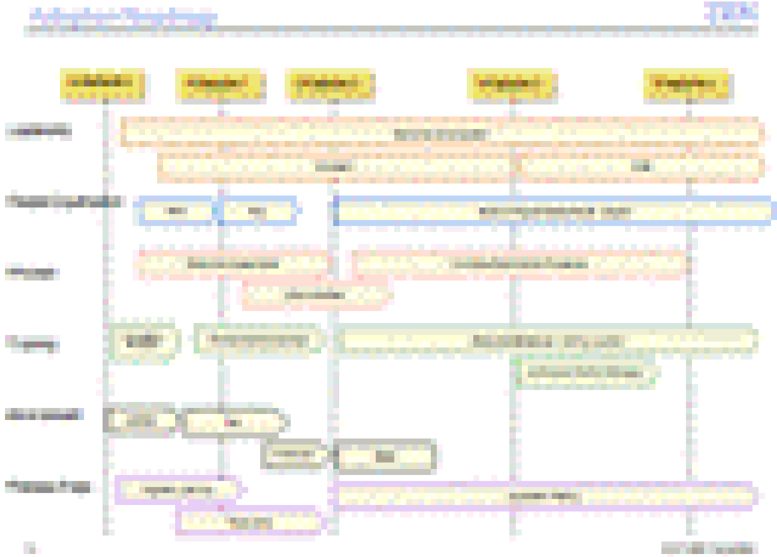
- Where are the bottlenecks?
- Where are errors introduced?
- Where is communication weak?

Use our assessment tool

https://ibm.biz/devops-practices-assessment

Agree on a Vision

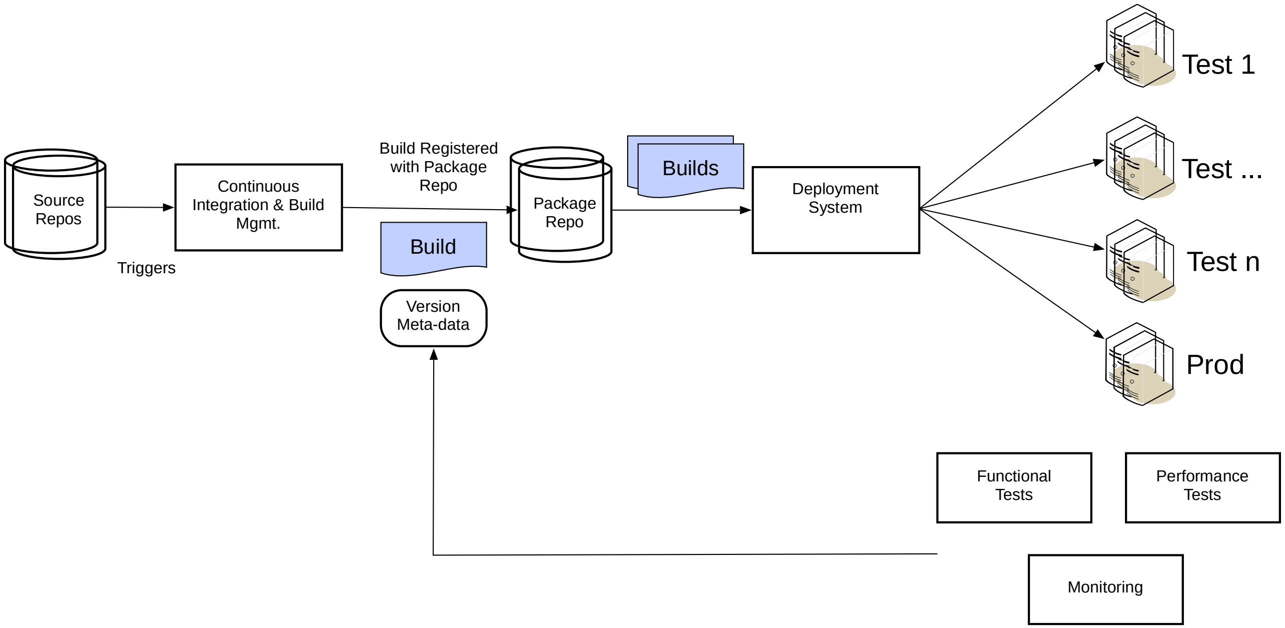
Roadmap path forward



Begin adopting tools

What's Normal? Left to Right, then Right to Left.

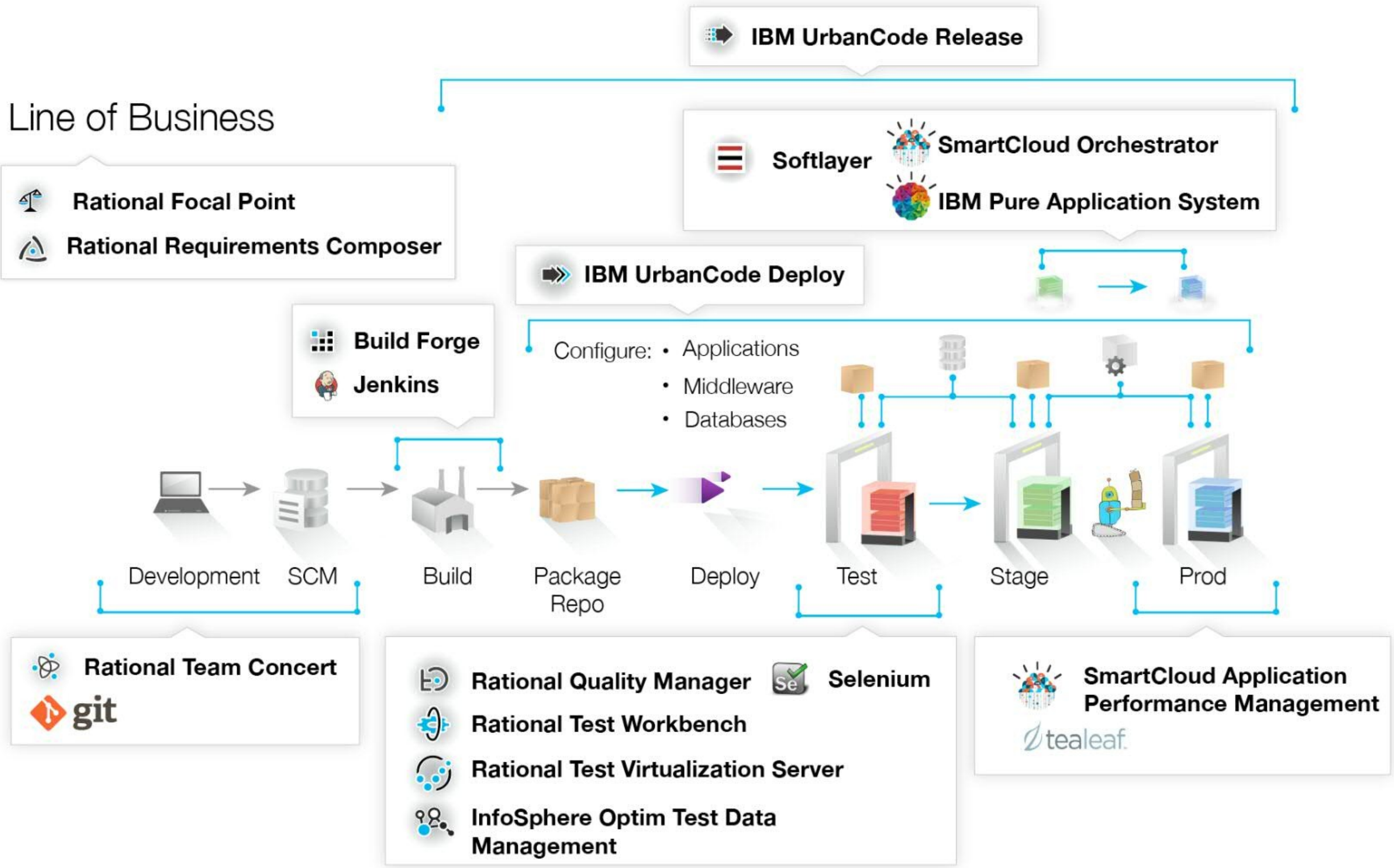
1. Ensure at least "ok" build
2. Work on the deployment / delivery tool chain
3. Attach feedback mechanisms opportunistically



Key Points

- Lots of tools
  - Build, Deploy, Provision, Config, Test, Monitoring
  - Multiple vendors: APIs & Direct Integrations
  - Expect and manage overlap
- Share tools across silos
- Automate Aggressively

An IBM heavy tool chain



IBM UrbanCode Release

Line of Business

Rational Focal Point

Rational Requirements Composer

Softlayer

SmartCloud Orchestrator

IBM Pure Application System

IBM UrbanCode Deploy

Build Forge

Jenkins

Configure:
- Applications
- Middleware
- Databases

Rational Team Concert

git

Rational Quality Manager

Selenium

Rational Test Workbench

Rational Test Virtualization Server

InfoSphere Optim Test Data Management

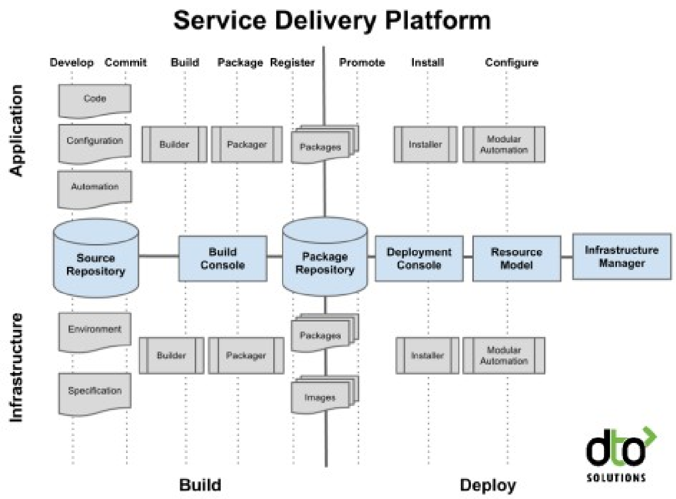
SmartCloud Application Performance Management

tealeaf

Other good DevOps tool chain resources

- Gartner:
  - “DevOps Toolchains Work to Deliver Integratable IT Process Management”

- Dev2Ops.org



http://dev2ops.org/blog/2012/7/23/integrating-devops-tools-into-a-service-delivery-platform-vi.html

UrbanCode References

http://developer.ibm.com/urbancode

- White Papers
  - –Enterprise CD Maturity Model
  - –Deployment Automation Basics

- Stay in touch

https://developer.ibm.com/urbancode/blog/

Twitter.com/UrbanCode Twitter.com/EricMinick

Slideshare.net/UrbanCode

Q&A

@UrbanCode
@EricMinick

Slideshare.net/urbancode
Developer.ibm.com/urbancode/