Stairway to Heaven: 10 Best Practices for Enterprise Continuous Delivery with Jenkins

Alex Manly
VP Product Development
@MidVision

June 18, 2014

#jenkinsconf

A bit about me…

- VP Product Development at MidVision
- Worked in Deployment Automation Solutions in regulated environments for 10 years.
- 15 years experience developing (and deploying) Java Enterprise Applications
- Have worked in both Dev **and** Ops….
- Oh yes…. that is a picture of me in a UK pub!

A bit about MidVision...

MidVision™
Release the innovation

- Application Release Automation Platform - deploying both environments and applications at an enterprise scale
- Born in a Bank – regulatory compliance is key
- Cross industry customer success - proven at scale in large complex regulated enterprises
- Core Platform – with ability to integrate to Open Source and Commercial tool chains
- Gartner DevOps Cool Vendor

Gartner 2014 CoolVendor

Successes

LLOYDS BANKING GROUP

RBS Citizens

Domestic & General

gsk

DWP Department for Work and Pensions

BR PETROBRAS

Agenda

A. Setting the Scene:
   What is and Why Continuous Delivery?

B. Stairway to Continuous Delivery Heaven :
   10 Best Practices for Enterprise Continuous Delivery with Jenkins

C. Making it real :
   Simple to Complex Deployment Scenarios

A. Setting the Scene: What is Continuous Delivery (CD)?

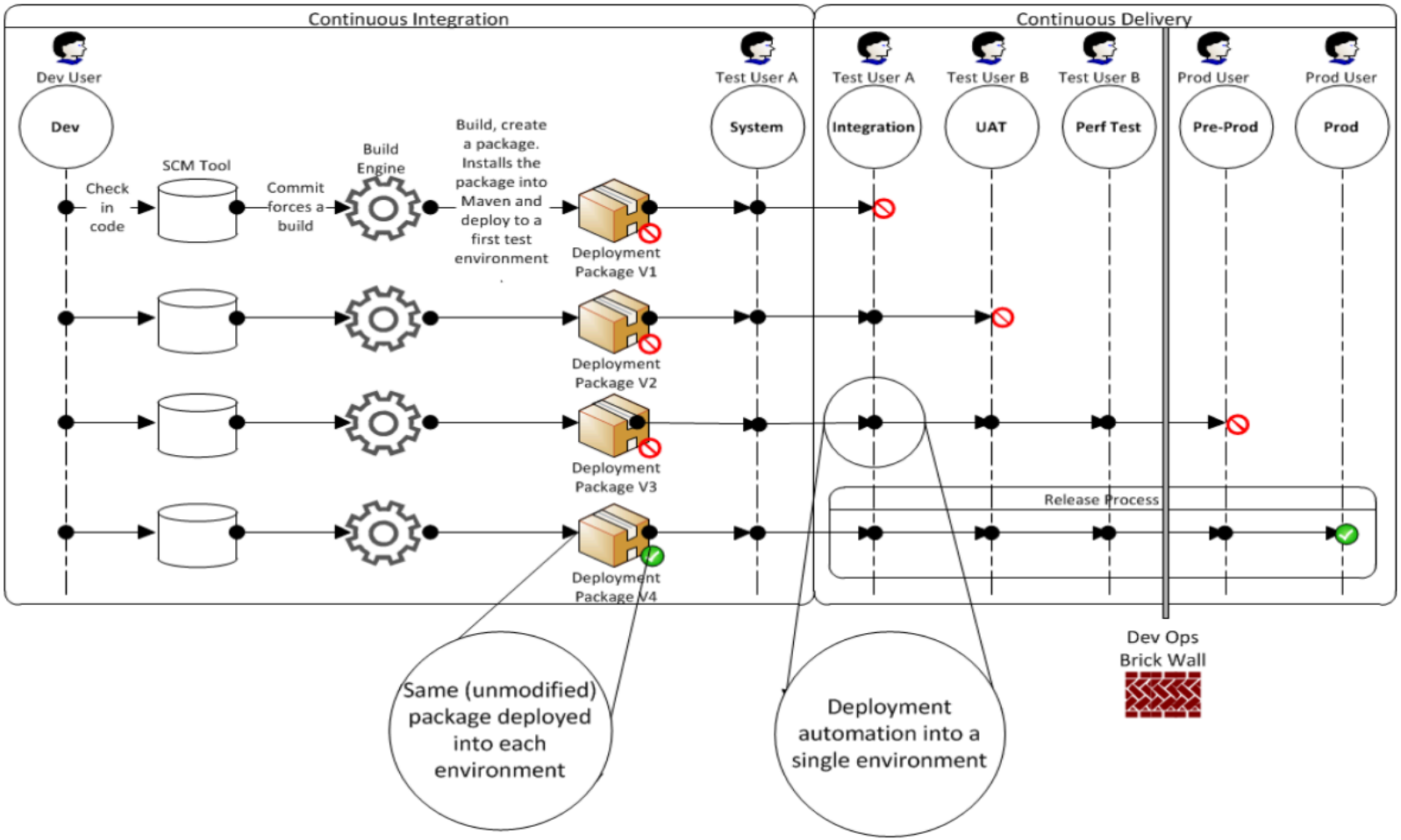
Continuous

- Development
- Integration
- Deployment
- Testing
- Release

- Complete application release process.
- Pipeline, chaining processes.

- Configuration as code.

A. Setting the Scene: What is Continuous Delivery?



A. Setting the Scene: Why Continuous Delivery?

- C-Suite agenda.
- Game changer.
- Release quality
- Speed up change.
- Reduce Costs.

B. Stairway to Continuous Delivery Heaven 10 Best Practices for Enterprise CD with Jenkins

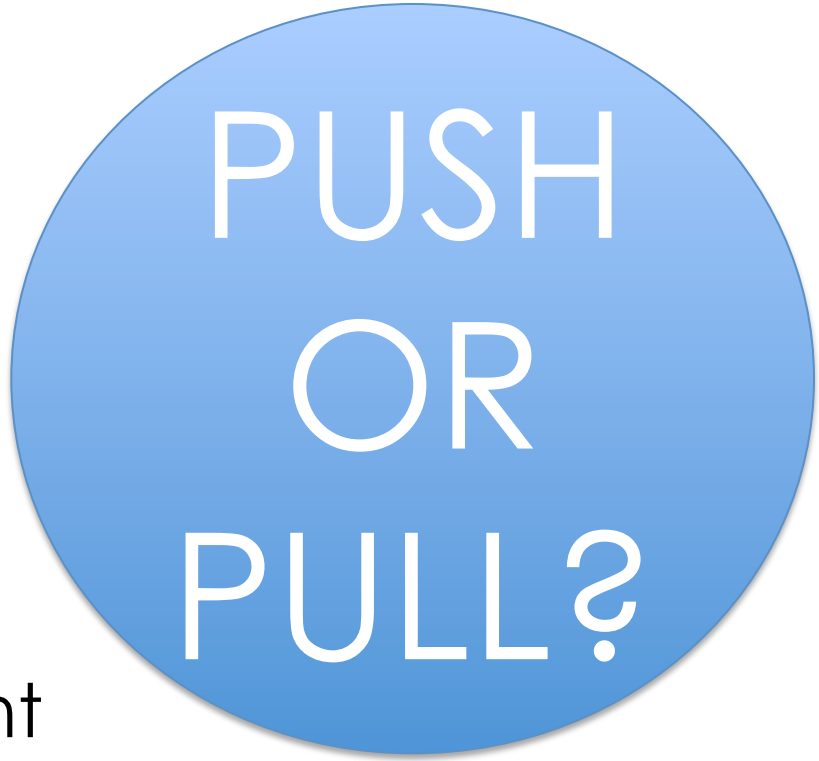
- Follow these steps to help you make the right decisions when implementing your CD approach and solution.
- Decide on the right tools for the task you are trying to automate.
- There is no “one size fits all” solution to Continuous Delivery.

Step 1 – Continuous Delivery Method Manual vs. Automation?

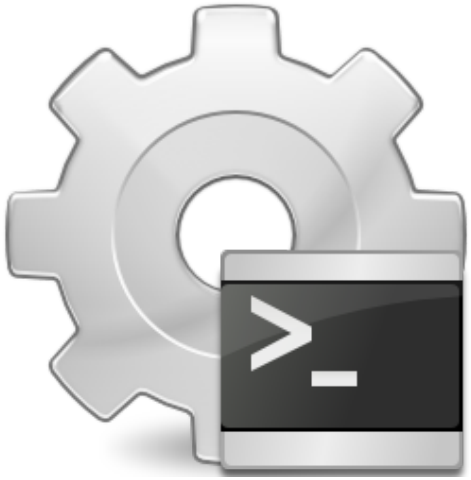
- Choose the low hanging fruit.
- Cost of Automation.
- Cost of **NOT** automating.
- Use a tool like Jenkins to interconnect jobs.

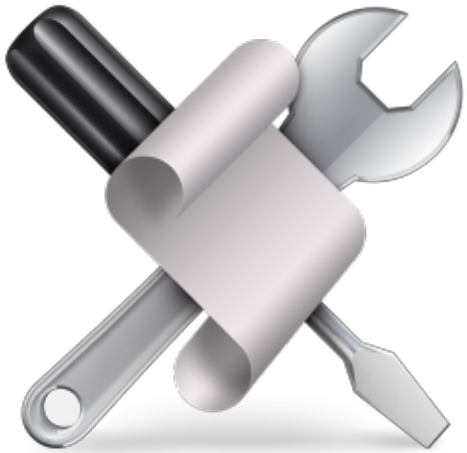
Step 2 – Deployment Paradigm
Directed, Convergent or Both

- Directive models
- Convergent models
- Homogenous or heterogeneous.
- Choose the deployment tool wisely.



Step 3 – Determine the Deployment Type API vs. File



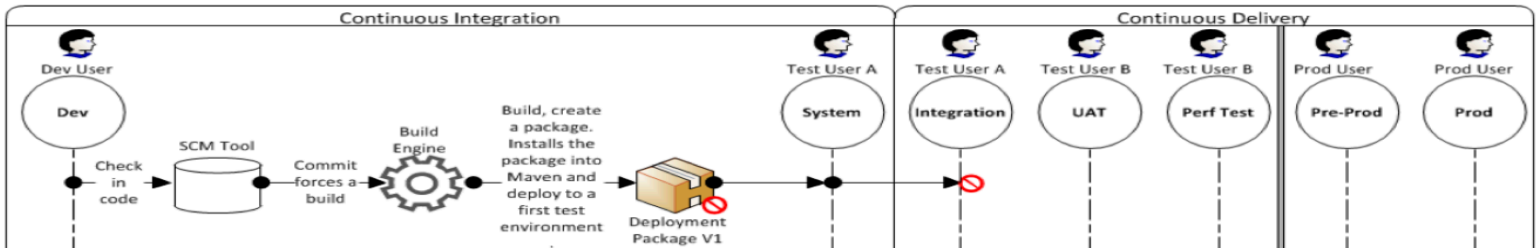


- What is the Deployment type?
- Many complex middleware systems require configuration to be managed by its own API.
- Simpler systems can be configured and deployed to by managing a set of files.

Step 4 – Packaging Principle
Build Once, Deploy Anywhere

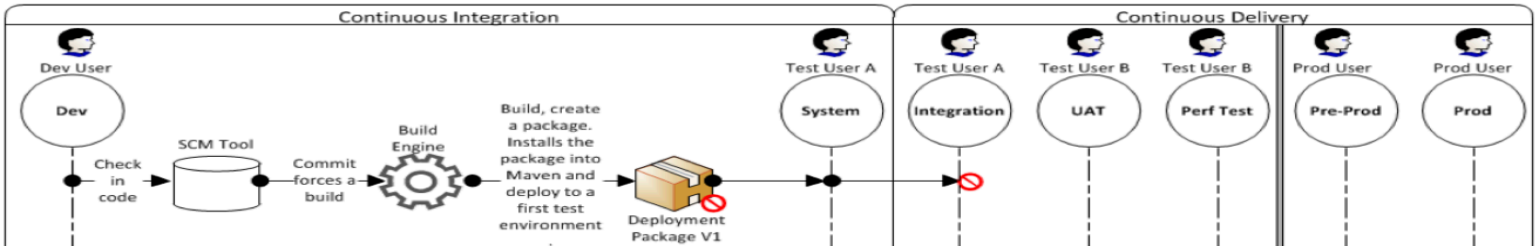
- Build once, deploy to any (defined) environment in the pipeline.

- A package should be a single compressed, **versioned** file.
  - Package **integrity** across all environments
  - Check-summed and labeled



Step 4 – Packaging Principle
Build Once, Deploy Anywhere

- Deployment model / Deployment instructions
- Deployment resources
- Can deploy as a stand-alone process.
- Store and retrieve deployment artifacts in a secured Definitive Software Library (DSL).
- Use Jenkins and to build your deployment package and store it in the DSL.



Step 5 – Picking the right Packaging Model for Deployment Artifacts

- Topology of different environments can define the packaging model to use.
- Single App / Single Cluster
  - Deploy application and configuration together

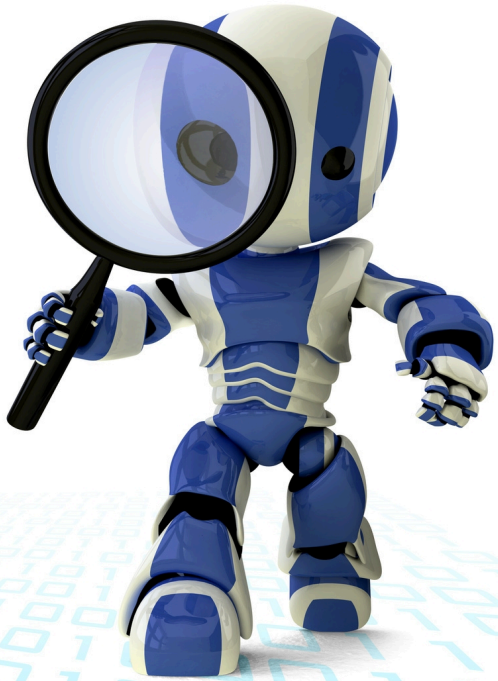
- Multiple App / Multiple Cluster
  - Deploy applications and configurations separately

Step 6 - Prerequisite Resource & Environment Testing

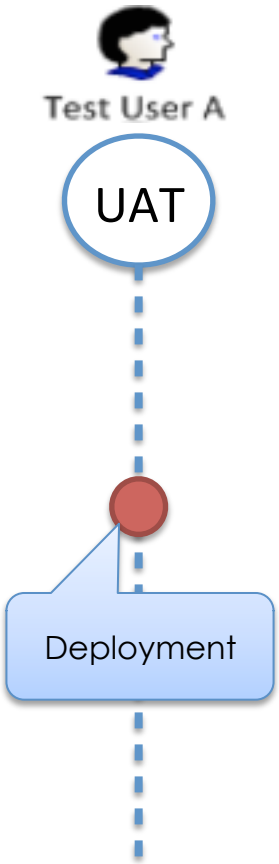
- Test deployment resources.
- Do **not** continue.
- Saves **a lot** of pain and time.
- **Keep adding.**
- Sometimes known as a "Deployment Dry Run".

Step 7 – Automated Testing Strategy

- Use Jenkins **post build plugins** or **chained jobs** to initiate the tests after successful a deployment.
- Use parameterized matrix (multi-configuration) jobs to test different versions of your application in one run. E.g. versions of **architecture** [32, 64], **db** [oracle, mysql, db2], **jdk** [1.6, 1.7, 1.8], **browser** [firefox, chrome, safari].
- Sanity check your builds with touchstone builds.
- Building pipelines can be quick, converting your manual tests to automated tests can be laborious and take months…..stick with it.

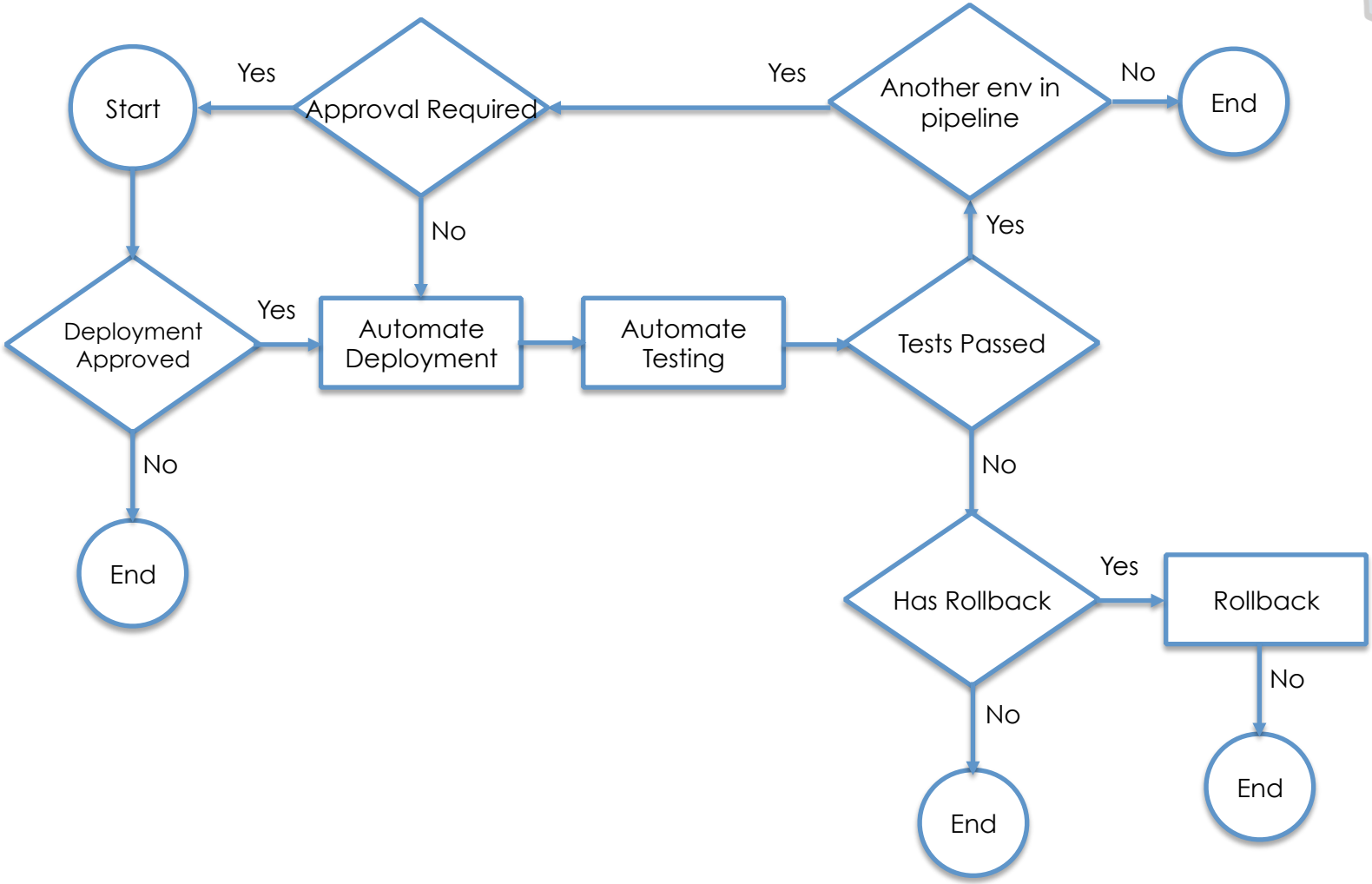


Step 8 - Approvals, Compliance and Audit Strategy



- A continuous delivery process in an enterprise often **requires approvals.**
- Allow for **one or more groups of people** to approve.
- An audit trail of approval requests, decisions and deployments is **mandatory for compliance** in regulated environments.
- Approval can be for a deployment or even a specific environment configuration change.

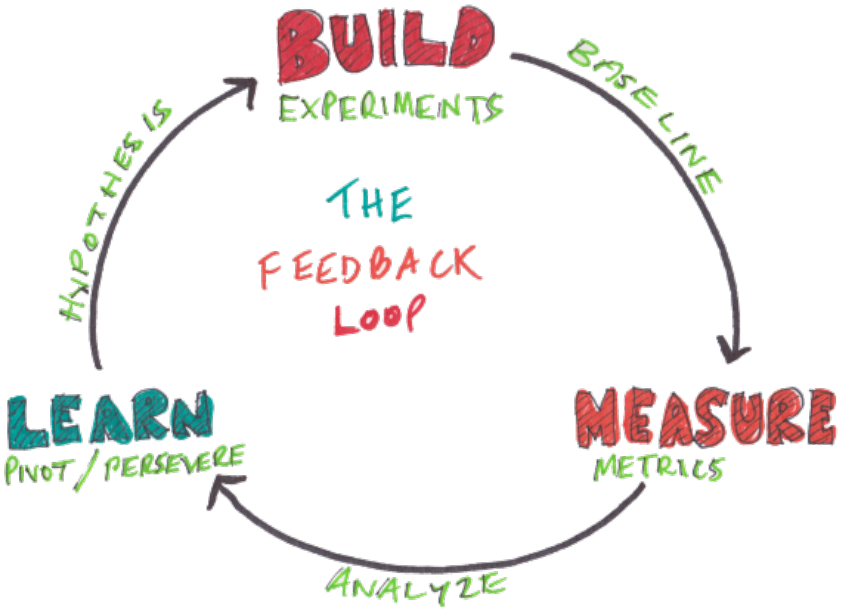
Example Deployment Approval Process



Step 9 – Rollback Strategy

- Depends on the technology.
- Database changes cannot be rolled back automatically if business data has been added or amended since the deployment took place. **Needs a DBA**.
- Make sure DB changes are **backwards compatible**.
- If DBAs allow it, use a DB tool to manage the state of application DB scripts for migrating applications.
- Generally with databases, use a **fix forward** strategy.
- Middleware can be rolled back if the code is deployed with the configuration.

Step 10 - Metrics, Analytics & Feedback Loop



- Measure the process.
- Report the process
- Prove the positive impact of CD.
- Build sponsorship and business case to invest in CD.
- Make use of the CI and CD metrics from Jenkins to support the evidence for your business case.

Stairway to Continuous Delivery Heaven : 10 Best Practices to help you along your journey

1. CD Method - Manual vs. Automated
2. Deployment Paradigm - Directed or Convergent or both
3. Deployment Type - API vs. File
4. Packaging Principle - Build Once, Deploy Anywhere
5. Packaging Model for Deployment Artifacts
6. Pre-Requisite Resource & Environment Testing
7. Automated Testing Strategy
8. Rollback Strategy
9. Approvals, Compliance and Audit Strategy
10. Metrics, Analytics & Feedback Loop

C. Making it real : Simple to Complex Deployments Scenarios

- For example, WebSphere Application Server Cell is multi-node.
- Complex binary installation process - not just unpack.
  - IBM Installation Manager
  - Cell set-up and configuration
  - Security
  - Federation
- Thousands of configuration items to configure and manage.
- Small configuration changes can have a **BIG** effect (increasing a cluster count on each node).

Simple Deployment Scenario - Deploy an EAR file to a Cluster

- Even in an apparently easy and straightforward scenario there are complexities to be aware of.
  - Resource mappings / bindings
  - Search / Replace settings inside the EAR file….developer hacks!!!!
  - Application specific configuration - Classloading policies, start-up order, context root, etc.

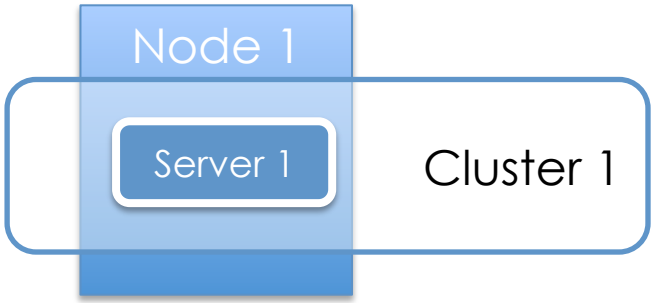
Complex Deployment Scenario - Manage Cell Resources

- Clusters - Ports, Heap sizes, etc
- Servers
- JDBC
- URL
- JMS
- Security
- System Integration Bus
- Shared Libraries
- and so on…

Simple Deployment Topology – Development

- Development Topology
  - 1 Cluster, 1 Server, 1 Node
  - 100s of configuration items
  - Size of WebSphere configuration XML is X

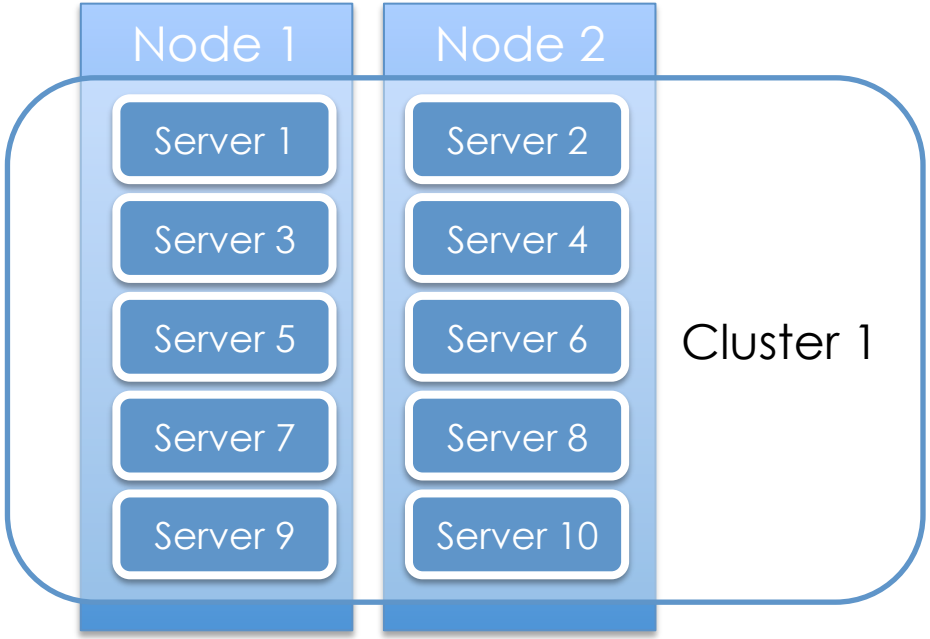
Development Topology



Complex Deployment Topology – Production

- Production Topology
  - 1 Cluster, 10 Servers, 2 Nodes
  - 10 x 100s of configuration items
  - Size of WebSphere configuration XML is 10X

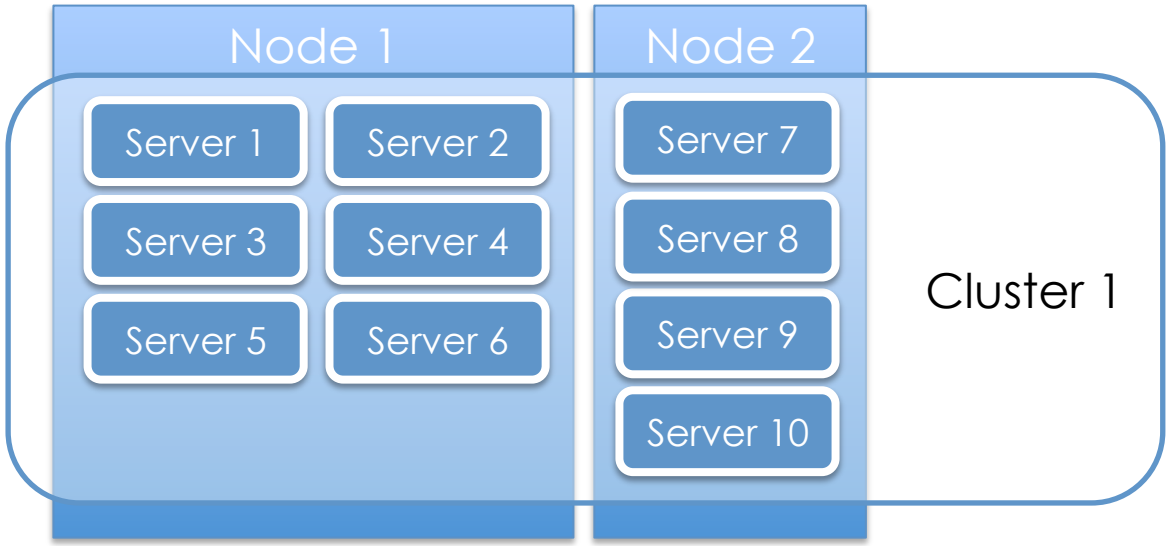
Production Topology



Example Scenario - Extended

- Slightly more complex example
- 6 members on 1 node and 4 on another.
- Need to direct the cluster members to specific nodes

Production Topology



Simplify the Deployment Configuration

- Use a tool to snapshot and copy the XML right?
- XML snapshot is **totally different** between environments.
- Only difference between **dev** and **prod** is the cluster member count.
  - Dev: cluster member count = 1
  - Prod: cluster member count = 10
- Only 1 configuration item needs to change between the environment definitions…Not 1000s.

Deployment Considerations

- What about managing the Ports of all the cluster members on each Node so there are no port conflicts?
- Port should be managed by convention.

- Why?
  - You have a lot of ports to manage.
  - Determine port type from address.
  - e.g. XXX050 is always the SOAP port.

Key Takeaways

- CD is……Continuous.
- Leverage the 10 steps.
- Start small.
- Evaluate and choose the right tools.
- Automating isn't easy, simplify it.
- Enjoy the positive impact of CD.
- Have fun along your journey….

Thank you for your time

Please visit us at our MidVision Stand @ JUC if you want to speak to us today

Please visit our Website for more information about what we do

http://www.midvision.com

If you would like to contact us following the JUC

info@midvision.com

You can try out our RapidDeploy Platform (Community Edition) for FREE

http://www.midvision.com/products/rapiddeploy-community-edition

We have already contributed with our Jenkins RapidDeploy plugin

https://wiki.jenkins-ci.org/display/JENKINS/RapidDeploy+plugin

Thank You To Our Sponsors

Platinum

CloudBees

Gold

black diamond SOFTWARE

MidVision™ Release the innovation

choose digital

JFrog

XebiaLabs the deployment automation company

Silver

SOASTA Test Faster. Release Sooner.

ELIASSEN GROUP Agile Consulting Services

KENSHO

SERENA™

assembla