



# Building Your Continuous Delivery Toolkit

**Andrew Phillips**

XebiaLabs

[xebialabs.com](http://xebialabs.com) | [@xebialabs](https://twitter.com/xebialabs)

October 23, 2014

*With thanks to TJ Randall*

#jenkinsconf

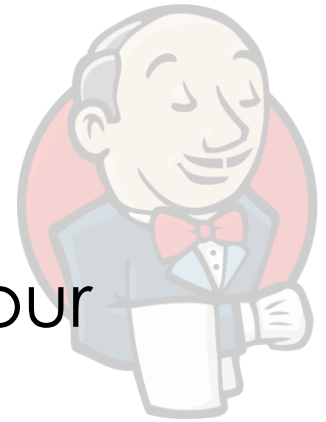
## You're talking about *other* tools at a JUC??



- It's OK to think you may need additional tooling to solve your issues rather than trying to use one tool for everything that you need to do!
- If you've expanded Jenkins into this area you'll see that there are indeed a bunch of options...after all, you can do *anything* with Jenkins!
- You're hitting some challenges:
  - Maybe you've already put this together, but you're having challenges maintaining it.
  - Or, maybe you need your solution to do more, and are finding that you're getting quite far away from the sweet spot.

# Jenkins rocks!

We love Jenkins ourselves and use it for our Continuous Integration:



**Jenkins**

[Jenkins](#) [Plugin Manager](#)

[Back to Dashboard](#) [Manage Jenkins](#)

Updates Available **Installed** Advanced

Enabled	Name ↓	Version	Previously installed version	Pinned	Uninstall
<input checked="" type="checkbox"/>	<a href="#">Mailer</a>	<a href="#">1.5</a>	<a href="#">Downgrade to 1.4</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">External Monitor Job Type Plugin</a>	<a href="#">1.2</a>	<a href="#">Downgrade to 1.1</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">LDAP Plugin</a>	<a href="#">1.6</a>	<a href="#">Downgrade to 1.4</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">PAM Authentication plugin</a>	<a href="#">1.1</a>	<a href="#">Downgrade to 1.0</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">Credentials Plugin</a> This plugin allows you to store credentials in Jenkins.	<a href="#">1.9.3</a>	<a href="#">Downgrade to 1.9.3</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">Ant Plugin</a>	<a href="#">1.2</a>	<a href="#">Downgrade to 1.1</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">Javadoc Plugin</a>	<a href="#">1.1</a>	<a href="#">Downgrade to 1.0</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">SSH Credentials Plugin</a>	<a href="#">1.6</a>	<a href="#">Downgrade to 1.6</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">Groovy plugin</a> This plugin executes Groovy code.	<a href="#">1.14</a>			<a href="#">Uninstall</a>
<input checked="" type="checkbox"/>	<a href="#">Build Monitor Plugin</a> Provides a highly visible view of the status of selected Jenkins jobs. It easily accommodates different computer screen sizes and is ideal as an Extreme Feedback Device to be displayed on a screen on your office wall.	<a href="#">1.3+build.68</a>	<a href="#">Downgrade to 1.2+build.64</a>		<a href="#">Uninstall</a>
<input checked="" type="checkbox"/>	<a href="#">Jenkins Port Allocator Plug-in</a> This plugin allocates free ports as environment variables.	<a href="#">1.8</a>	<a href="#">Downgrade to 1.5</a>		<a href="#">Uninstall</a>
<input checked="" type="checkbox"/>	<a href="#">Jenkins jQuery plugin</a> This allows other plugins to use jQuery in UI.	<a href="#">1.7.2-1</a>	<a href="#">Downgrade to 1.7.2-1</a>		<a href="#">Uninstall</a>
<input checked="" type="checkbox"/>	<a href="#">Subversion Plugin</a>	<a href="#">1.54</a>	<a href="#">Downgrade to 1.50</a>	<a href="#">Unpin</a> <a href="#">?</a>	
<input checked="" type="checkbox"/>	<a href="#">Maven Project Plugin</a>	<a href="#">1.509.4</a>			
	<a href="#">Repeating Builds Plugin</a>				

# Jenkins rocks!

Some of our favourite plugins...

- **Build Failure Analyzer**
  - Helps us to understand why builds are failing, especially when analyzing possible infrastructure issues.
- **Matrix Reloaded Plugin**
  - Allows users to easily rebuild parts of an already built Matrix build.
- **Promoted Builds Plugin**
  - Distinguish good builds from bad builds by introducing the notion of 'promotion'.
- **Throttle Concurrent Builds Plugin**
  - For throttling the number of concurrent builds of a project running per node or globally.



# Beyond Jenkins..?



## When do you hit the edges of your CI tool?

- Challenges can be conceptual, technical or organizational

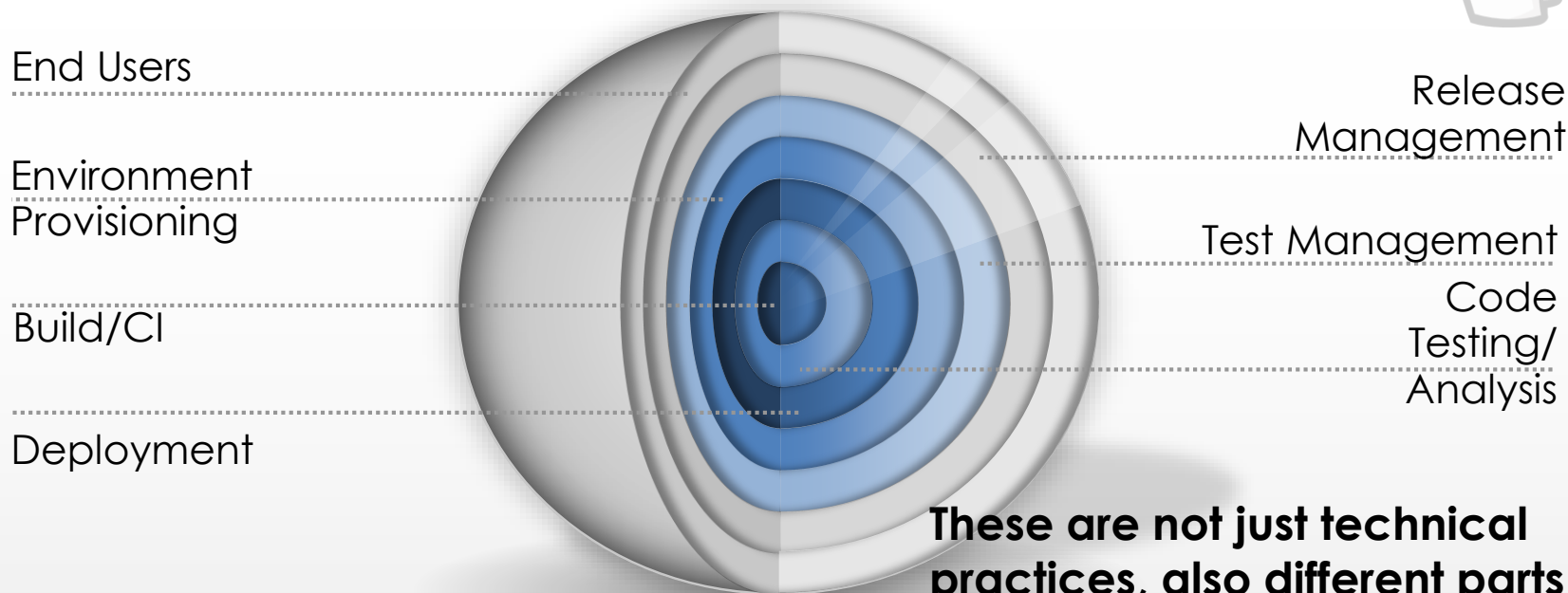
## Jenkins has a luxury problem based on its success

- Many, many plugins for many different types of tasks..."anything is possible"

# The “Continuous Delivery Onion”



The onion is bigger than Jenkins



**These are not just technical practices, also different parts of the organization**

- Audience needs information presented in an appropriate way

# Outline for a Continuous Delivery Toolkit



Before we talk about tools...

- CD is a *means*, not a *goal*!
- So there will be different needs depending on what you are looking to achieve
- We'll talk about what we see frequently...
- ...but it's not intended to be taken as a silver bullet!

# Outline for a Continuous Delivery Toolkit



Let's take a look at five technical needs for Continuous Delivery:

- Build/Continuous Integration
- Environment Provisioning
- Deployment
- Test Management
- Release Management/Pipeline Orchestration



# Outline for a Continuous Delivery Toolkit

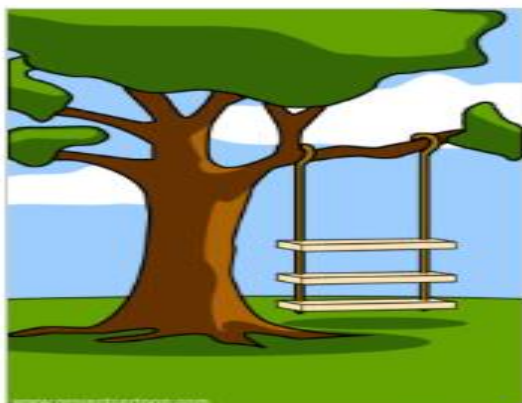


Let's not forget...these are different *teams*, too:

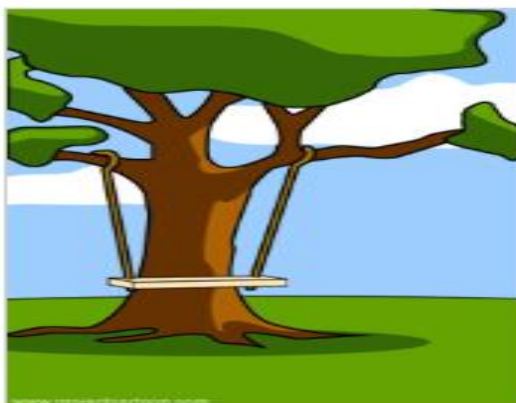
- Build/Continuous Integration: **Dev**
- Environment Provisioning: **Ops**
- Deployment: **Release/Ops**
- Test Management: **QA**
- Release Management/Pipeline Orchestration: **Release/DBAs/Ops/Business**

# Discovering the Elephant

Each team will have a different idea of CD!



How the customer explained it



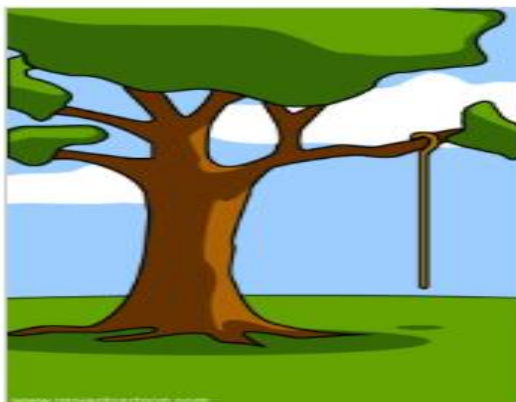
How the project leader understood it



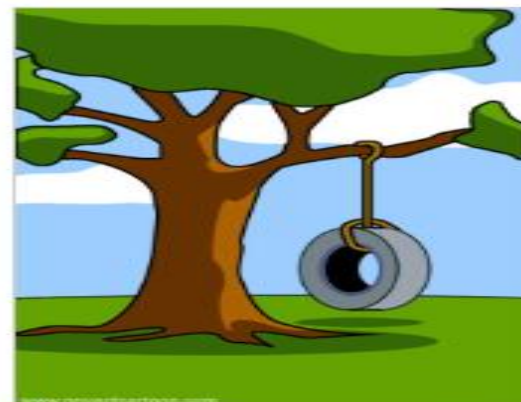
How the analyst designed it



How the programmer wrote it



What operations installed



What the customer really needed

## Example: CI



Let's not forget...these are different *teams*, too:

- Dev: “It’s that server that we run that does builds”
- QA: “What’s a CI tool?”
- DBAs: “Database changes in source control??”
- Release: “What’s a CI tool?”
- Ops: “What’s a CI tool?”

# Continuous Integration

- Scalability!
- Handling different types of builds
- Access control and visibility to the right groups



# Environment Provisioning



- This is not so much about scaling your CD infrastructure itself...
- ...but about creating target environments in which to run your applications
- Need to define and instantiate environments
- Also need to *register* these environments with your other tools

# Deployment



- How to define the entire application, including configuration?
- How to orchestrate multiple components across multiple machines?
- Any way to avoid duplication in scripts?
- Plus handling “standard” deployment problems such as environment-specific values, access control etc.

# Test Management



- Current trend: more and more tests in more and more different test tools
- How to combine all the output to answer the “go/no-go” question with confidence?
- How to choose the right tests for the context and manage your test suite over time?

# Pipeline Orchestration



- How to get the pipeline to go to production?
- How to handle variation and manual tasks in your process?
- How to provide the right level of insight to the right groups?
- How to start with the process you have today and improve over time?

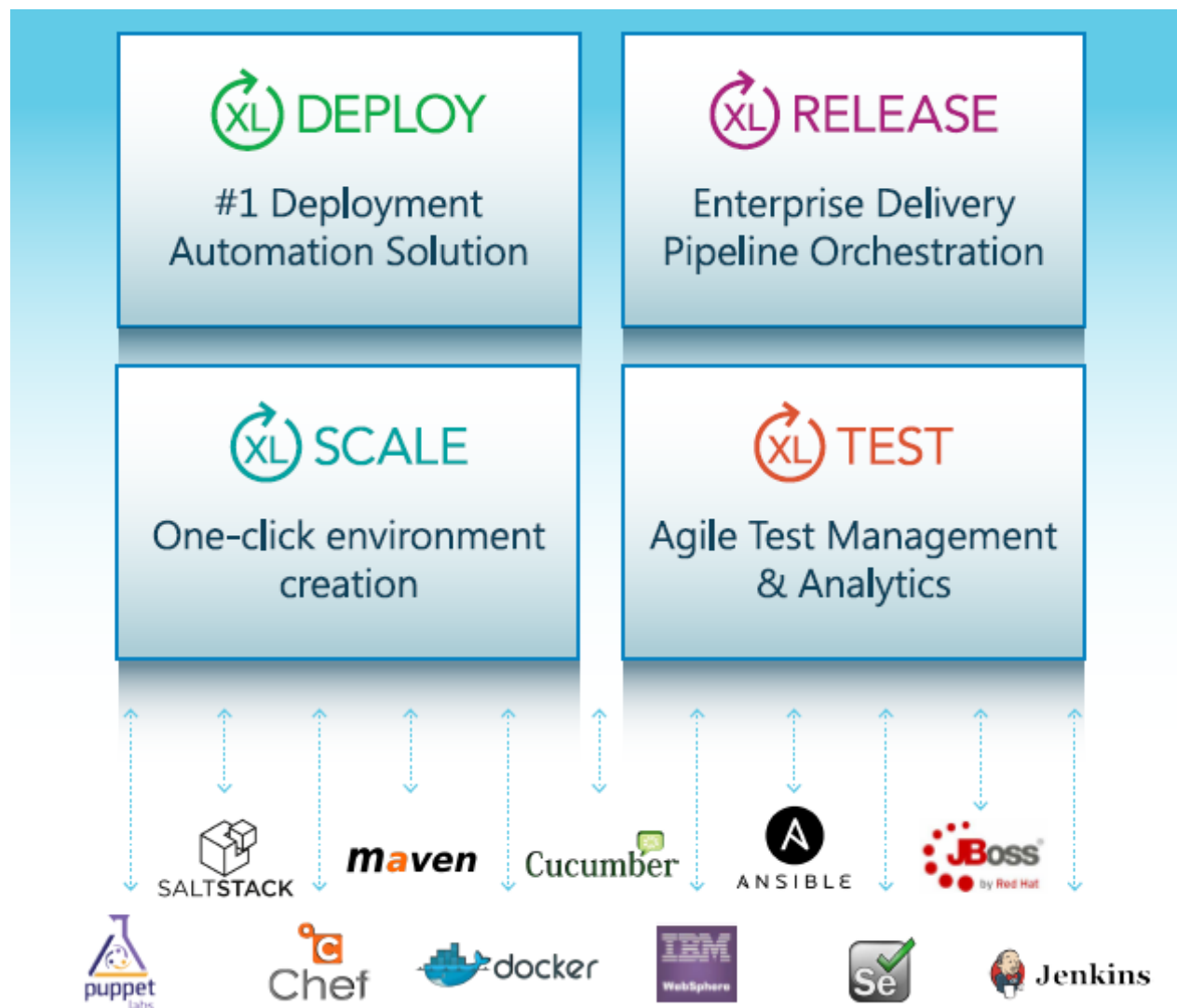


# How do you Introduce a Toolkit?



- How do you involve all of your teams in your Continuous Delivery tool set? How did your automation tools get chosen?
- Most organizations try to **limit/avoid** cross-team activities when choosing a tool.

.....**PLUG!**



# Thank You To Our Sponsors

## Platinum



## Gold



## Silver



## Corporate

