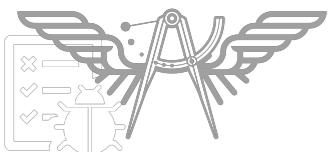


Microservices workshop





Somkiat Puisungnoen

Somkiat Puisungnoen

Update Info 1

View Activity Log 10+

...
Timeline About Friends 3,138 Photos More

When did you work at Opendream?
... 22 Pending Items

Post Photo/Video Live Video Life Event

What's on your mind?

Public Post

Intro
Software Craftsmanship

Software Practitioner at สยามชัมนาภิกิจ พ.ศ. 2556

Agile Practitioner and Technical at SPRINT3r

Somkiat Puisungnoen 15 mins · Bangkok · ⚙️

Java and Bigdata





Page

Messages

Notifications 3

Insights

Publishing Tools

Settings

Help ▾



somkiat.cc

@somkiat.cc

Home

Posts

Videos

Photos



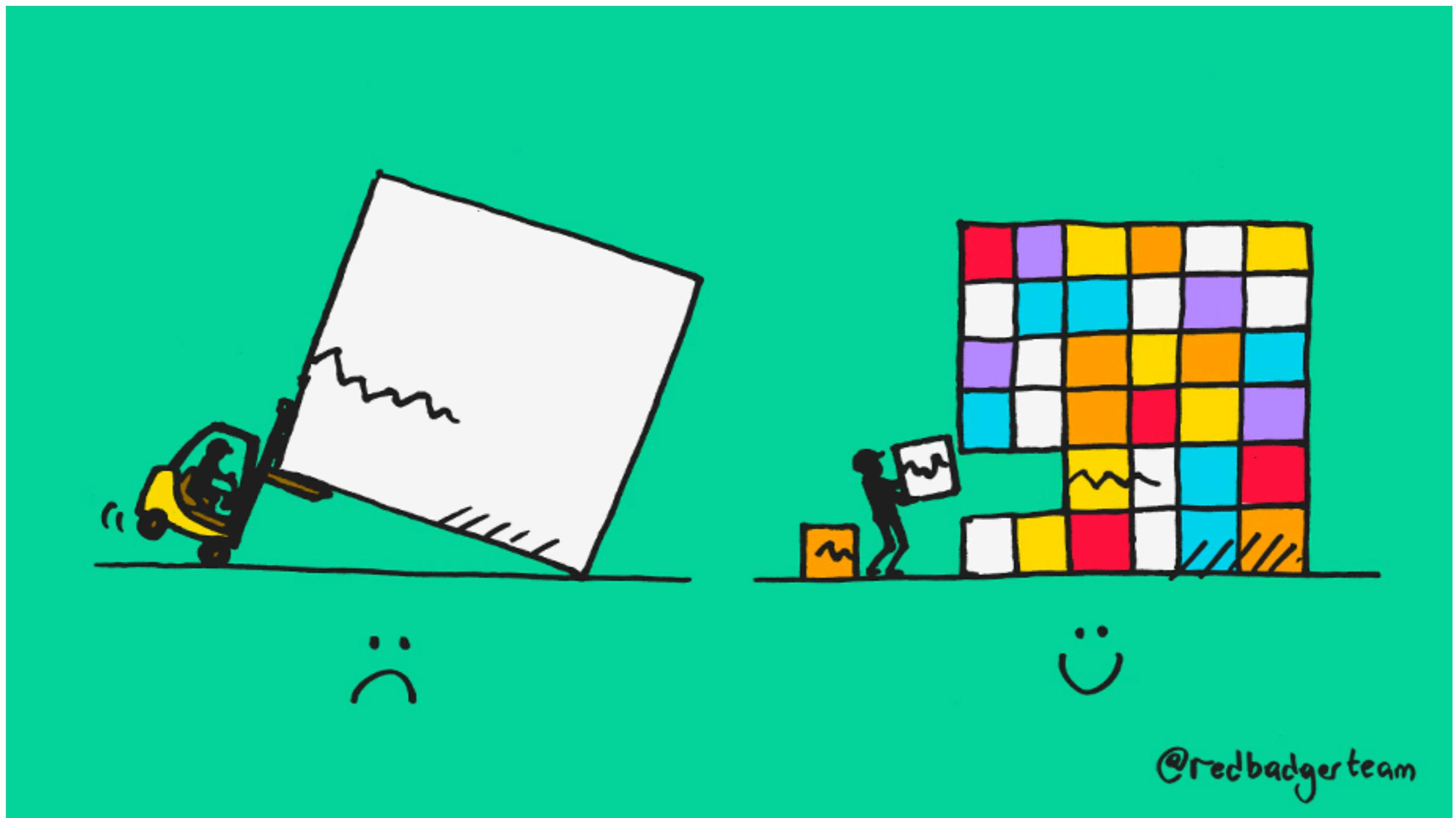
Agenda

How to deploy Microservices ?
Continuous Integration and Delivery
Practices of Continuous Integration
Deployment strategies
Working with containerization (Docker)
Workshop

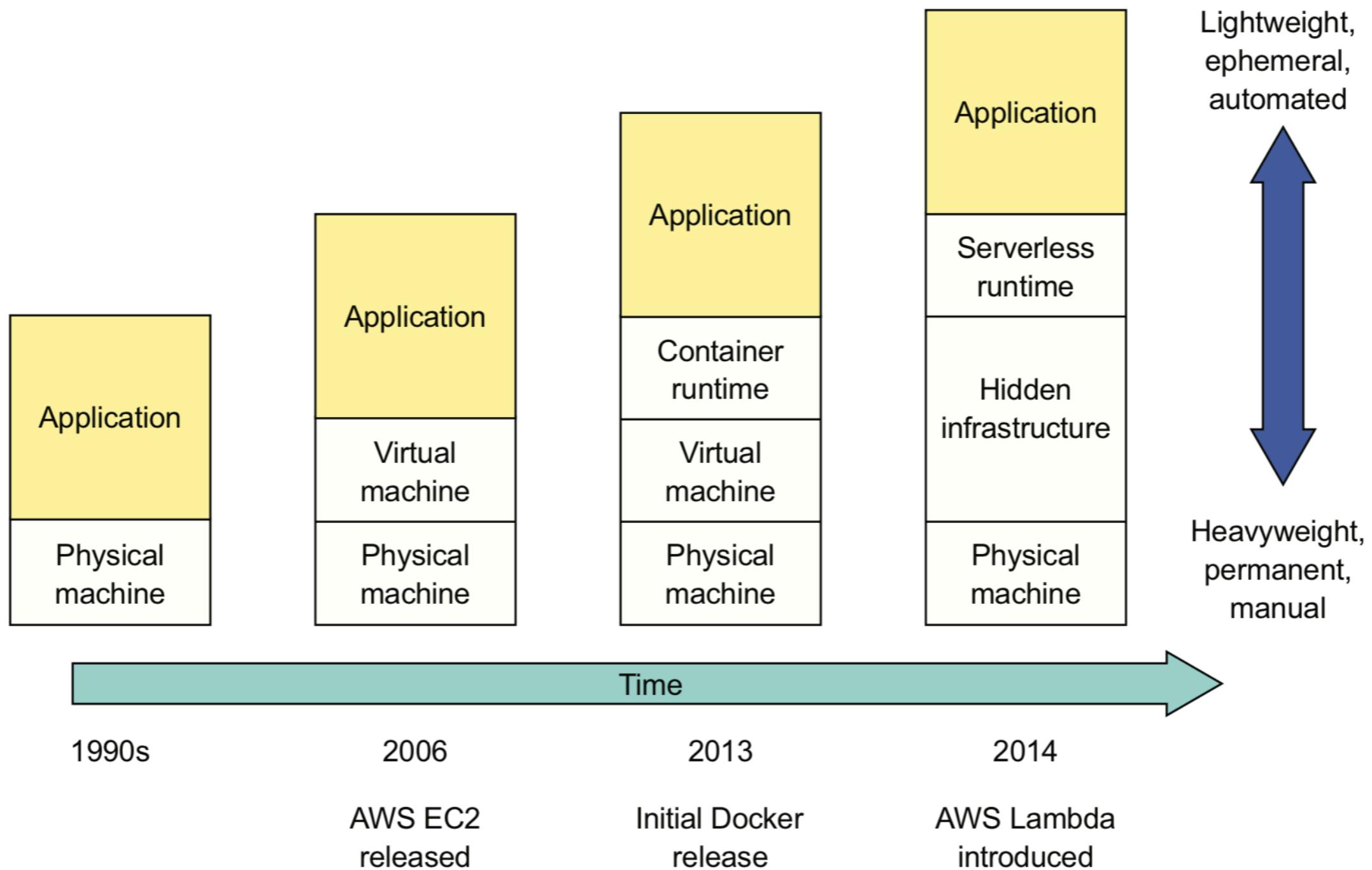


Deployment services

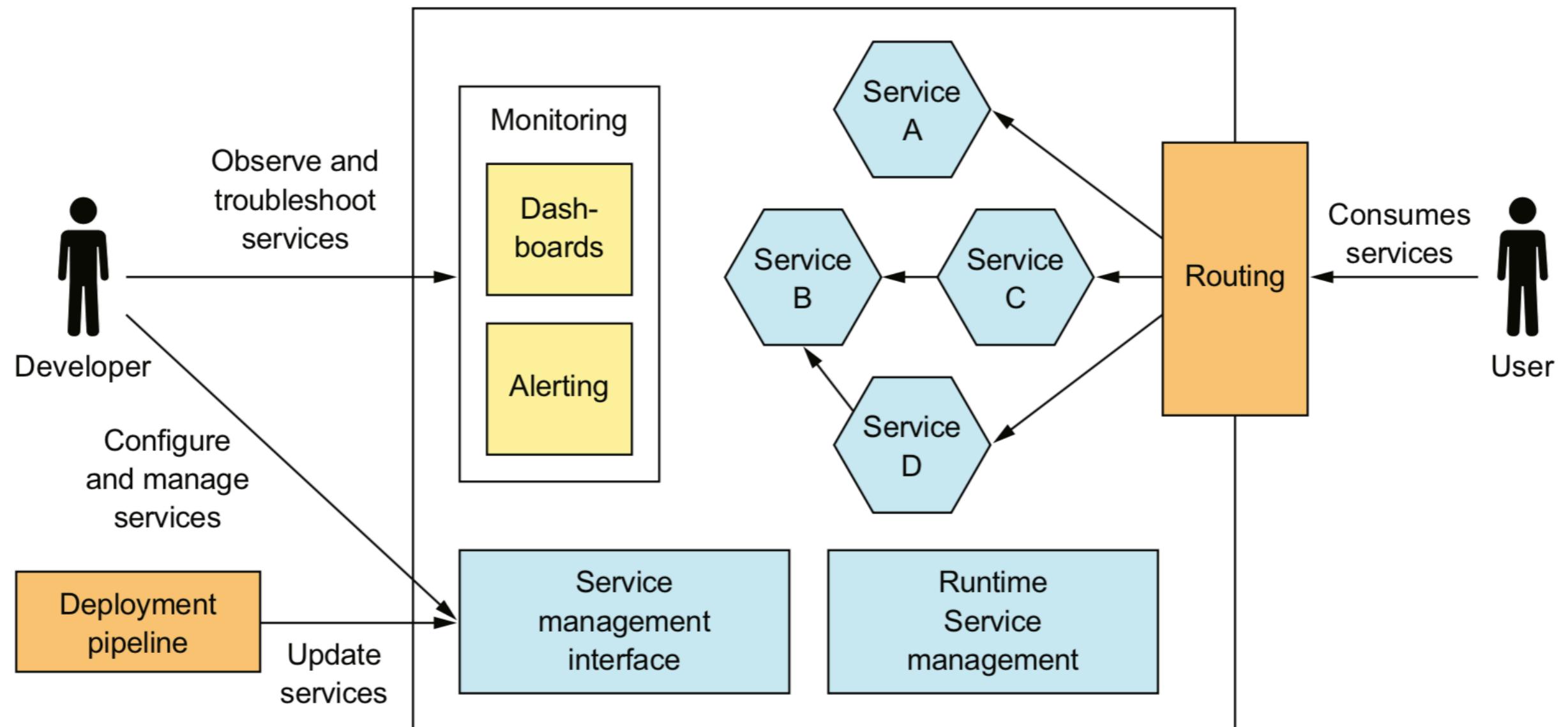




Infrastructure



View of production environment



Deploy Microservices

Language-specific packaging

Virtual Machine (VM)

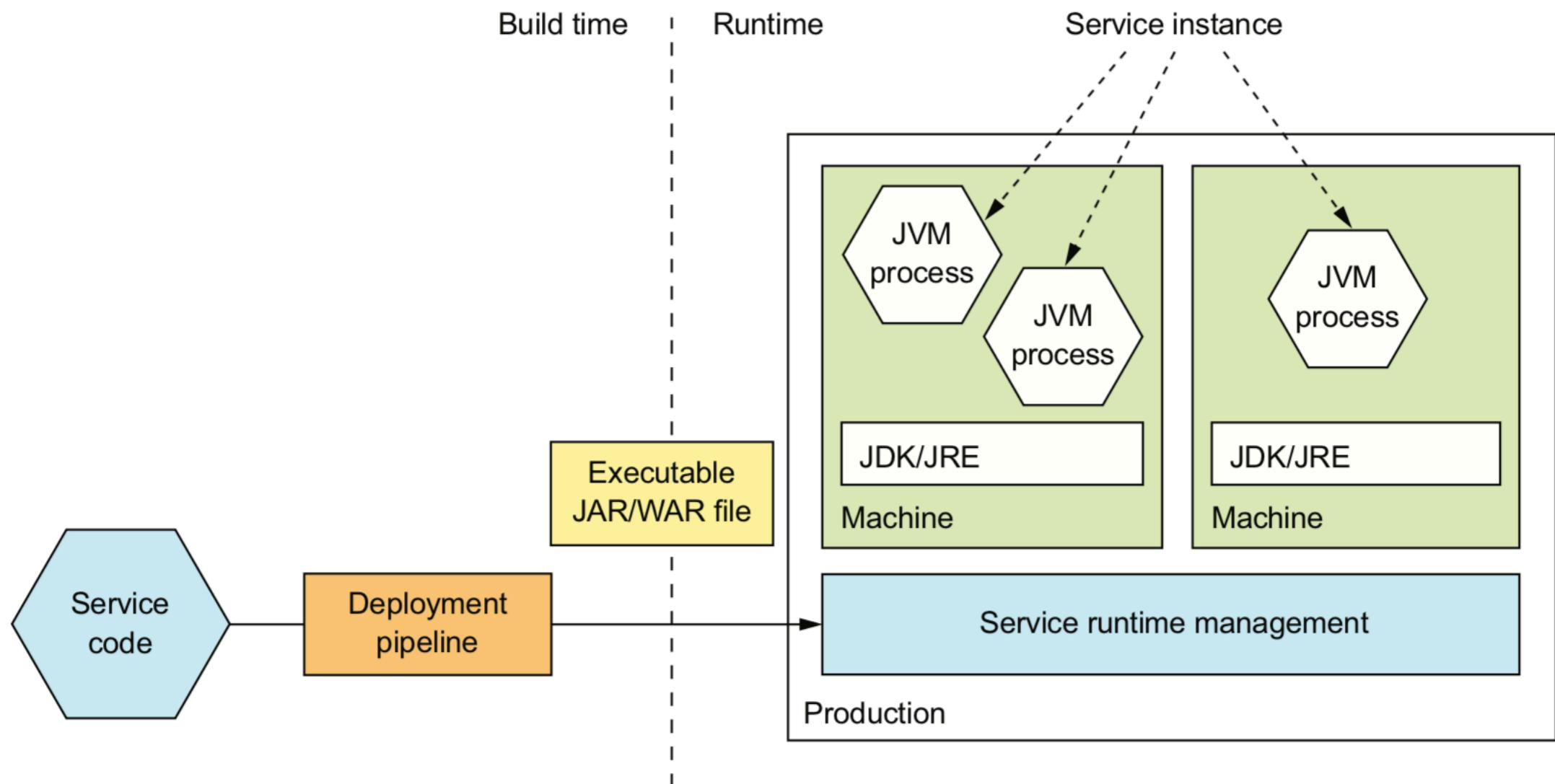
Container

Kubernetes

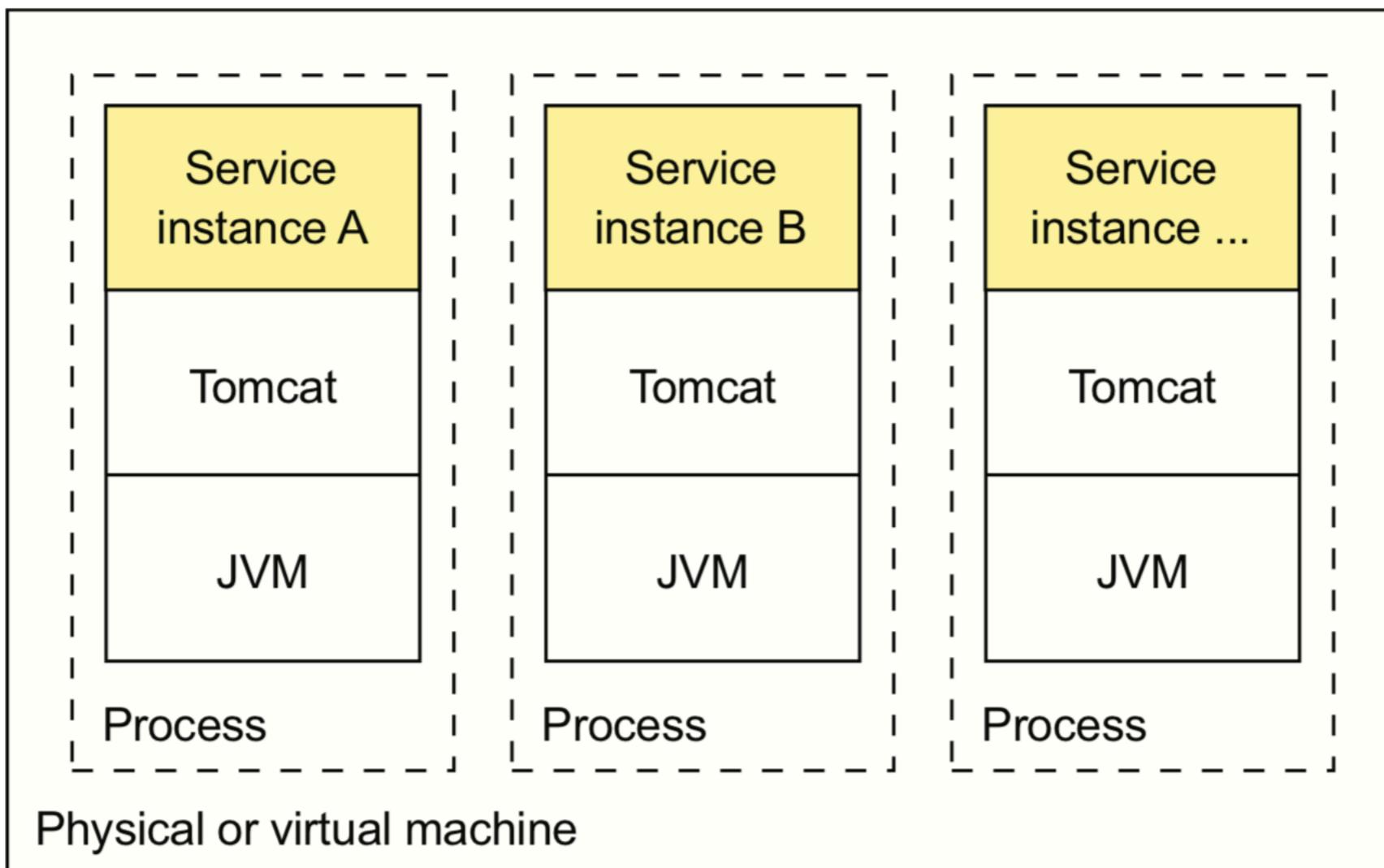
Serverless/FaaS



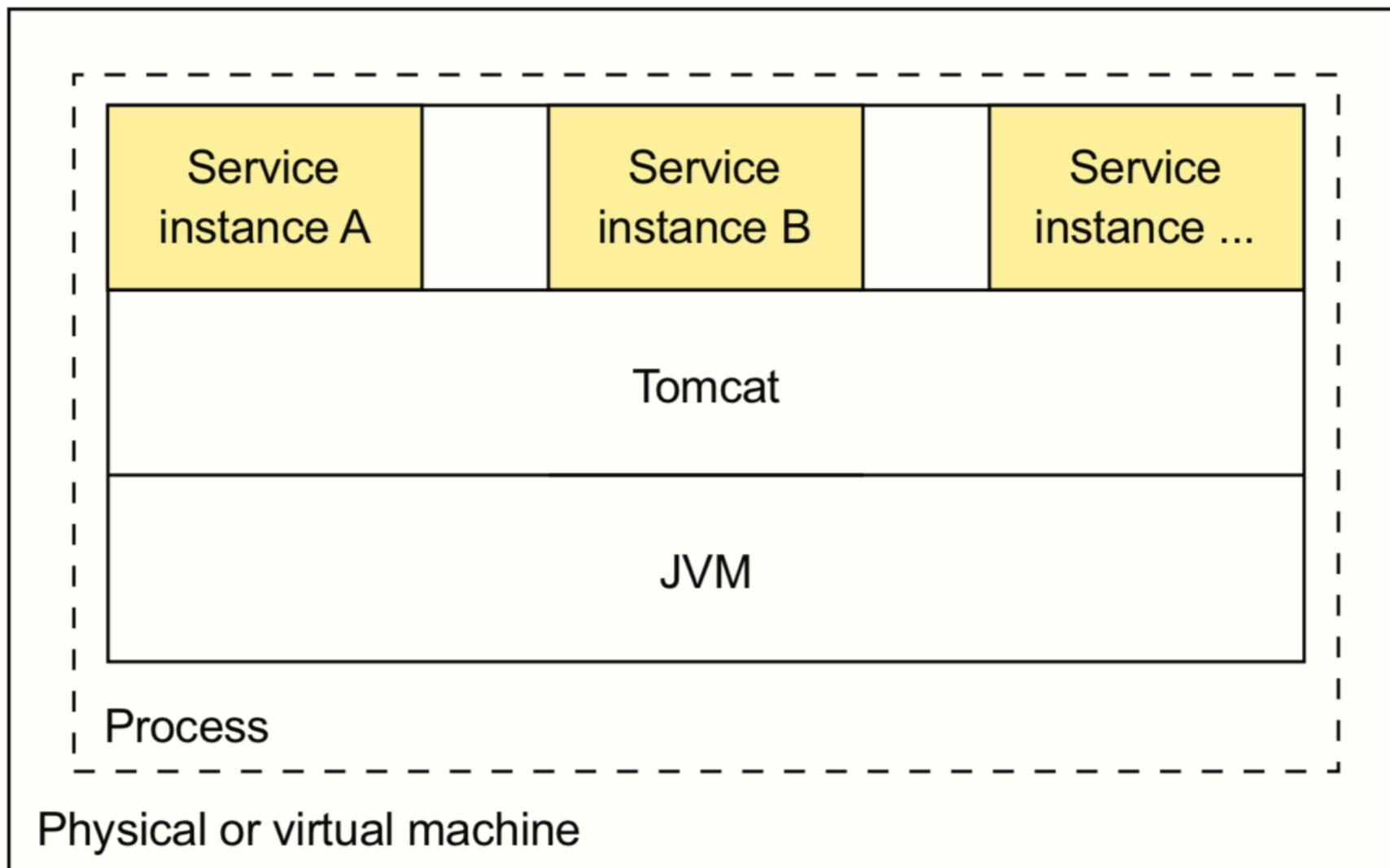
1. Language-specific packaging



Multiple services on same machine



Multiple services on same process



Benefits

Fast deployment

Efficient resource utilization

Service instances's resources are constrained



Drawbacks

Lack of encapsulation of technology stack

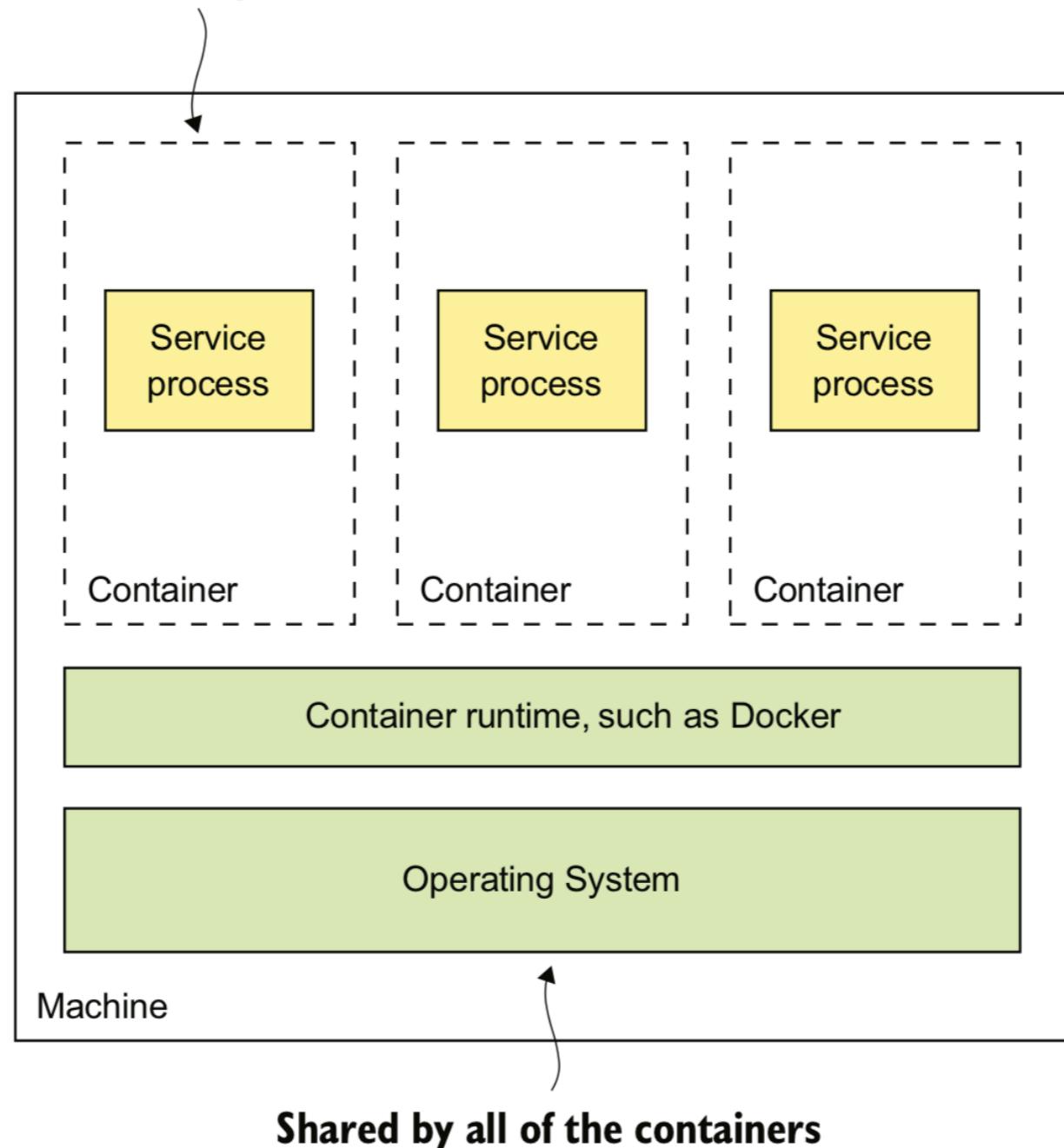
No ability to constrain resources of service

Lack of isolation

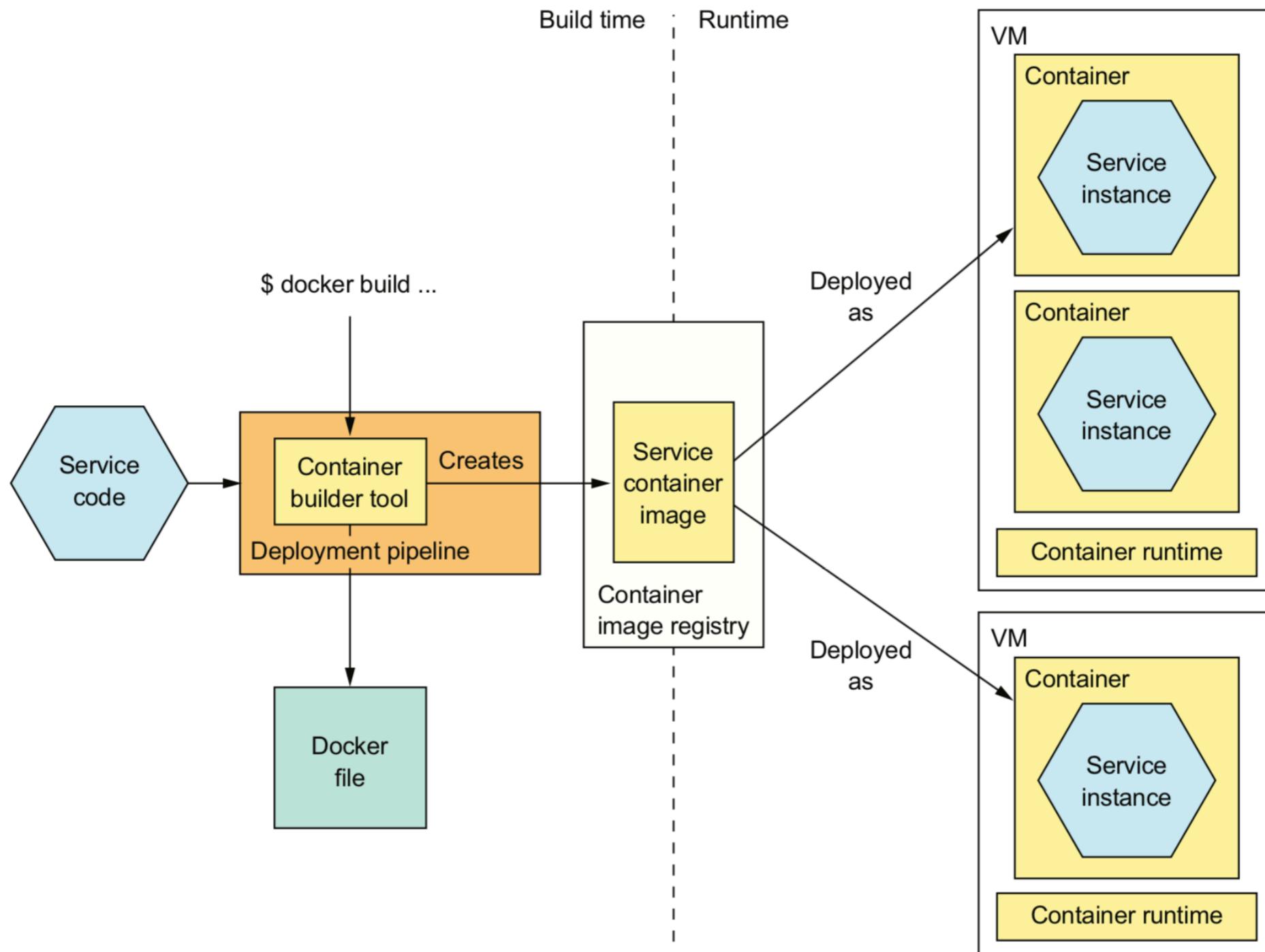


2. Working with container

**Each container is a sandbox
that isolates the processes.**



Deployment with container



Deploy services with Docker

Build a docker image

Push docker image to a registry

Run docker container

Working with docker-compose



Benefits

Encapsulate technology stack

Service instances are isolated

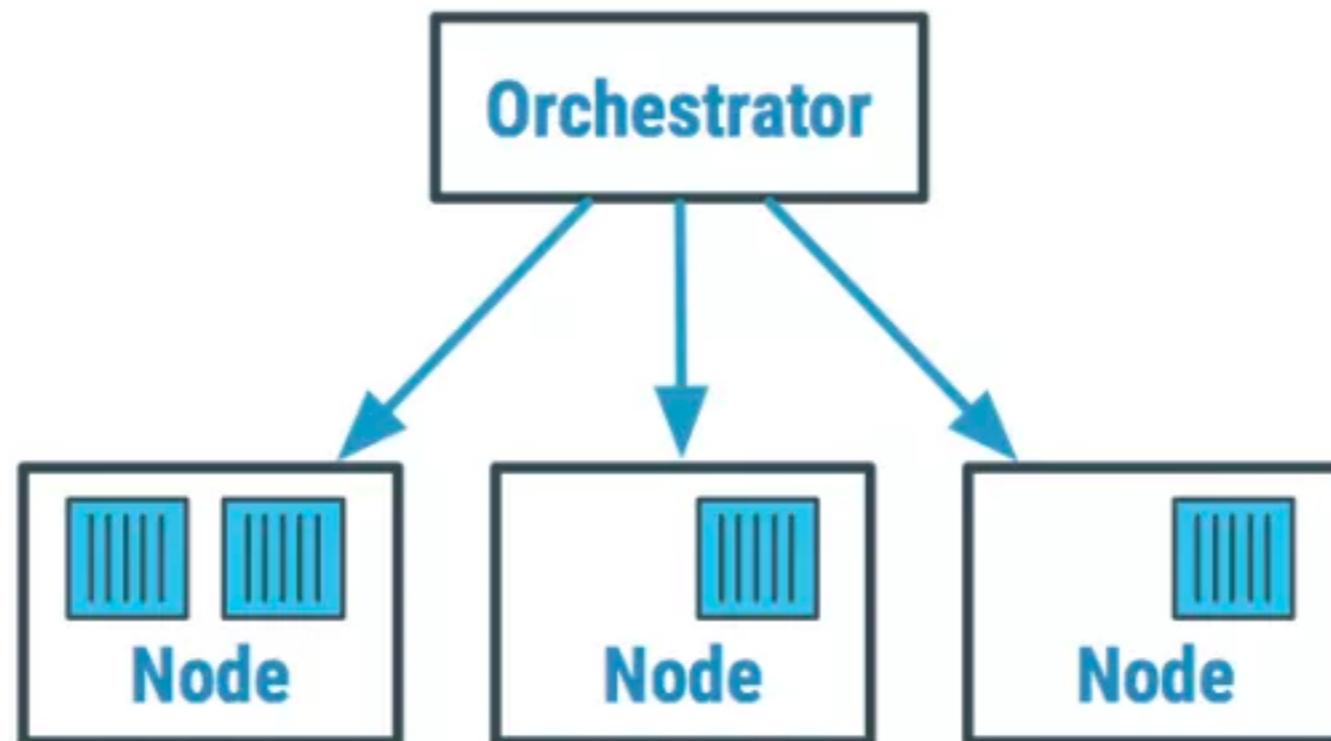
Service instances's resources are constrained



Workshop



Container Orchestration ?



Orchestration tools

Configuration Management



CI/CD orchestration



Container orchestration



Cloud-specific orchestration



PaaS orchestration



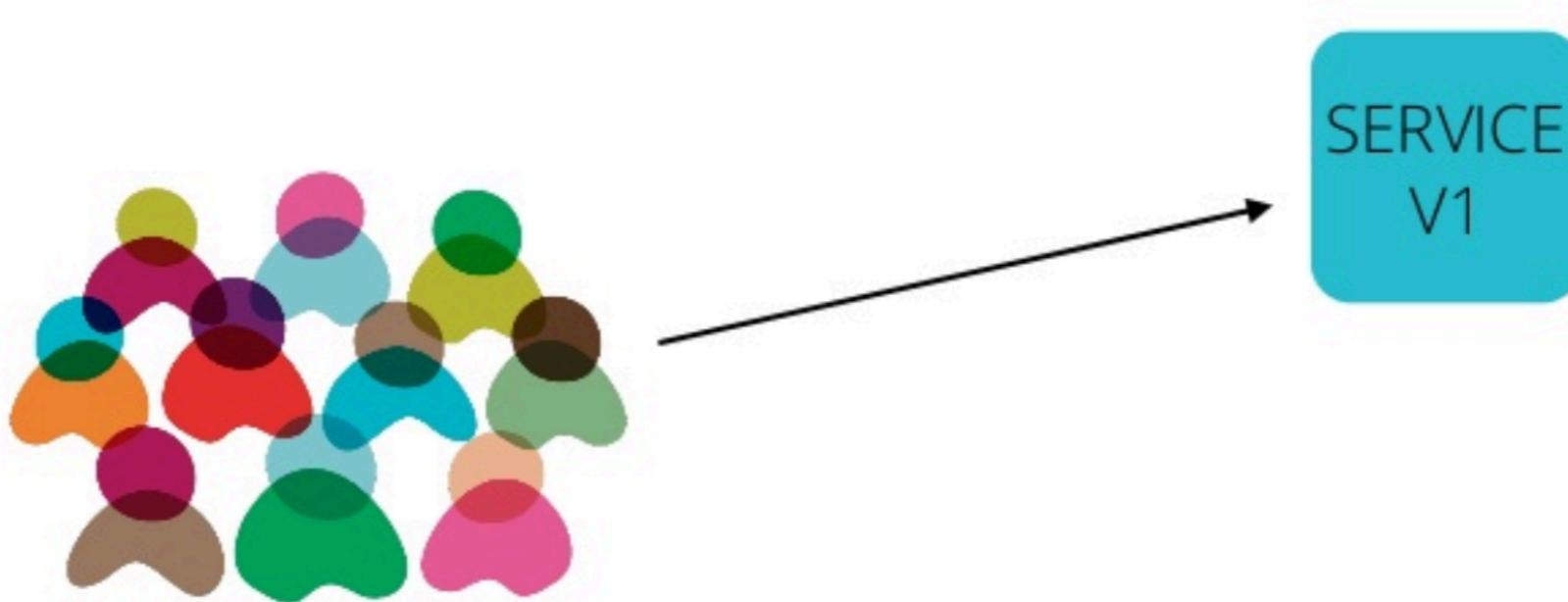
Deployment strategies



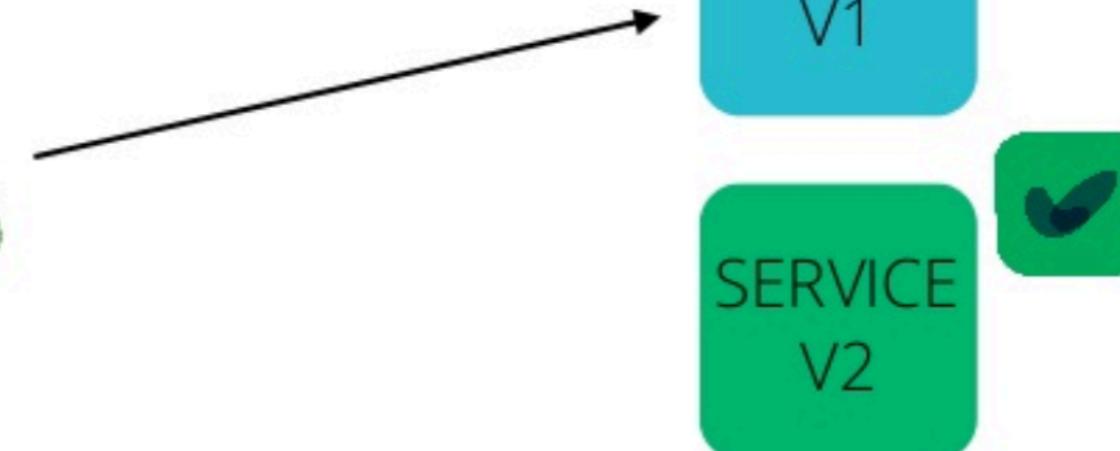
Blue Green Deployment



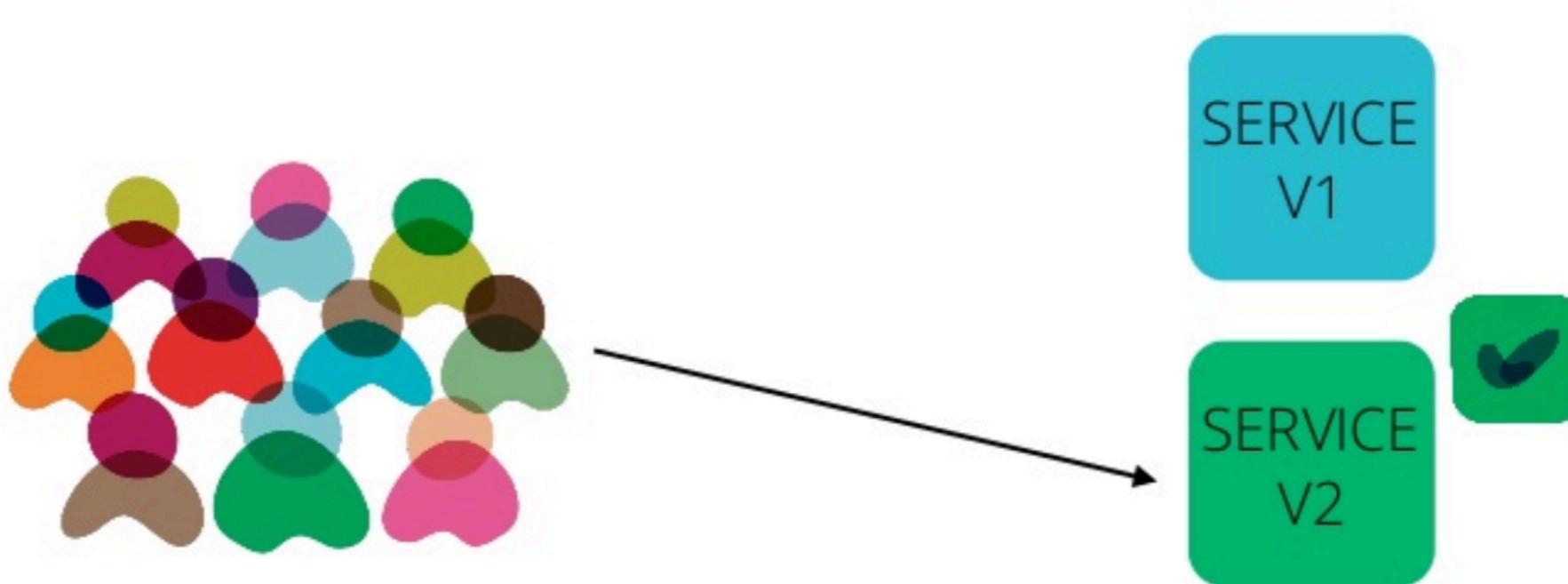
Blue Green Deployment



Blue Green Deployment



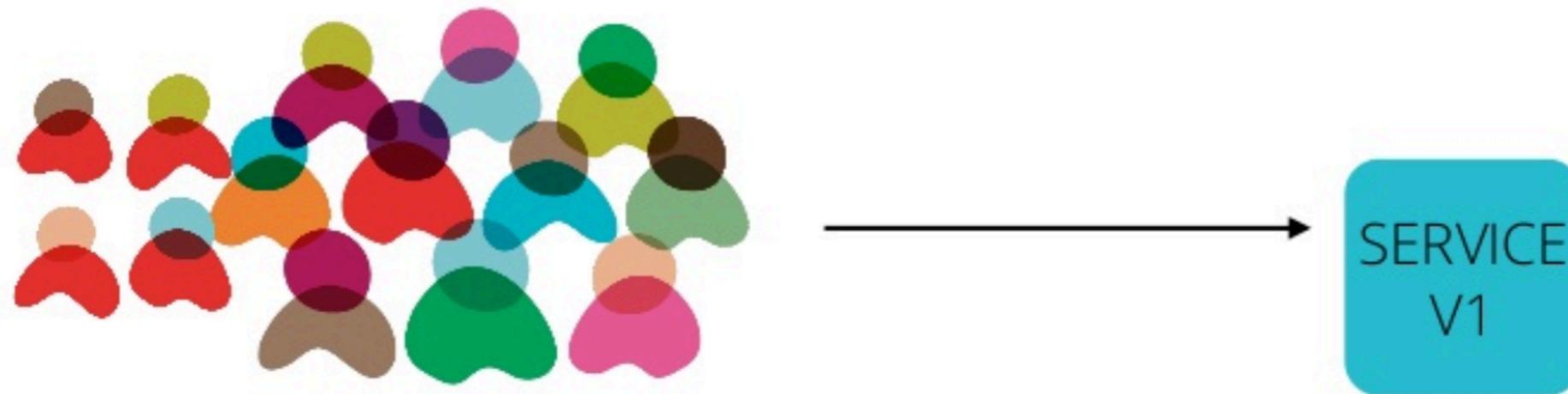
Blue Green Deployment



Canary Release



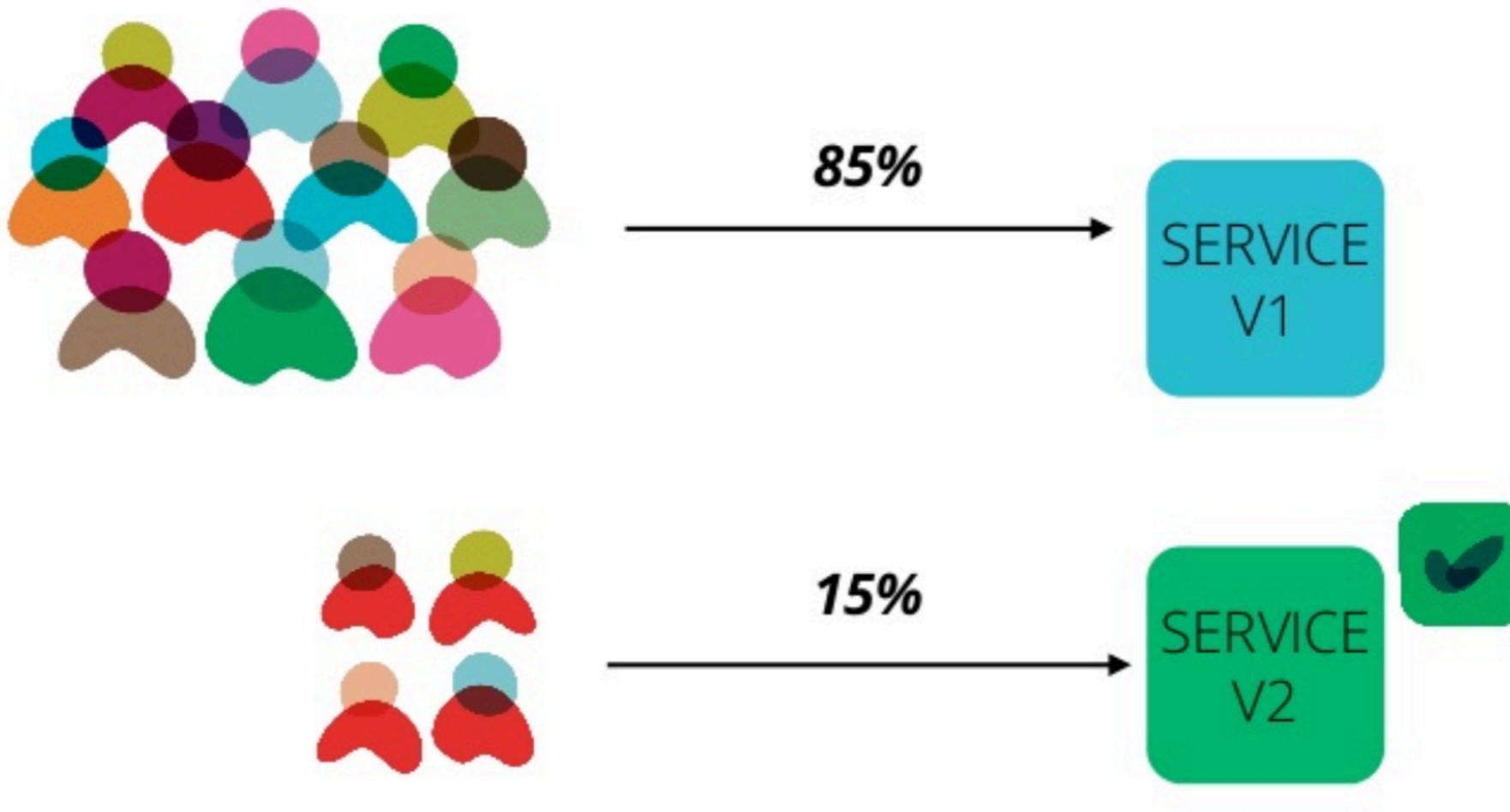
Canary Release



Canary Release



Canary Release



Mean Time to Recover (MTTR)



Mean Time to Recover (MTTR)

Tests are very important to reduce amount of defects in your systems. However, it's important to acknowledge that bugs will always happen in production.

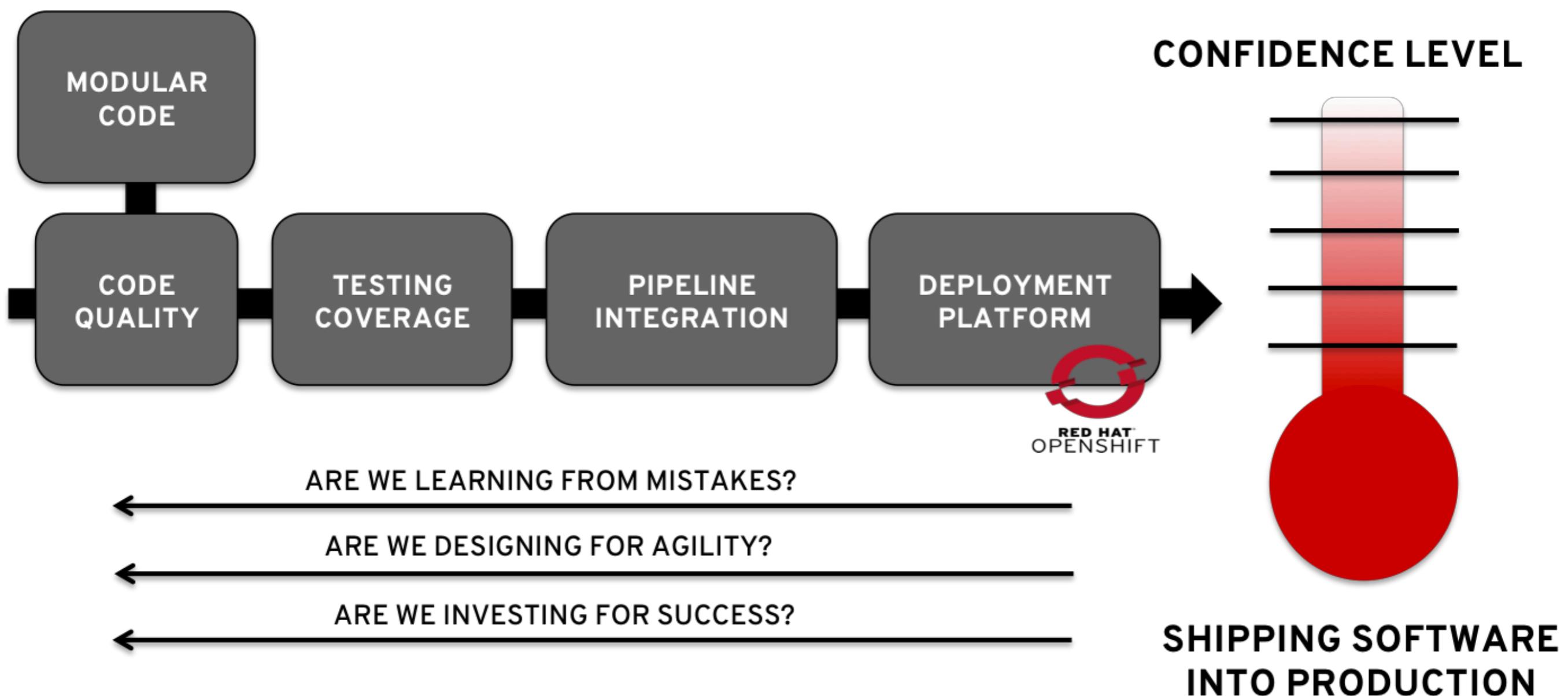


Mean Time to Recover (MTTR)

How **fast** to recover from them will help determining our success !



What can i measure ?



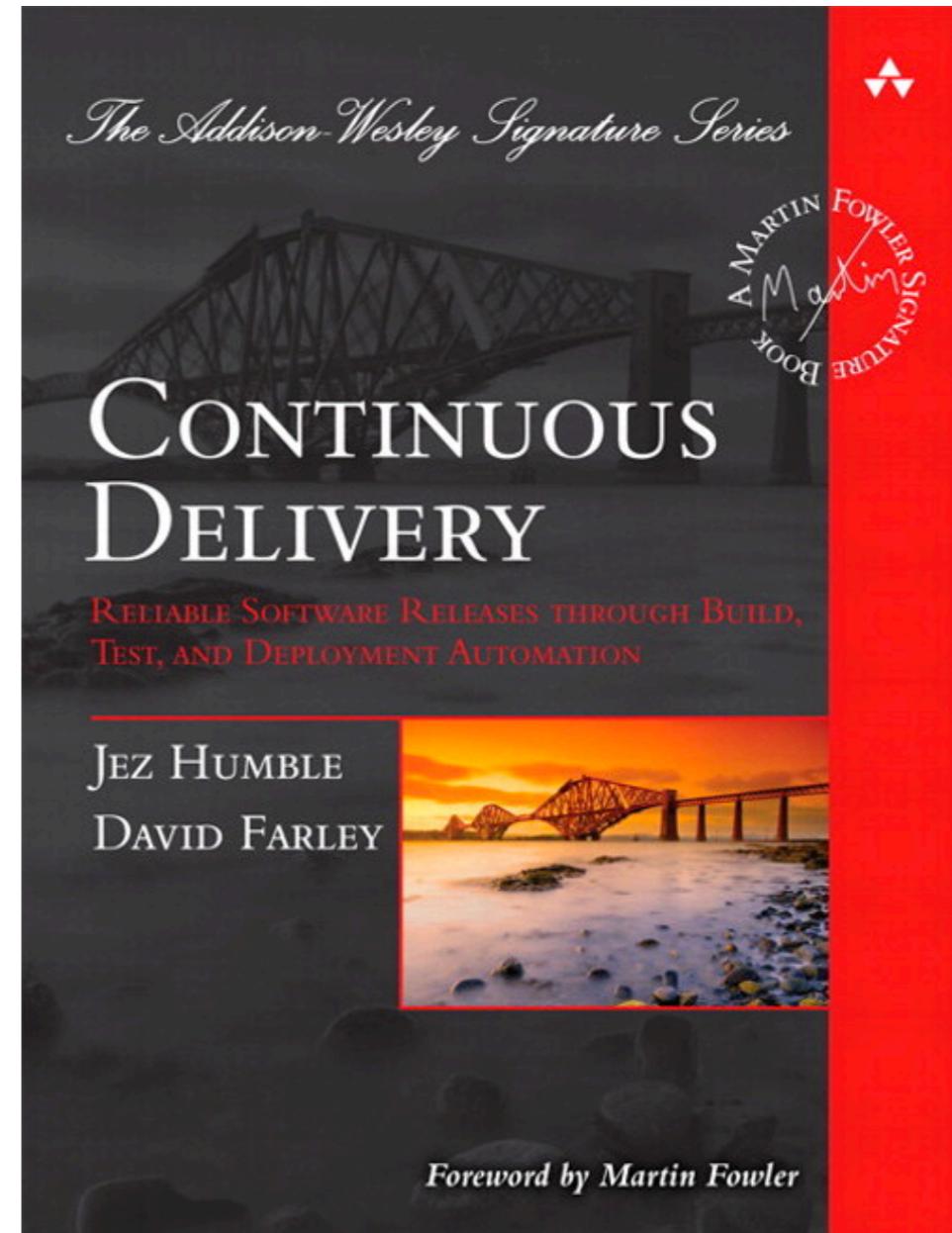
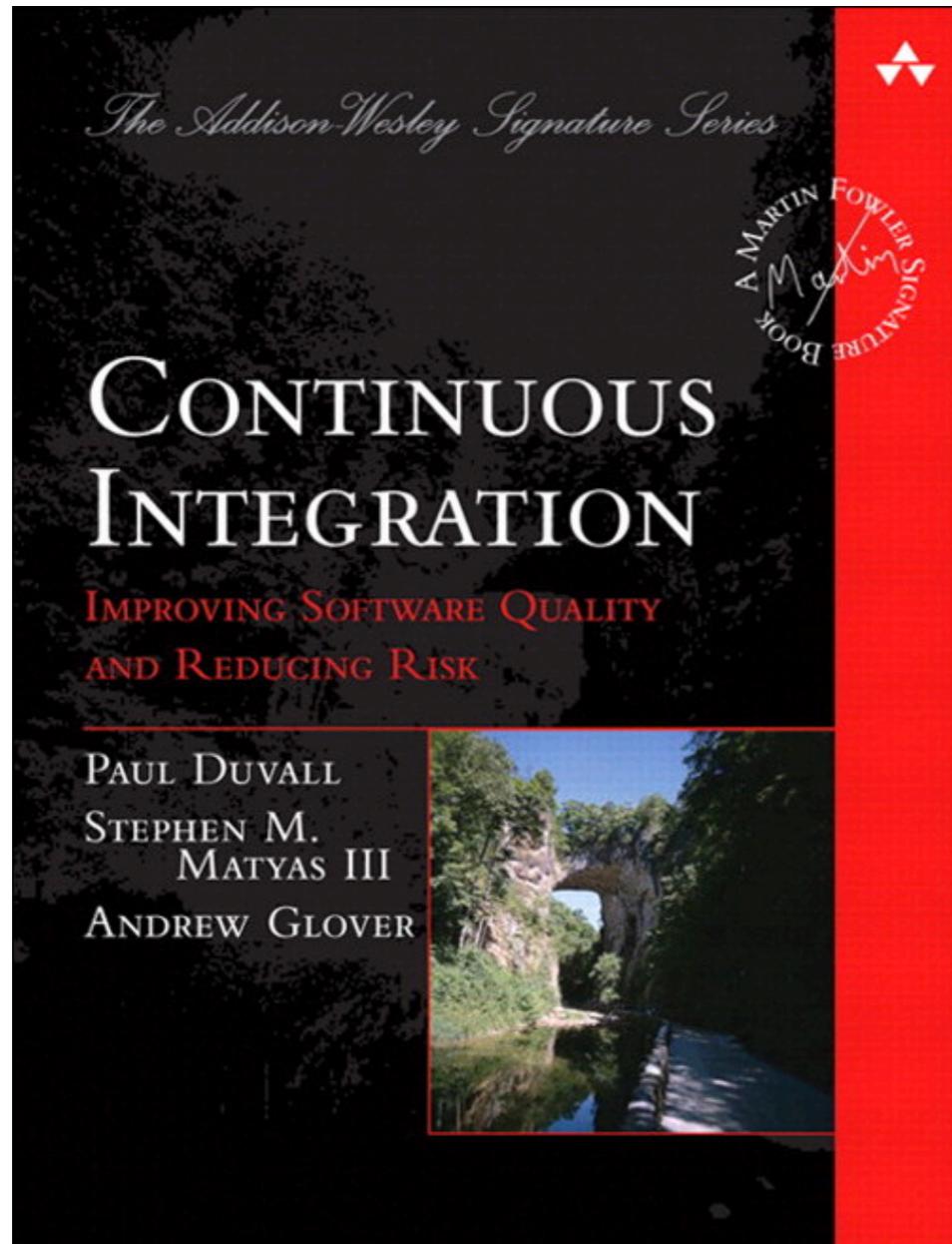
Start with Continuous Integration Continuous Delivery



**“Behind every successful agile
project, there is a
Continuous Integration Server”**



Improve quality and reduce risk

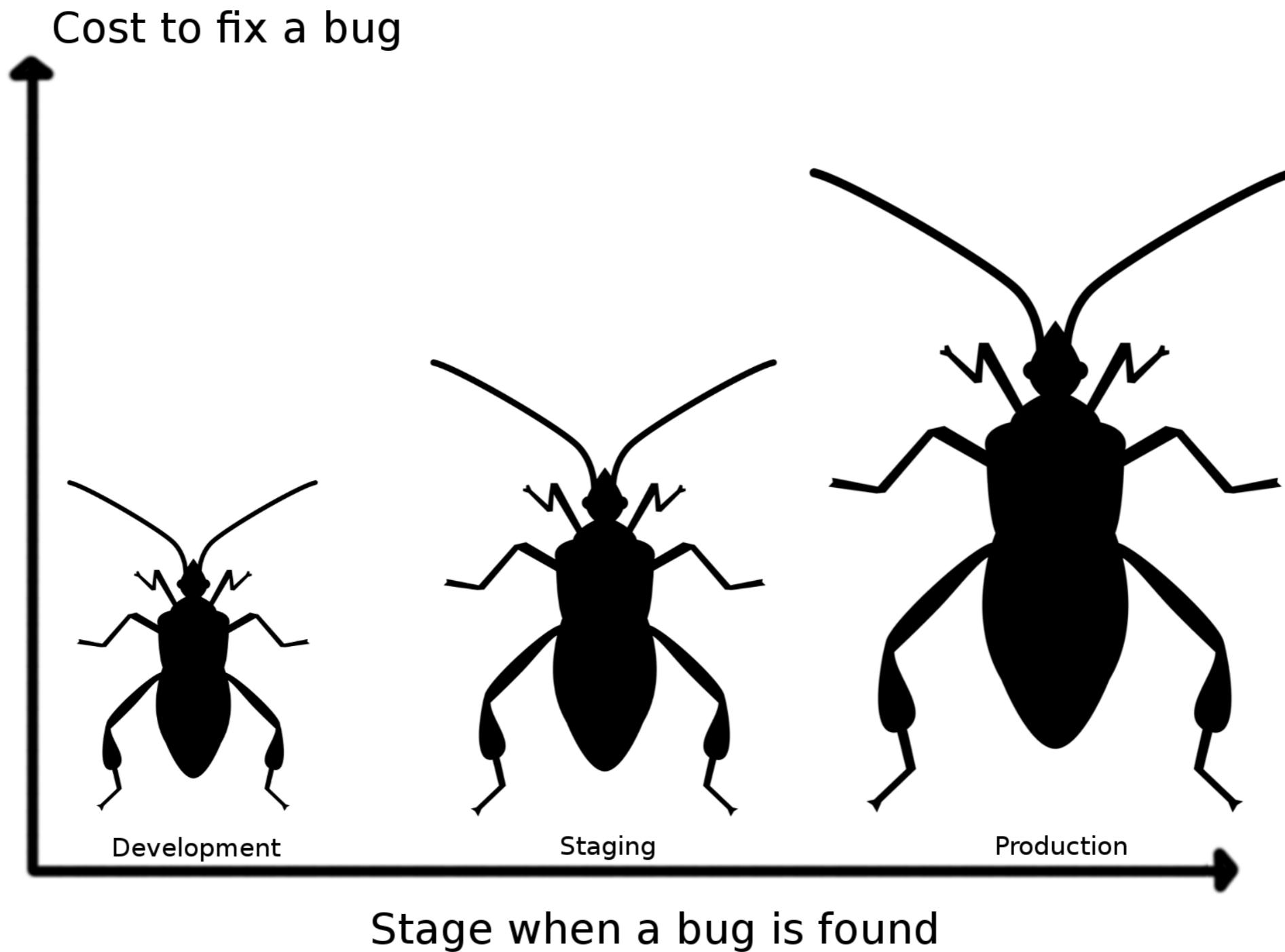


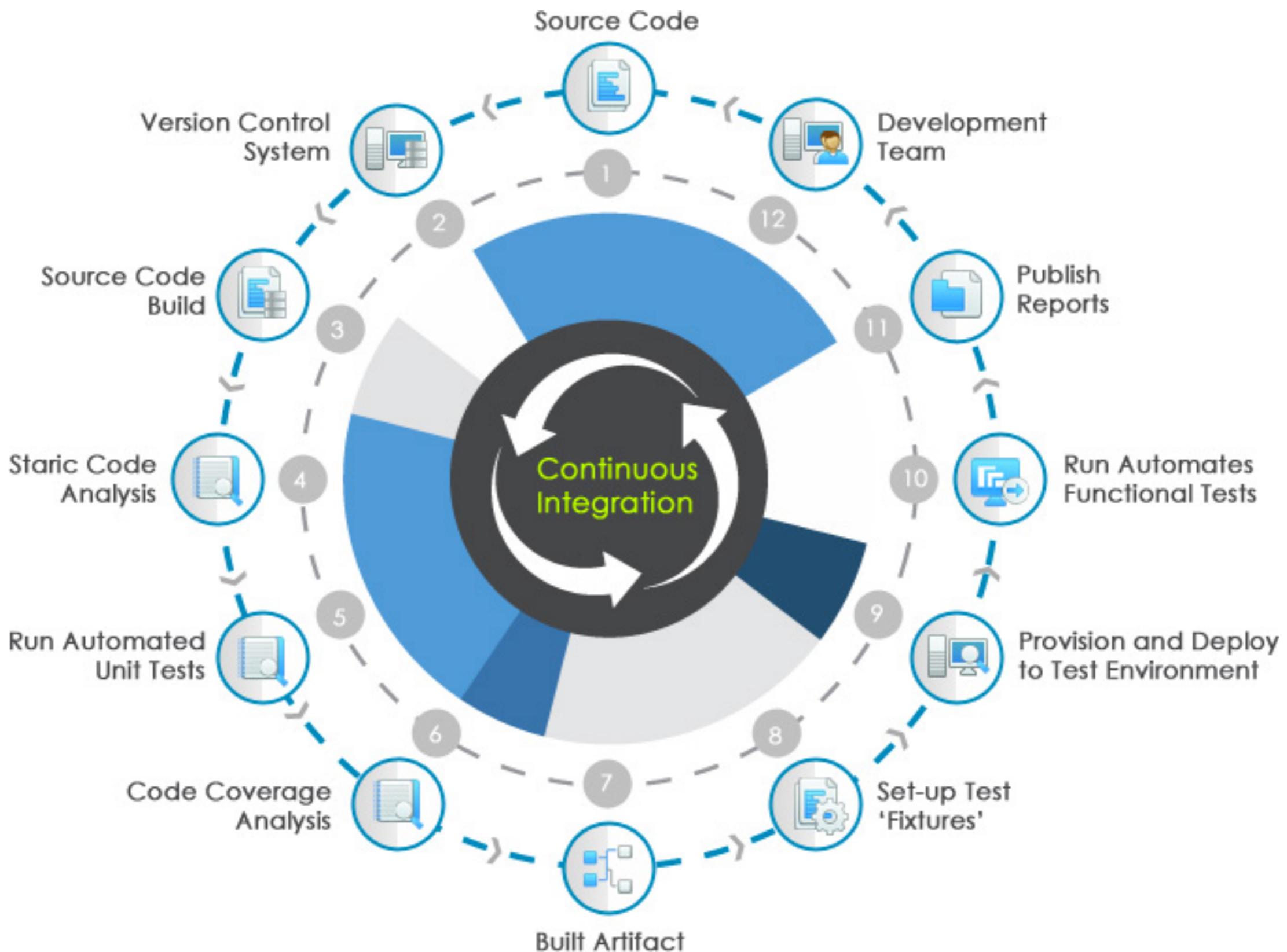
The cost of integration

1. Merging the code
2. Duplicate changes
3. Test again again !!
4. Fixing bugs
5. Impact on stability



The cost of integration







Jenkins

Bamboo



TeamCity



Hudson





Jenkins



Bamboo

CI is about what people do
not about what tools they use



Visual Studio



Team Foundation Server

Hudson



Microservices

© 2017 - 2018 Siam Chamnankit Company Limited. All rights reserved.

Continuous Integration

Discipline to integrate frequently



Continuous Integration

Strive to make **small change**

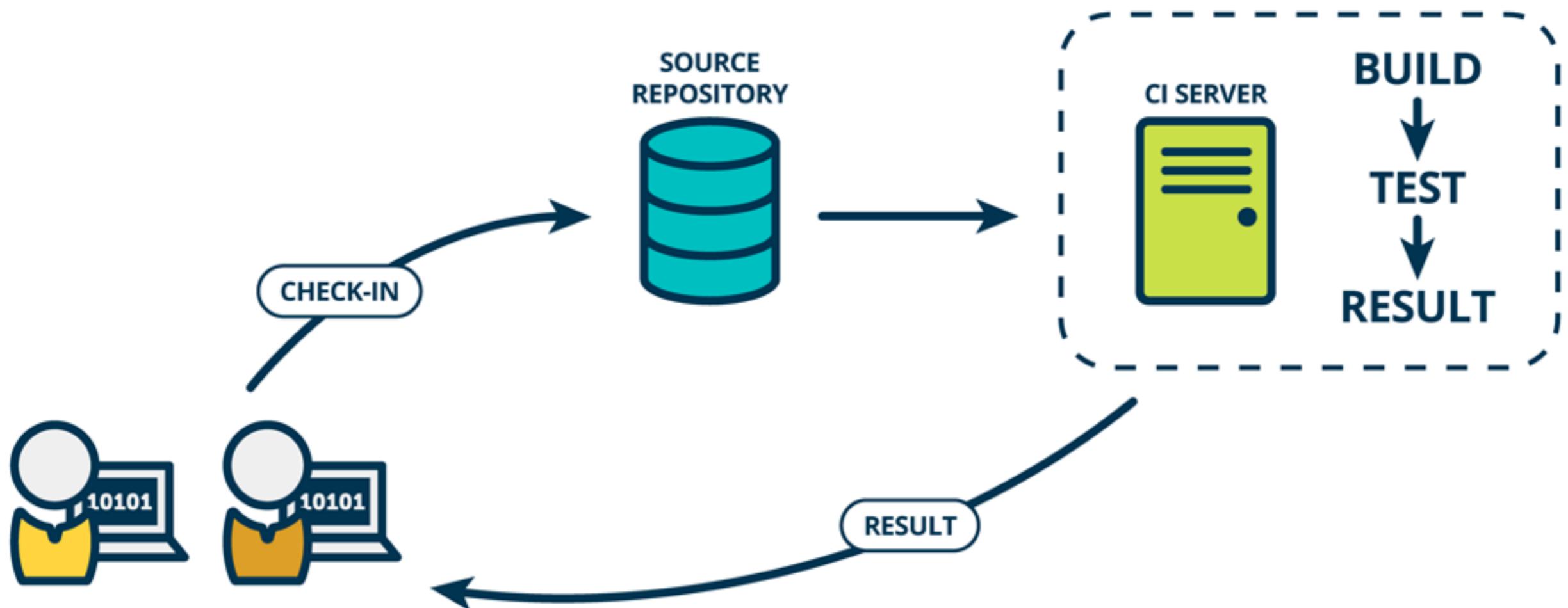


Continuous Integration

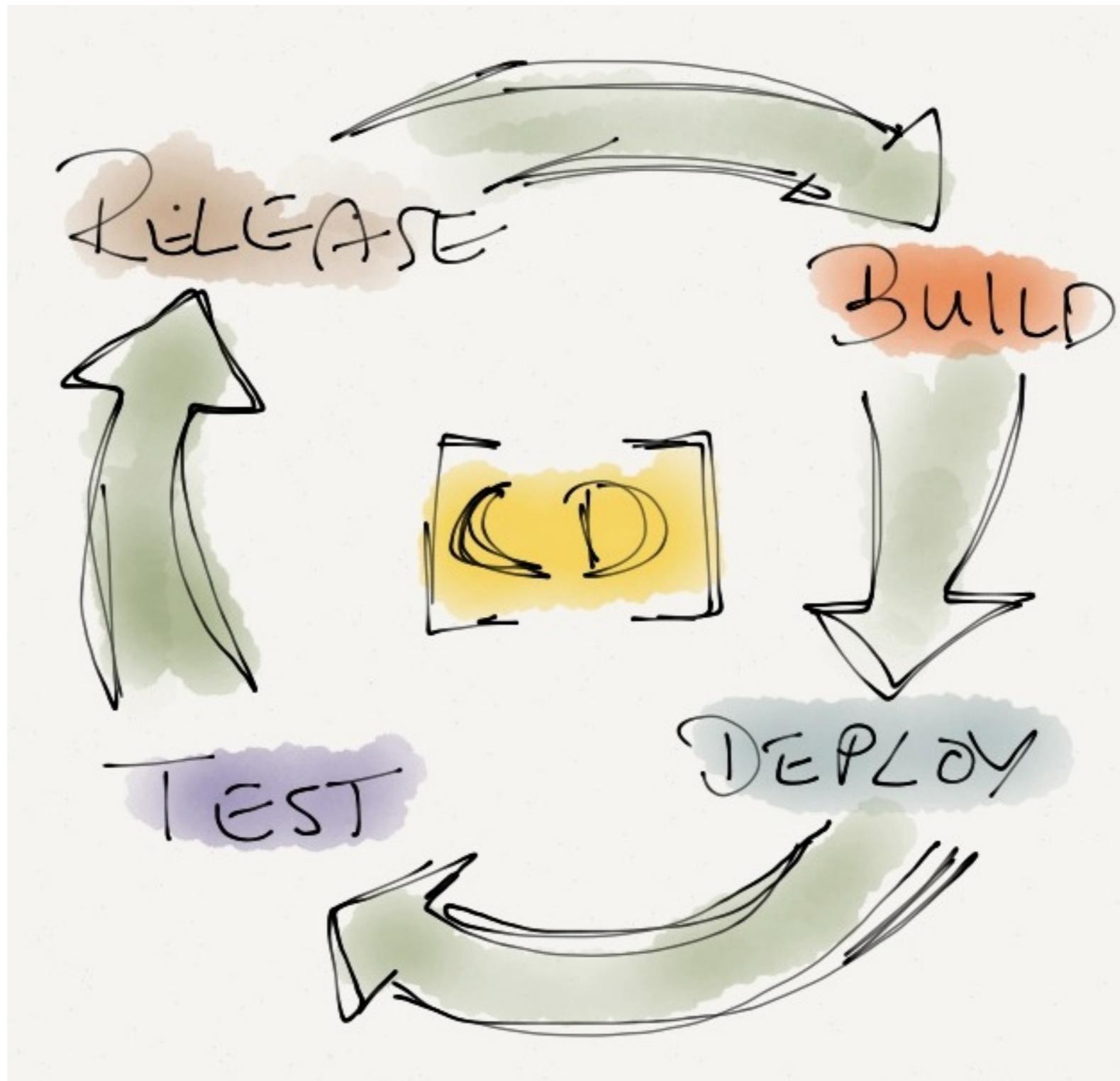
Strive for **fast feedback**



Continuous Integration



CD ?



CD ?

CONTINUOUS DELIVERY



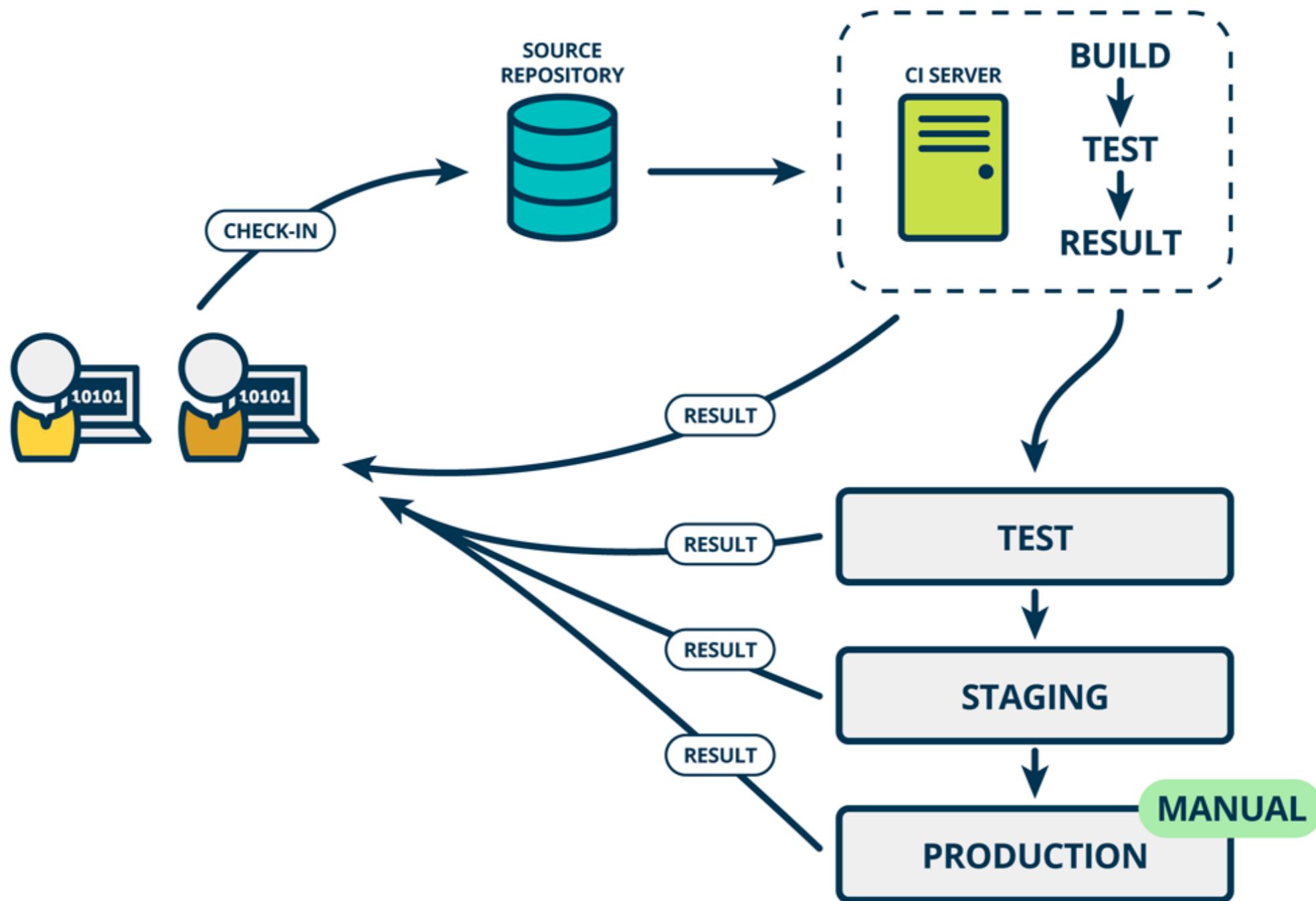
CONTINUOUS DEPLOYMENT



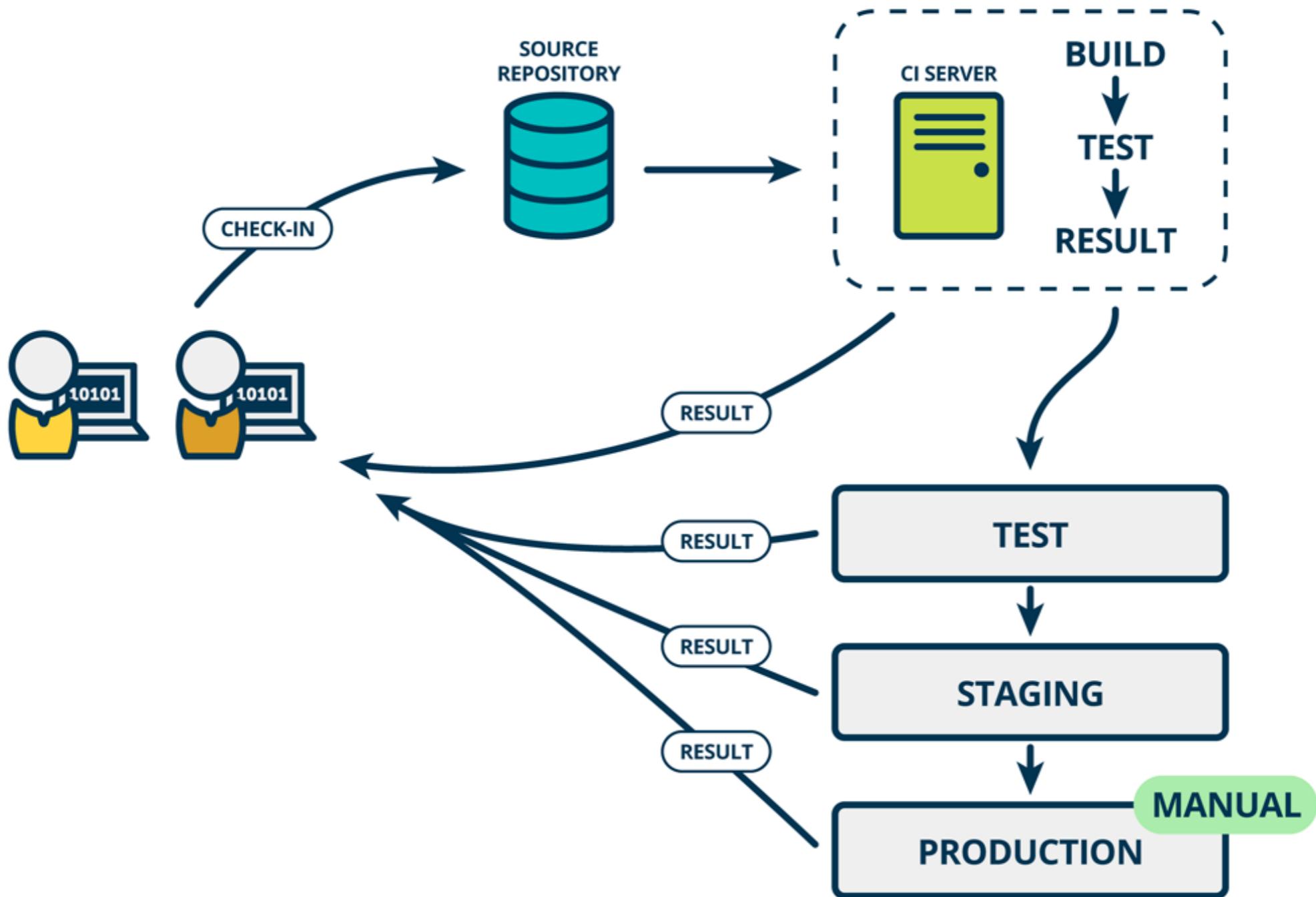
<http://blog.crisp.se/2013/02/05/yassalsundman/continuous-delivery-vs-continuous-deployment>



Continuous Delivery



Rise of DevOps



Continuous Integration

is a Software development practices



Practice 1

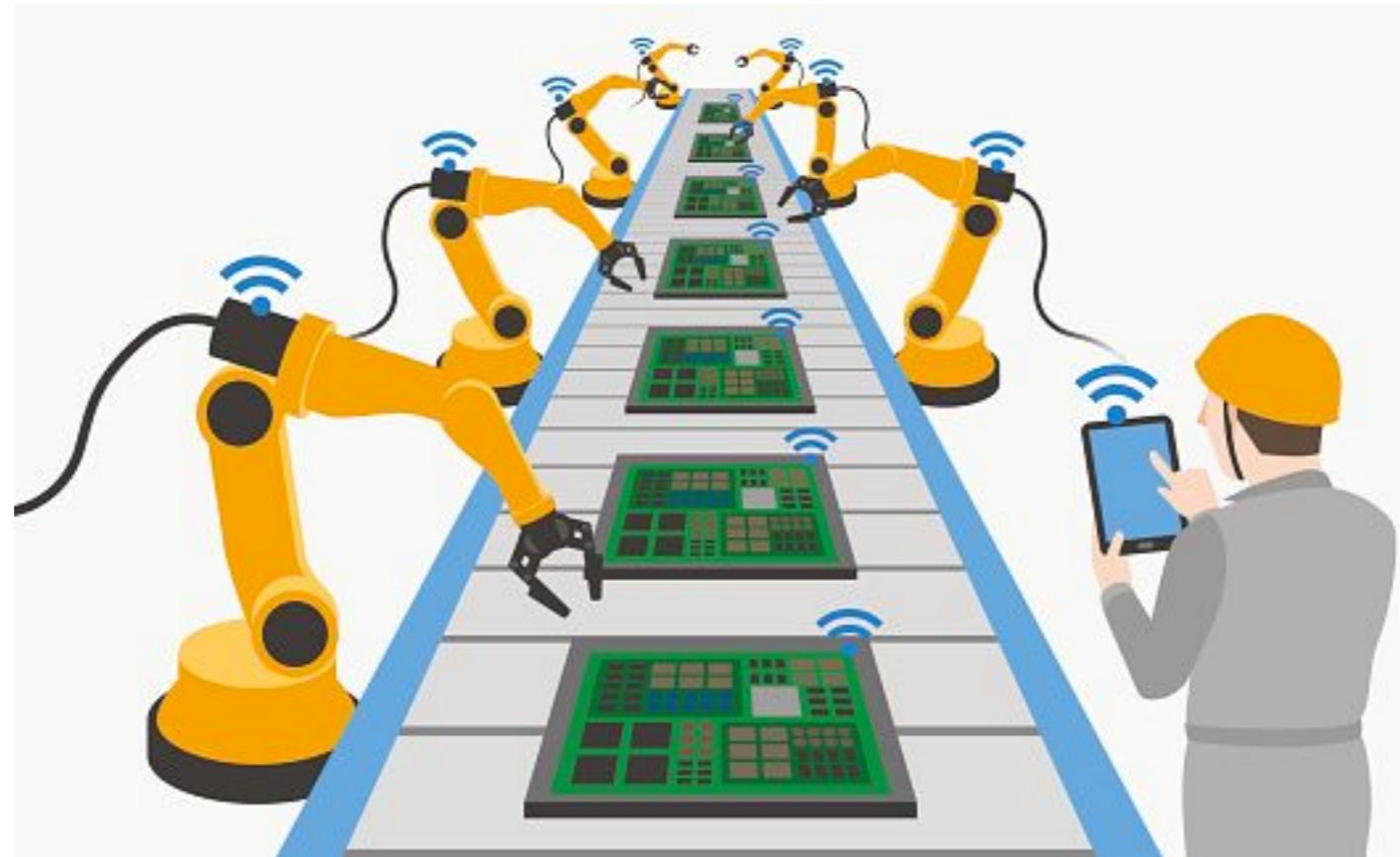
Maintain a single source repository

In general, you should store in source control
everything you need to build anything



Practice 2

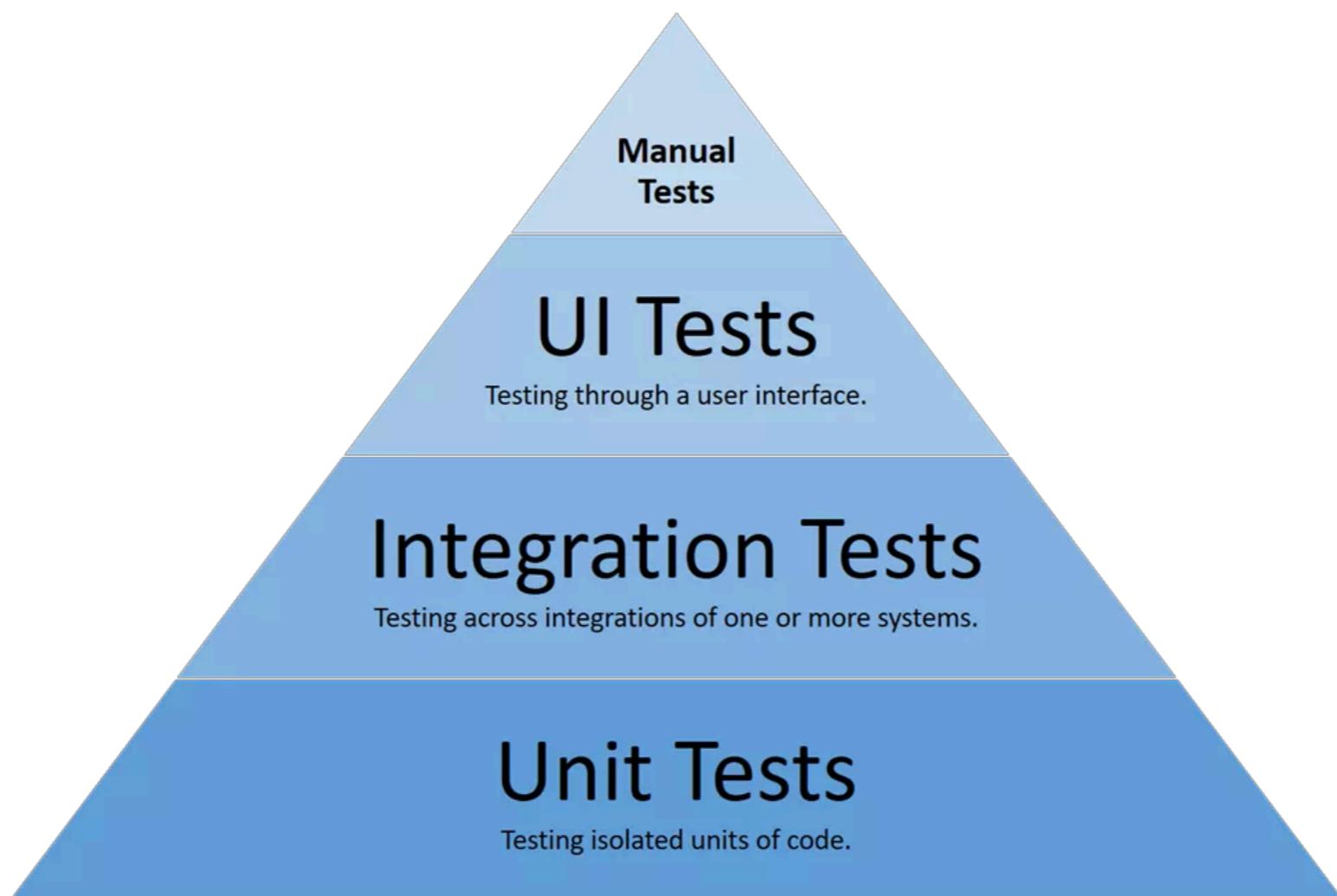
Automated the build
Automated environment for builds



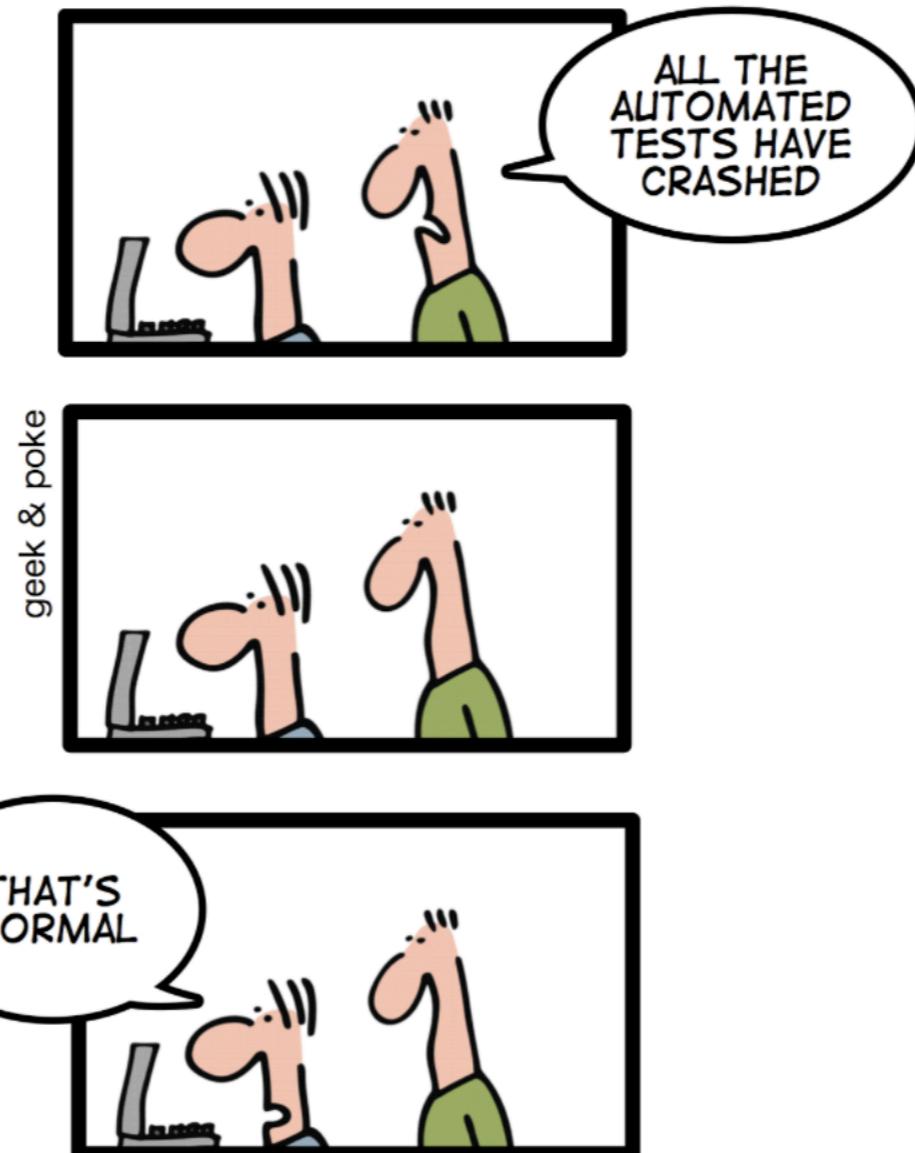
Practice 3

Make your build **self-testing**

Build process => compile, linking and **testing**



*TODAY: CONTINUOUS INTEGRATION
GIVES YOU THE COMFORTING
FEELING TO KNOW THAT
EVERYTHING IS NORMAL*

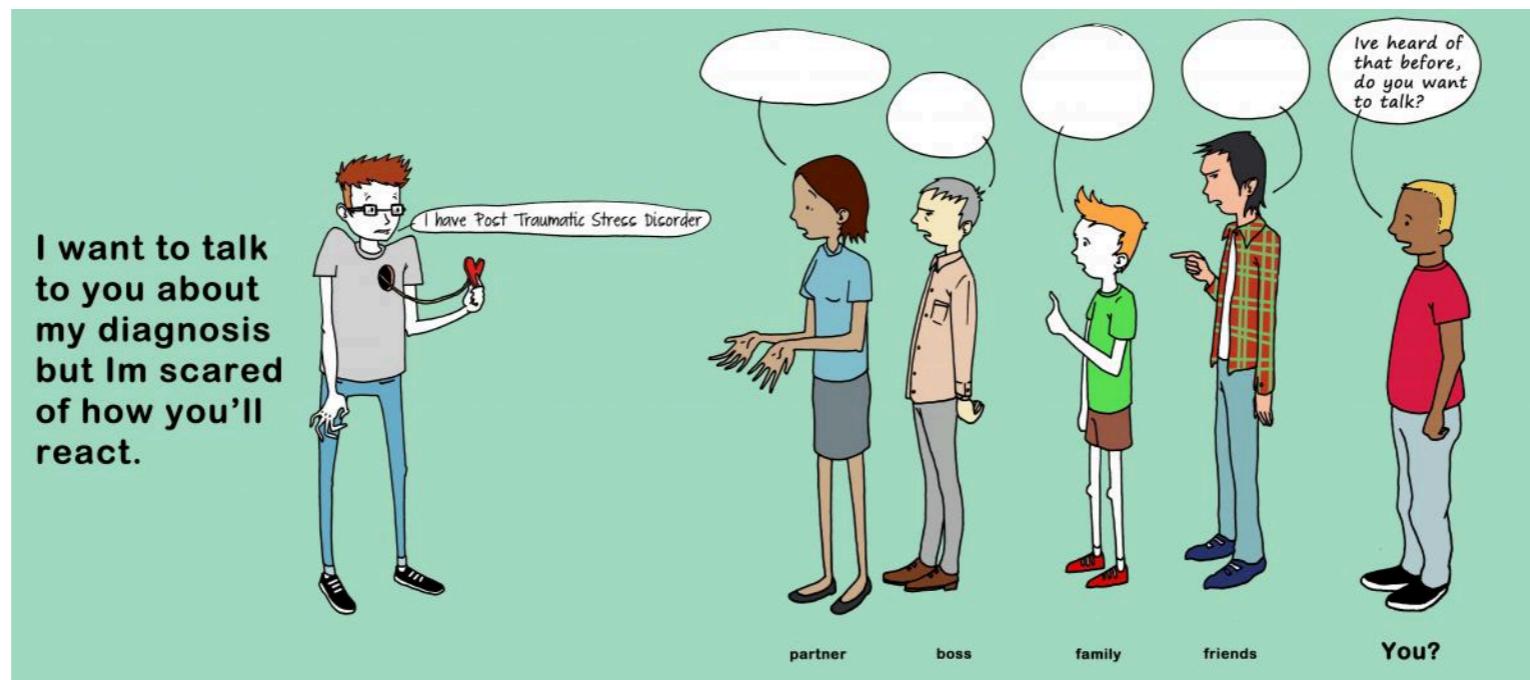


Practice 4

Everyone **commits** to the mainline everyday

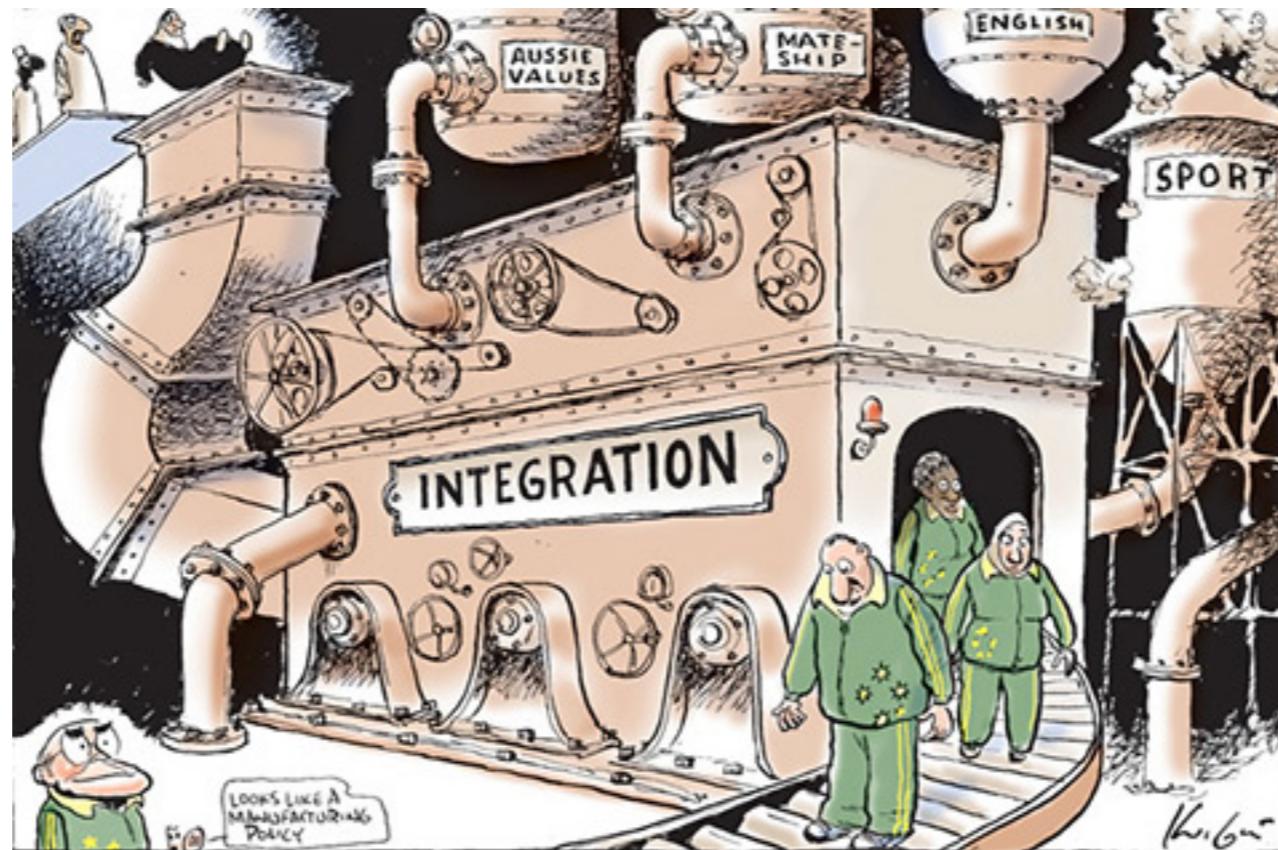
Integration is about **communication**

Integration allows developers to **tell** other developers



Practice 5

Every commits should build the mainline on an
Integration machine



Nightly build is not enough for Continuous Integration



Practice 6

Fix broken builds immediately

“Nobody has a higher priority task than fixing the build”



Practice 7

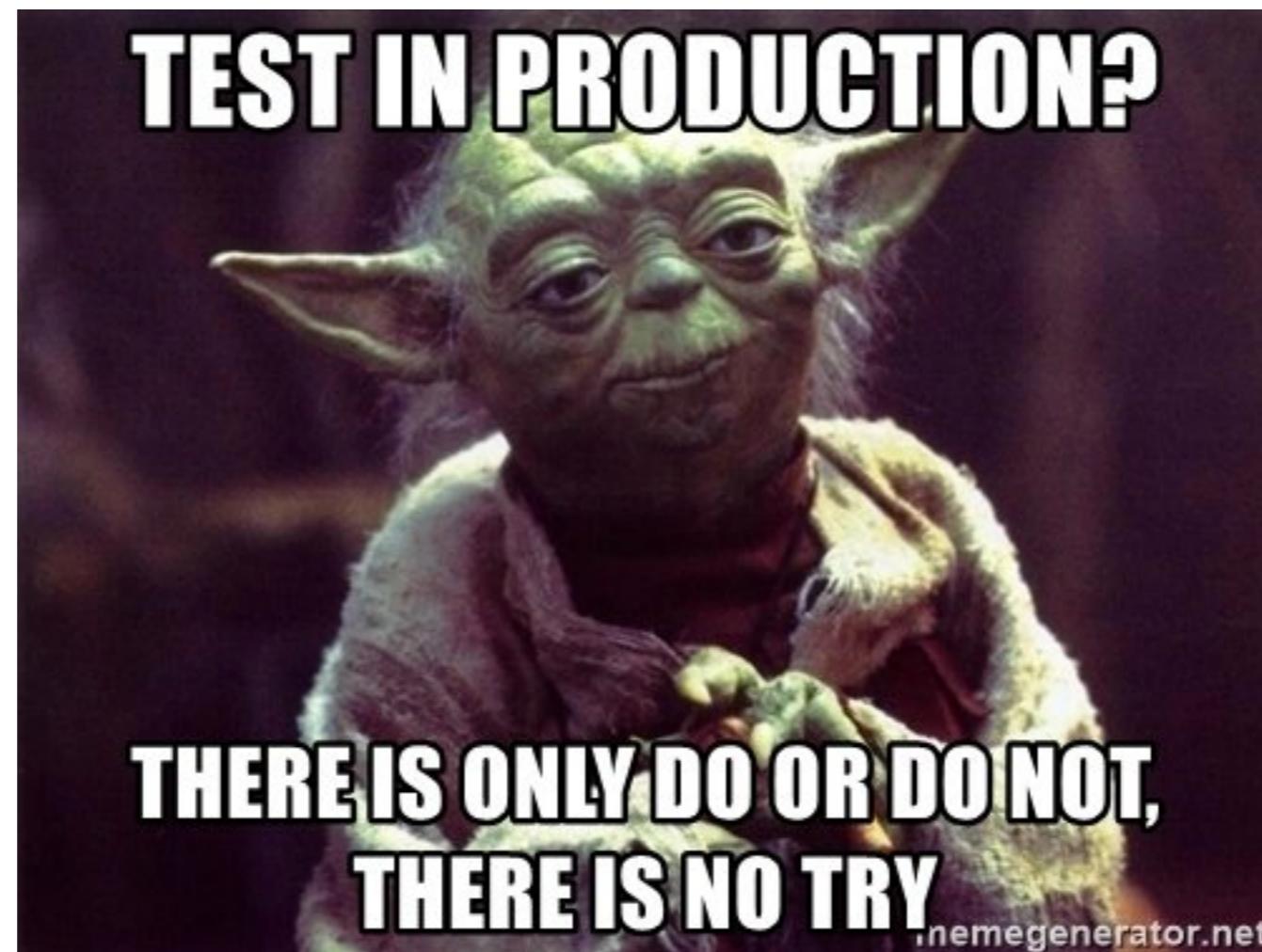
Keep the build **fast**

Continuous Integration is to provide rapid feedback



Practice 8

Test in clone of the **Production** environment



Practice 9

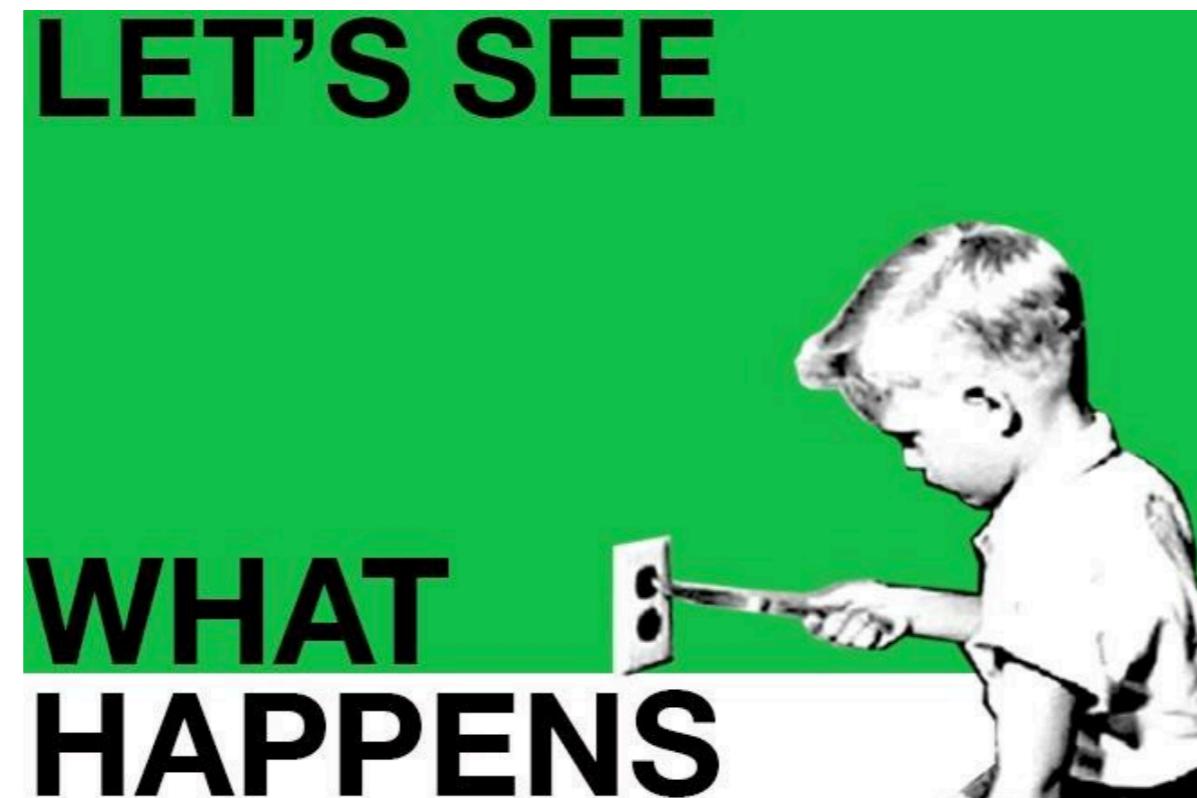
Make it easy for anyone to get
the latest executable

Make sure well known place where people can find



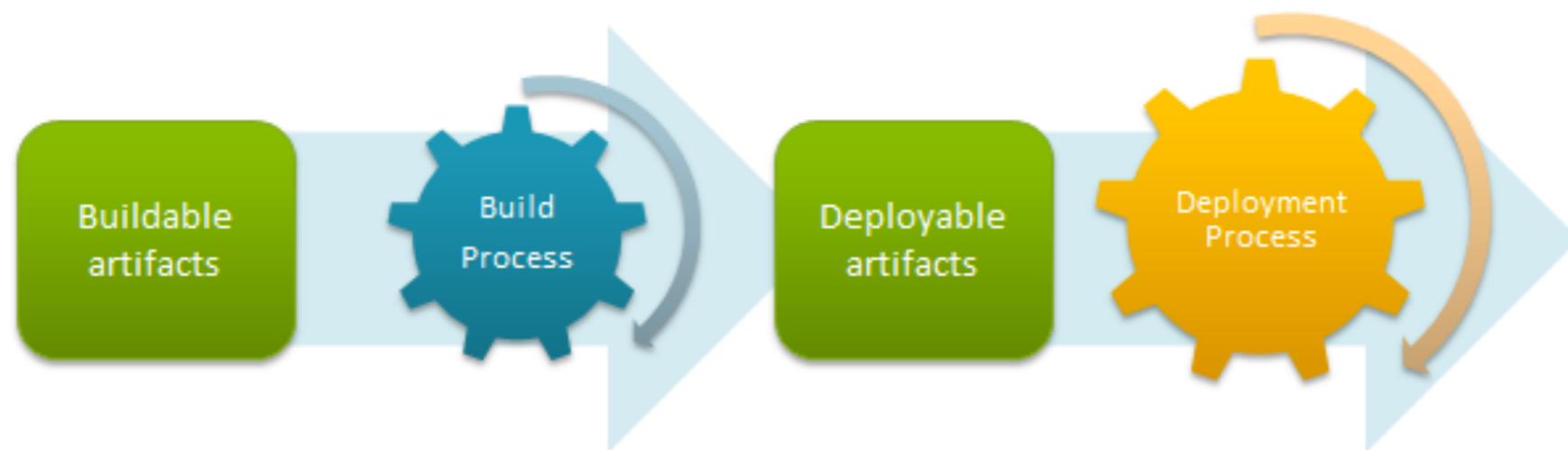
Practice 10

Everyone can see what's happening
Easier to see the state of the system and changes
Show the good information



Practice 11

Automated deployment



Continuous Delivery



Continuous Delivery

Use version control for all production artifacts

Automate your deployment process

Implement continuous integration (CI)

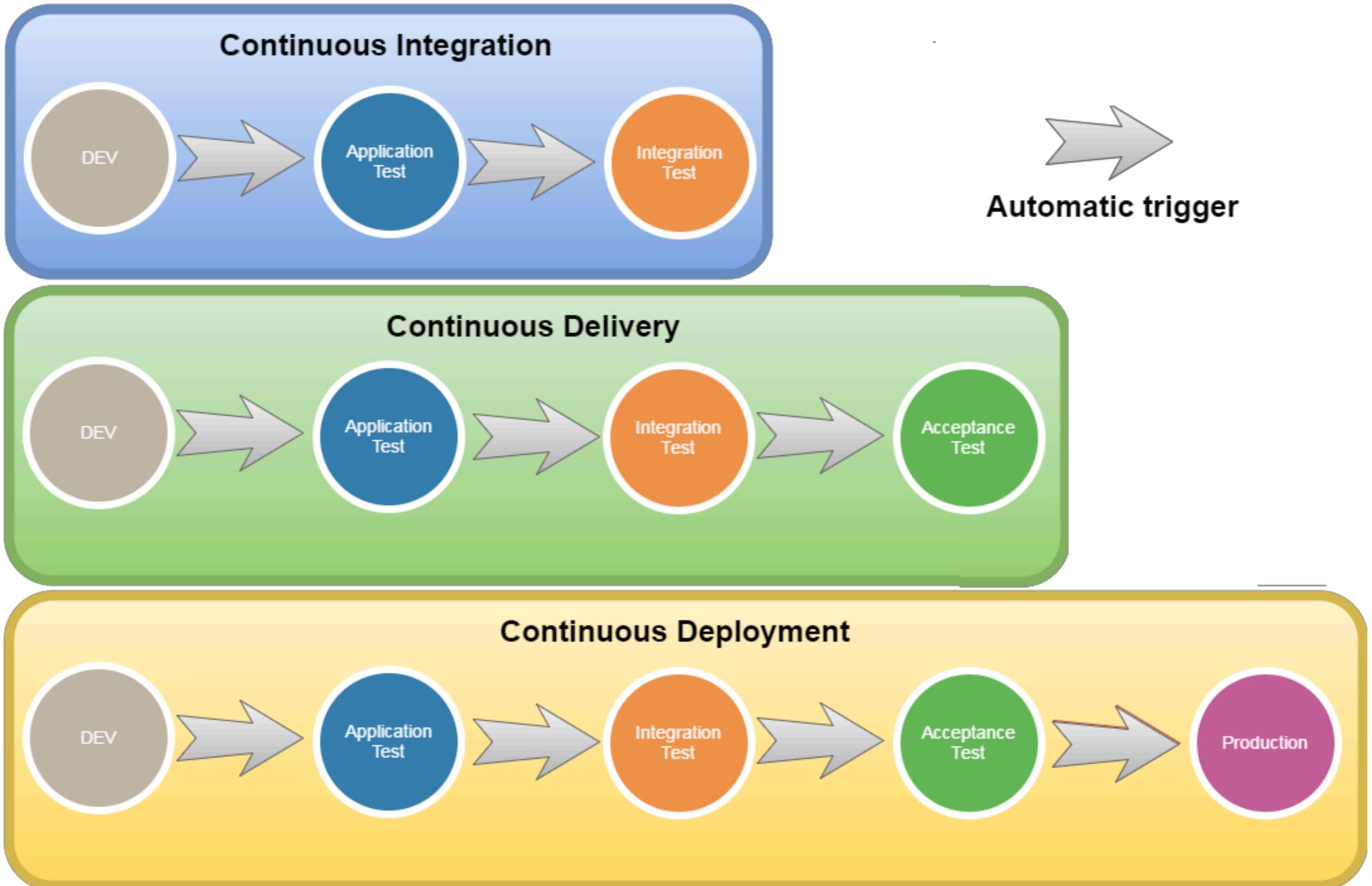
Use trunk-based development methods

Implement test automation

Support test data management

Integrate security into software development process





How to achieve the CI ?



1. Use good version control

Local
Centralize
Distributed



VSS = A brown pile of excrement with three wavy lines above it, representing a bad version control system.

JUST SAY NO!



2. Choose Branch strategy

Main only

Development isolation

Feature isolation

Release isolation

Service and Release isolation

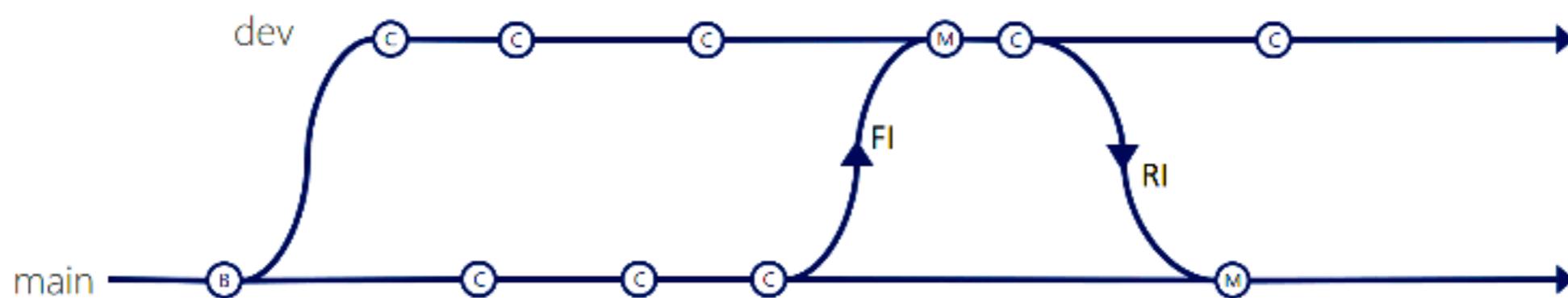
Service, Hotfix and Release isolation



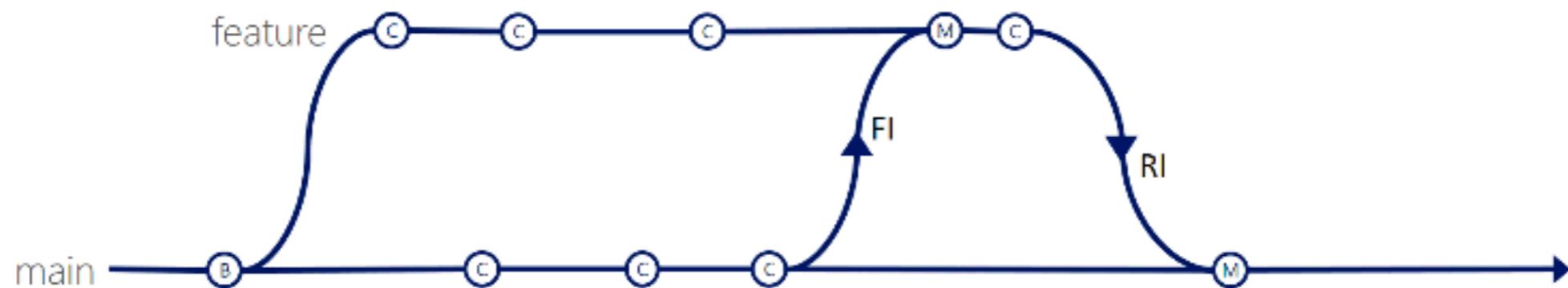
Main only



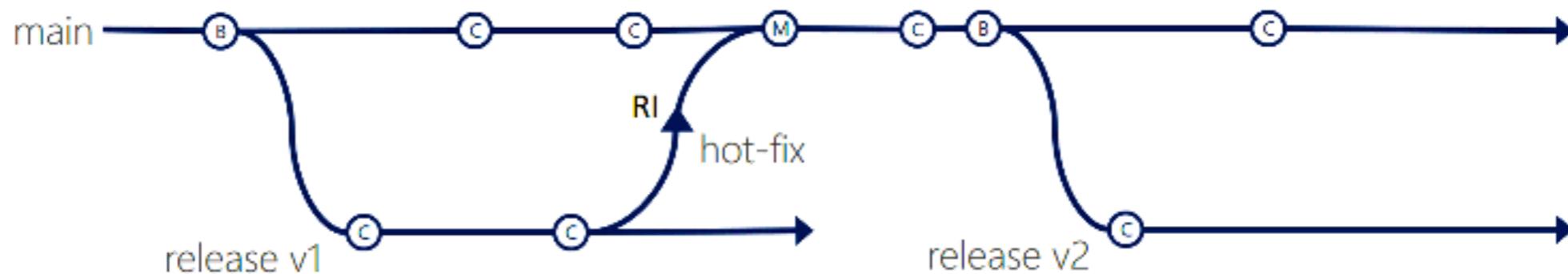
Development isolation



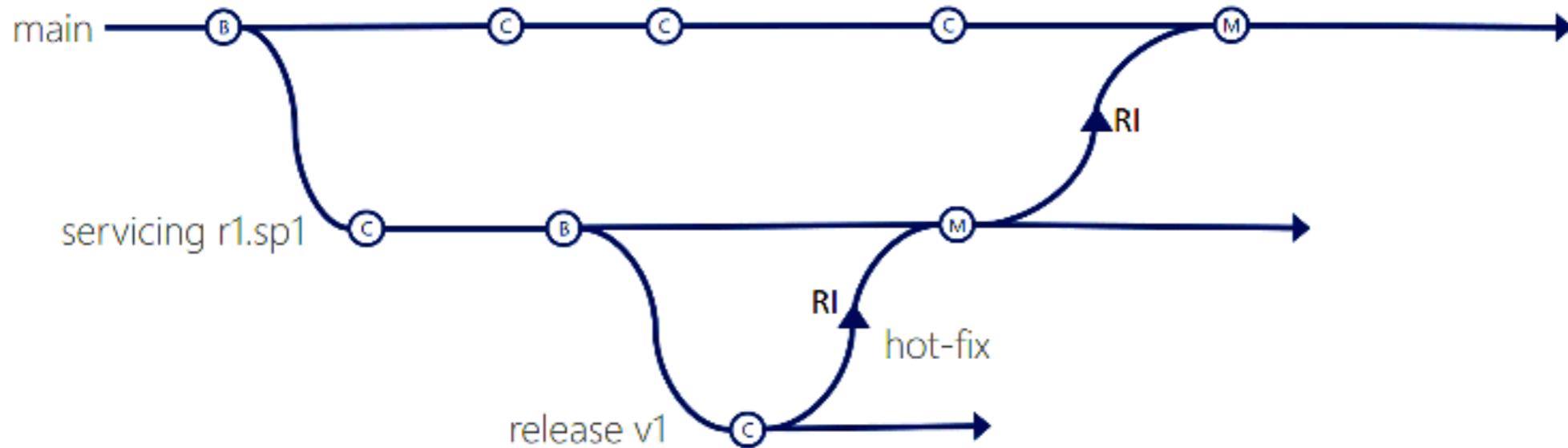
Feature isolation



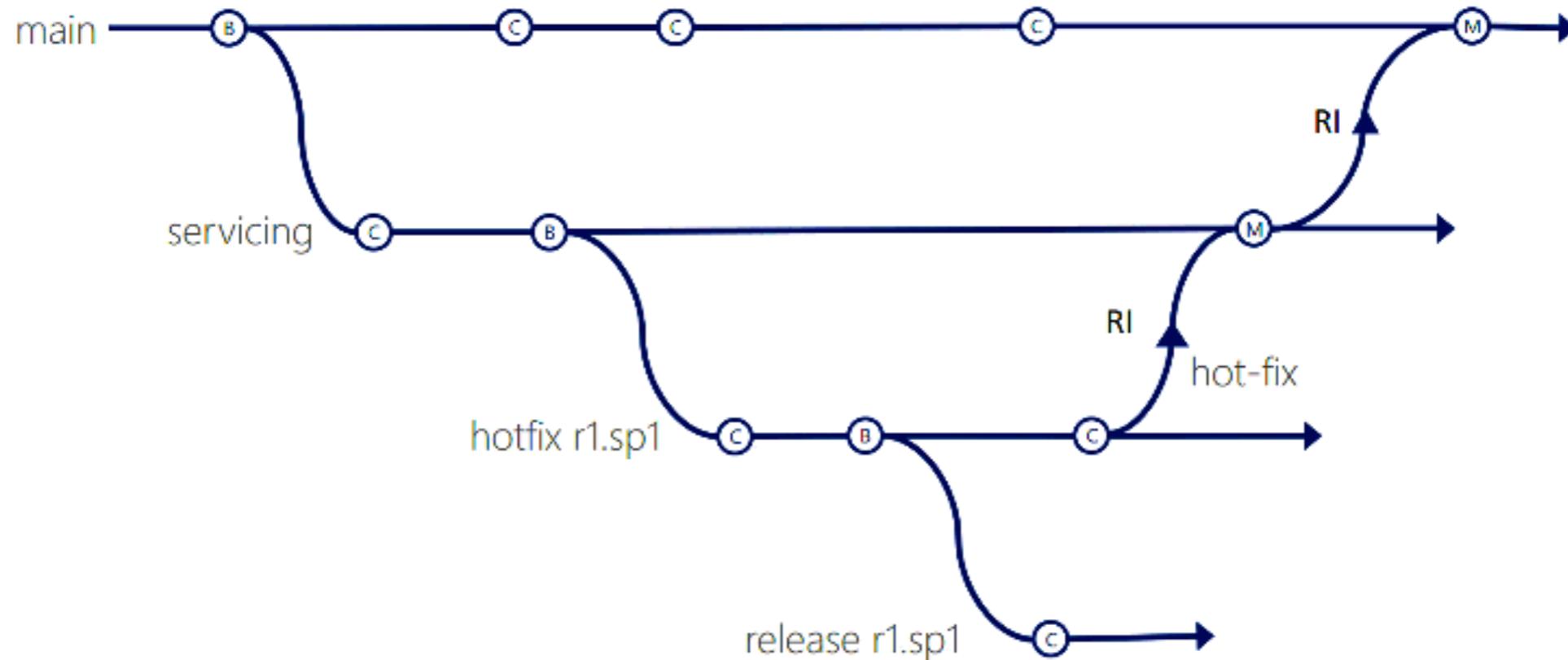
Release isolation



Service and Release isolation



Service, Hotfix, Release isolation



Validate, Validate and Validate

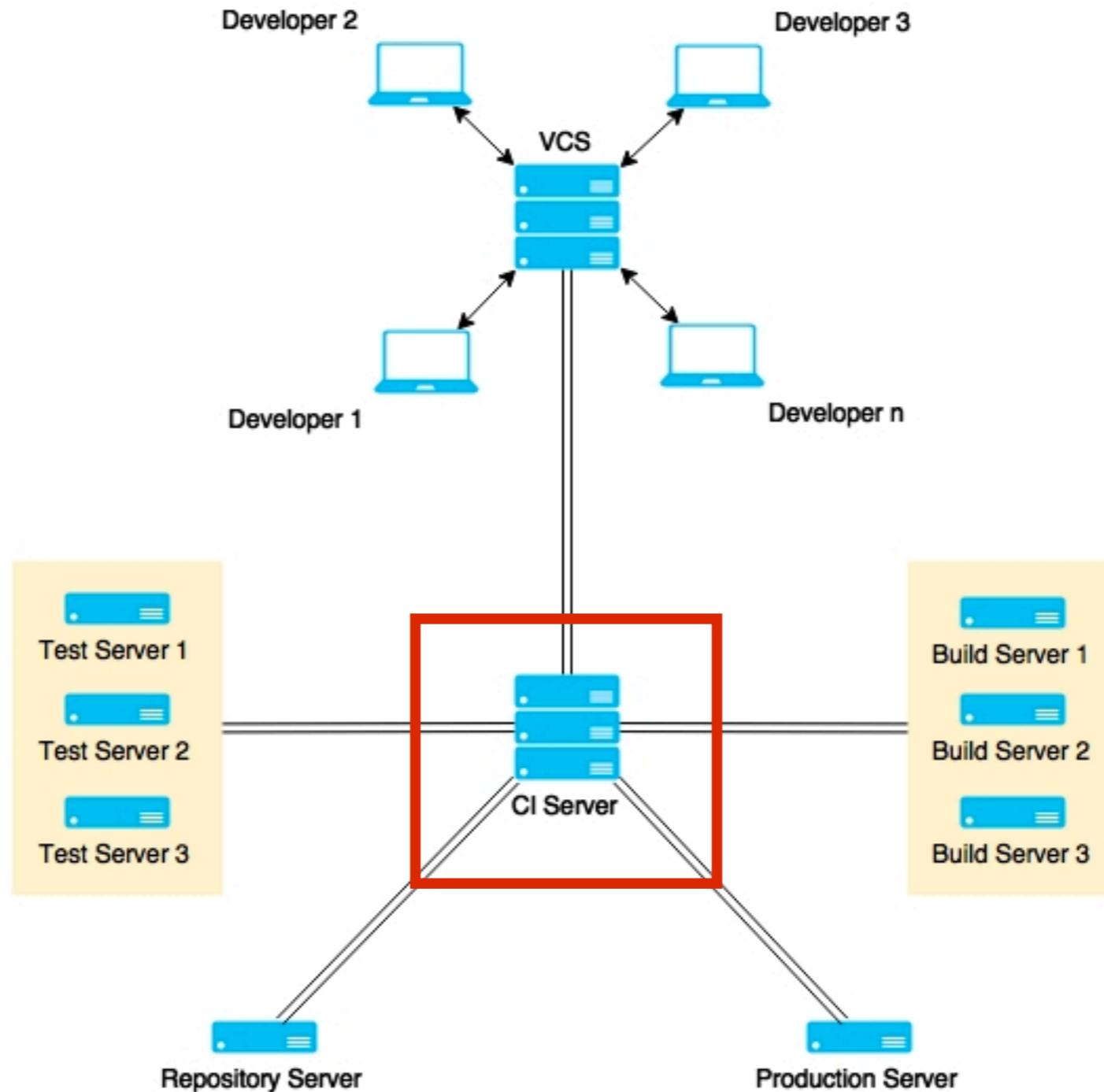


Suggestion

Keeping branches short-lived, merge conflicts are keep to as few as possible



3. Use good CI tool





Jenkins

Bamboo



TeamCity

> go™



Hudson



4. Use good build tool

- Javascript
 - Gulp, Grunt, Brocolli



- C#/.NET
 - Nant, MSBuild



- Java/JVM
 - Ant, Maven, Gradle, SBT, Leiningen



sbt gradle



More ...

Use static code analysis
Automated testing
Automated deployment
People discipline/habit



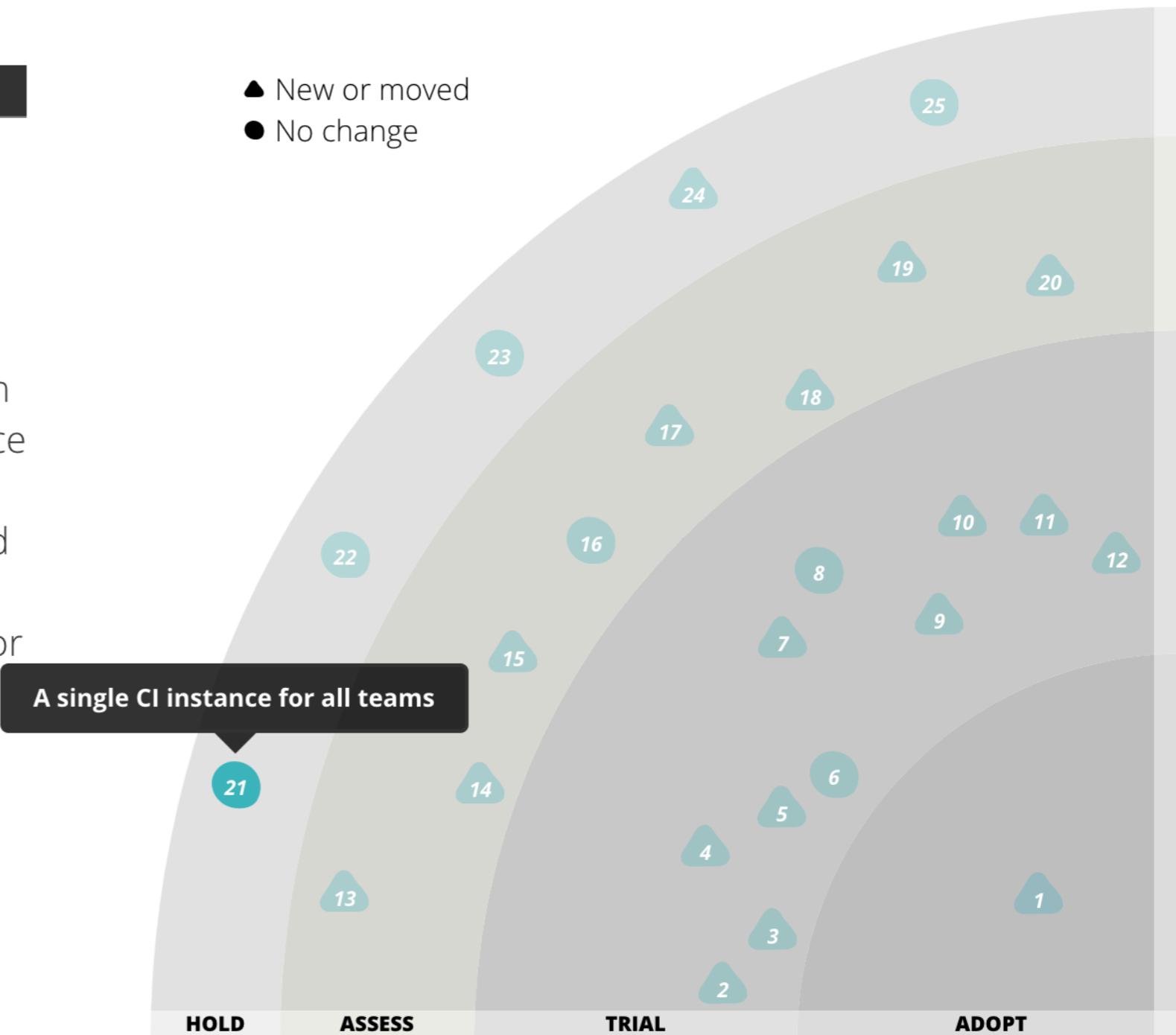
Anti-pattern for CI Server

● HOLD ?

21. A single CI instance for all teams

We're compelled to caution, again, against creating **a single CI instance for all teams**. While it's a nice idea in theory to consolidate and centralize Continuous Integration (CI) infrastructure, in reality we do not see enough maturity in the tools and products in this space to achieve the desired outcome. Software delivery teams which must use the centralized CI offering regularly have long delays depending on a central team to perform minor configuration tasks, or to troubleshoot problems in the shared infrastructure and tooling. At this stage, we continue to recommend that organizations limit their centralized investment to establishing patterns, guidelines and support for delivery teams to operate their own CI infrastructure.

- ▲ New or moved
- No change



<https://www.thoughtworks.com/radar/techniques>



Let's start with CI/CD



Application and framework to manage and monitor
the executable of **repeated tasks**



Jenkins

<https://jenkins.io/>



Centralize Continuous Integration Server



Jenkins

<https://jenkins.io/>



Why Jenkins ?

Easy !!
Extensible
Scalable
Opensource
Large community
Lot of plugins



Who use Jenkins ?

We thank the following organizations for their major commitments to support the Jenkins project.



Microsoft



redhat.

We thank the following organizations for their support of the Jenkins project through free and/or open source licensing programs.

Atlassian

Datadog

JFrog

Mac Cloud

PagerDuty

XMission

<https://jenkins.io/>



Hardware requirements

For Jenkins server

RAM +2GB

More CPU

More Disk



Installation



Jenkins in containers

Apache Tomcat

Jetty

JBoss

Websphere

WebLogic

Glassfish



Download Jenkins



The Jenkins logo features a cartoon character with a large head, wearing headphones and a bow tie, holding a coffee cup.

Jenkins

Build great things at any scale

The leading open source automation server, Jenkins provides hundreds of plugins to support building, deploying and automating any project.

[Documentation](#) [Download](#)

<https://jenkins.io/>



Use Long Term Support (LTS)

Getting started with Jenkins

The Jenkins project produces two release lines, LTS and weekly. Depending on your organization's needs, one may be preferred over the other.

Both release lines are distributed as `.war` files, native packages, installers, and Docker containers.

Long-term Support (LTS)

LTS (Long-Term Support) releases are chosen every 12 weeks from the stream of regular releases as the stable release for that time period. [Learn more...](#)

[Changelog](#) | [Upgrade Guide](#) | [Past Releases](#)

 [Deploy Jenkins 2.46.3](#)

 [Deploy to Azure](#)

 [Download Jenkins 2.46.3 for:](#)

Docker

FreeBSD

Weekly

A new release is produced weekly to deliver bug fixes and features to users and plugin developers.

[Changelog](#) | [Past Releases](#)

 [Download Jenkins 2.65 for:](#)

Arch Linux

Docker

FreeBSD

Gentoo



Start Jenkins

\$java -jar jenkins.war

Default port of server is **8080**

```
org.eclipse.jetty.server.AbstractConnector doStart  
ector@3e2fc448{HTTP/1.1, [http/1.1]}{0.0.0.0:8080}  
org.eclipse.jetty.server.Server doStart
```

```
winstone.Logger logInternal  
ngine v4.0 running: controlPort=disabled  
jenkins.InitReactorRunner$1 onAttained  
:ion  
jenkins.InitReactorRunner$1 onAttained  
jenkins.InitReactorRunner$1 onAttained
```



Change port of Jenkins

```
$java -jar jenkins.war --httpPort=<port>
```



Open in browser

<http://localhost:8080>

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/Users/somkiat/data/slide/ci-cd/swpark/software/keep/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue



Copy password from console

```
*****  
*****  
*****
```

Jenkins initial setup is required. An admin user has been created.

Please use the following password to proceed to installation:

a4b3a5231b8048419192d0c5afd3fce8

This may also be found at: /Users/somkiat/data/slide/ci-cd/swpi/initialAdminPassword

```
*****  
*****  
*****
```



Customize plugins

Getting Started



Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.46.3



Waiting

Getting Started

Getting Started

<input type="radio"/> Folders Plugin	<input type="radio"/> OWASP Markup Formatter Plugin	<input type="radio"/> build timeout plugin	<input type="radio"/> Credentials Binding Plugin
<input type="radio"/> Timestamper	<input type="radio"/> Workspace Cleanup Plugin	<input type="radio"/> Ant Plugin	<input type="radio"/> Gradle Plugin
<input type="radio"/> Pipeline	<input type="radio"/> GitHub Organization Folder Plugin	<input type="radio"/> Pipeline: Stage View Plugin	<input type="radio"/> Git plugin
<input type="radio"/> Subversion Plug-in	<input type="radio"/> SSH Slaves plugin	<input type="radio"/> Matrix Authorization Strategy Plugin	<input type="radio"/> PAM Authentication plugin
<input type="radio"/> LDAP Plugin	<input type="radio"/> Email Extension Plugin	<input type="radio"/> Mailer Plugin	

** - required dependency

Jenkins 2.46.3



Success

Getting Started

Installation Failures

Some plugins failed to install properly, you may retry installing them or continue with

✓ Folders Plugin	✓ OWASP Markup Formatter Plugin	✓ build timeout plugin	✓ Credentials Binding Plugin
✓ Timestamper	✓ Workspace Cleanup Plugin	✓ Ant Plugin	✓ Gradle Plugin
✓ Pipeline	✓ GitHub Organization Folder Plugin	✓ Pipeline: Stage View Plugin	✓ Git plugin
✓ Subversion Plug-in	✓ SSH Slaves plugin	✓ Matrix Authorization Strategy Plugin	✓ PAM Authentication plugin
✓ LDAP Plugin	✓ Email Extension Plugin	✓ Mailer Plugin	

Jenkins 2.46.3

[Continue](#)

[Retry](#)



Create a new user

Getting Started

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

Jenkins 2.46.3

[Continue as admin](#)

[Save and Finish](#)



Ready to use

Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Jenkins 2.46.3



Welcome to Jenkins

Jenkins

search [?](#) somkiat | log out

[ENABLE AUTO REFRESH](#)

New Item [add description](#)

People

Build History

Manage Jenkins

My Views

Credentials

Welcome to Jenkins!

Please [create new jobs](#) to get started.

Build Queue
No builds in the queue.

Build Executor Status
1 Idle
2 Idle

Page generated: Jun 14, 2017 2:08:57 PM ICT [REST API](#) [Jenkins ver. 2.46.3](#)



About Jenkins's HOME

Default of **JENKINS_HOME** is
`<path of user>/jenkins`



About Jenkins's HOME

Data in JENKINS_HOME

```
/Users/somkiat/.jenkins
├── config.xml
├── failed-boot-attempts.txt
├── hudson.model.UpdateCenter.xml
├── jenkins.CLI.xml
├── jenkins.install.UpgradeWizard.state
├── jobs
├── logs
├── nodeMonitors.xml
├── nodes
├── plugins
├── queue.xml
├── queue.xml.bak
├── secret.key
├── secret.key.not-so-secret
├── secrets
├── updates
├── userContent
├── users
└── war
```



About Jenkins's HOME

File and Folder name	Description
config.xml	All about configuration
jobs	Keep all jobs/project
plugins	Keep all plugins
nodes	Keep all nodes
logs	Keep all logs



Change Jenkins's HOME

For Windows

```
set JENKINS_HOME=<your path>
```

For Linux/Mac

```
export JENKINS_HOME=<your path>
```

try to restart Jenkins ...



Disable Jenkins's security



Set useSecurity=false

<JENKINS HOME>/config.xml

```
<?xml version='1.0' encoding='UTF-8'?>
<hudson>
    <disabledAdministrativeMonitors/>
    <version>2.89.3</version>
    <numExecutors>2</numExecutors>
    <mode>NORMAL</mode>
    <useSecurity>false</useSecurity>
    <authorizationStrategy class="hudson.security.LoggedInAuthorizationStrategy">
        <denyAnonymousReadAccess>true</denyAnonymousReadAccess>
    </authorizationStrategy>
```



Learn to use Jenkins in the right way



Manage Jenkins

For Administrator to config anything in Jenkins

Manage Jenkins

- [Configure System](#)
Configure global settings and paths.
- [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
- [Configure Credentials](#)
Configure the credential providers and types
- [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
- [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modify configuration files.
- [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- [System Information](#)
Displays various environmental information to assist trouble-shooting.
- [System Log](#)
System log captures output from `java.util.logging` related to Jenkins.
- [Load Statistics](#)
Check your resource utilization and see if you need more computers for your builds.



Configure System

Global setting and paths

Jenkins search ?

New Item

People

Build History

Manage Jenkins

My Views

Credentials

New View

Build Queue
No builds in the queue.

Build Executor Status
1 Idle
2 Idle

Manage Jenkins

-  [Configure System](#)
Configure global settings and paths.
-  [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
-  [Configure Credentials](#)
Configure the credential providers and types
-  [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
-  [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modify configuration files.
-  [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
-  [System Information](#)
Displays various environmental information to assist trouble-shooting.
-  [System Log](#)
System log captures output from `java.util.logging` related to Jenkins.
-  [Load Statistics](#)
Check your resource utilization and see if you need more computers for your builds.



Configure System

JENKINS Home

of executors

Label name of node

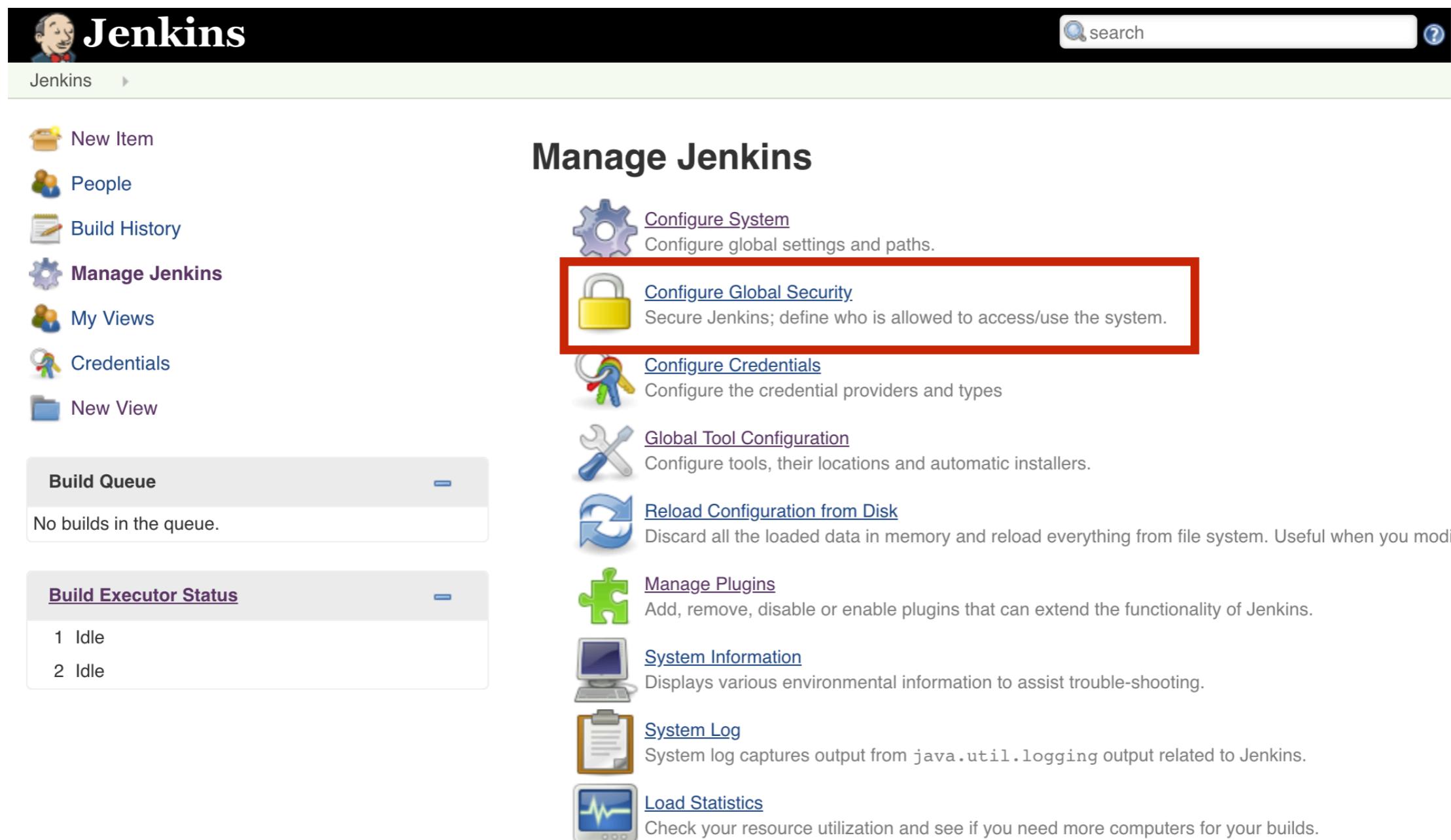
Environment variables

Email notification



Configure Global Security

Setting for secure Jenkins



The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links like 'New Item', 'People', 'Build History', 'Manage Jenkins' (which is selected), 'My Views', 'Credentials', and 'New View'. Below that are sections for 'Build Queue' (empty) and 'Build Executor Status' (2 Idle). The main content area is titled 'Manage Jenkins' and lists several configuration options: 'Configure System' (gear icon), 'Configure Global Security' (padlock icon, highlighted with a red box), 'Configure Credentials' (key icon), 'Global Tool Configuration' (wrench icon), 'Reload Configuration from Disk' (refresh icon), 'Manage Plugins' (puzzle piece icon), 'System Information' (monitor icon), 'System Log' (clipboard icon), and 'Load Statistics' (ECG icon). The 'Configure Global Security' item is described as 'Secure Jenkins; define who is allowed to access/use the system.'

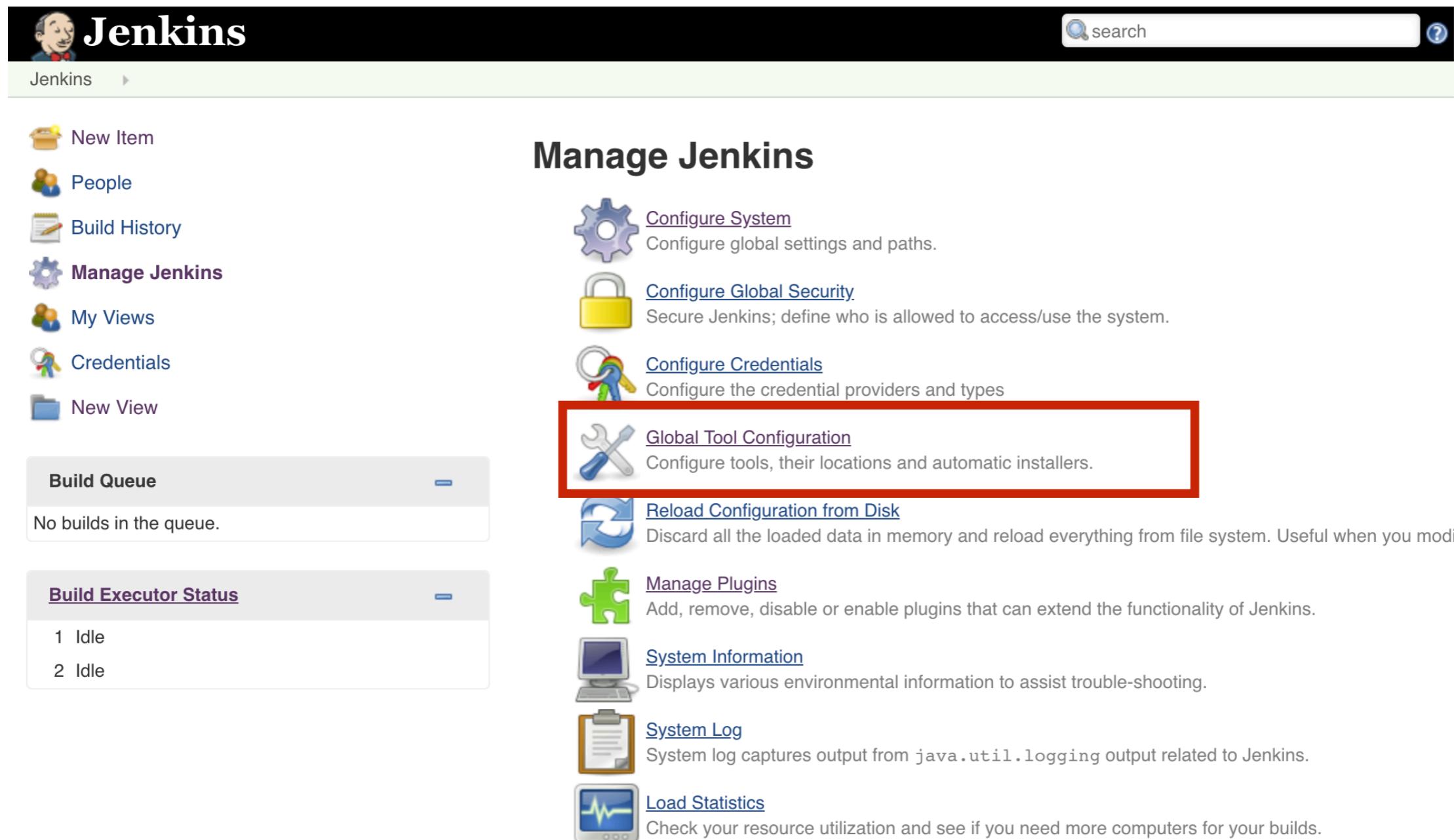
Manage Jenkins

-  [Configure System](#)
Configure global settings and paths.
-  [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
-  [Configure Credentials](#)
Configure the credential providers and types
-  [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
-  [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modify configuration files.
-  [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
-  [System Information](#)
Displays various environmental information to assist trouble-shooting.
-  [System Log](#)
System log captures output from `java.util.logging` related to Jenkins.
-  [Load Statistics](#)
Check your resource utilization and see if you need more computers for your builds.



Global Tool Configuration

Config tools, location and automatic installers



The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links like New Item, People, Build History, Manage Jenkins (which is selected and highlighted in purple), My Views, Credentials, and New View. Below that are sections for Build Queue (No builds in the queue) and Build Executor Status (1 Idle, 2 Idle). The main content area is titled "Manage Jenkins" and lists several configuration options: Configure System, Configure Global Security, Configure Credentials, Global Tool Configuration (which is highlighted with a red box), Reload Configuration from Disk, Manage Plugins, System Information, System Log, and Load Statistics.

Manage Jenkins

-  [Configure System](#)
Configure global settings and paths.
-  [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
-  [Configure Credentials](#)
Configure the credential providers and types
-  [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
-  [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modify configuration files.
-  [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
-  [System Information](#)
Displays various environmental information to assist trouble-shooting.
-  [System Log](#)
System log captures output from `java.util.logging` related to Jenkins.
-  [Load Statistics](#)
Check your resource utilization and see if you need more computers for your builds.



Global Tool Configuration

Apache Maven
JDK (Java Development Kit)
Git
Gradle
Docker



Manage Plugins

Add, remove, enable/disable plugins

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links like New Item, People, Build History, Manage Jenkins (which is selected and highlighted in purple), My Views, Credentials, and New View. Below that are sections for Build Queue (No builds in the queue) and Build Executor Status (1 Idle, 2 Idle). The main content area is titled "Manage Jenkins" and contains several configuration links: Configure System, Configure Global Security, Configure Credentials, Global Tool Configuration, Reload Configuration from Disk, Manage Plugins (which is highlighted with a red box), System Information, System Log, and Load Statistics.

Manage Jenkins

- [Configure System](#)
Configure global settings and paths.
- [Configure Global Security](#)
Secure Jenkins; define who is allowed to access/use the system.
- [Configure Credentials](#)
Configure the credential providers and types
- [Global Tool Configuration](#)
Configure tools, their locations and automatic installers.
- [Reload Configuration from Disk](#)
Discard all the loaded data in memory and reload everything from file system. Useful when you modi
- [Manage Plugins](#)
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- [System Information](#)
Displays various environmental information to assist trouble-shooting.
- [System Log](#)
System log captures output from `java.util.logging` related to Jenkins.
- [Load Statistics](#)
Check your resource utilization and see if you need more computers for your builds.



Manage Plugins

Add, remove, enable/disable plugins

				Filter: <input type="text"/>
Updates	Available	Installed	Advanced	
Install	Name ↓	Version	Installed	
No updates				

Update information obtained: 14 hr ago

[Check now](#)

Select: [All](#), [None](#)

This page lists updates to the plugins you currently use.



Manage Plugins

Filter plugins

Filter:

Updates Available Installed Advanced

Install	Name ↓	Version	Installed
		No updates	

Update information obtained: 14 hr ago [Check now](#)

Select: [All](#), [None](#)

This page lists updates to the plugins you currently use.



Finds Jenkins's plugin

Plugins Index

Discover the 1000+ community contributed Jenkins plugins to support building, deploying and automating any project.

Browse ▶ Find plugins...

<https://plugins.jenkins.io/>



Try to install a first plugin

Choose Available tab and select a plugin

Filter:

Updates Available Installed Advanced

Install ↓

Name	Version
CCM Plug-in <input type="checkbox"/> This plug-in generates the trend report for CCM, an open source static code analysis program.	3.1
FxCop Runner plugin <input type="checkbox"/>	1.1
MSBuild Plugin <input type="checkbox"/>	1.27
MSTest plugin <input type="checkbox"/> Generates test reports for MSTest.	0.19
MSTestRunner plugin <input type="checkbox"/>	1.3.0
NAnt Plugin <input type="checkbox"/>	1.4.3
NCover plugin <input type="checkbox"/>	0.3
PowerShell plugin <input type="checkbox"/>	1.3
Violation Comments to Bitbucket Server Plugin <input type="checkbox"/> Finds violations reported by code analyzers and comments Bitbucket Server (or Stash) pull requests (or commits) with them.	1.50
Violations plugin <input type="checkbox"/>	0.7.11

[Install without restart](#) [Download now and install after restart](#)

Update information obtained: 9 hr 37 min ago [Check now](#)



Manage Nodes

Add, remove, status of nodes

Jenkins  Somkiat Puisungnoen | log out

ENABLE AUTO REFRESH

Back to Dashboard Manage Jenkins New Node Configure

Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

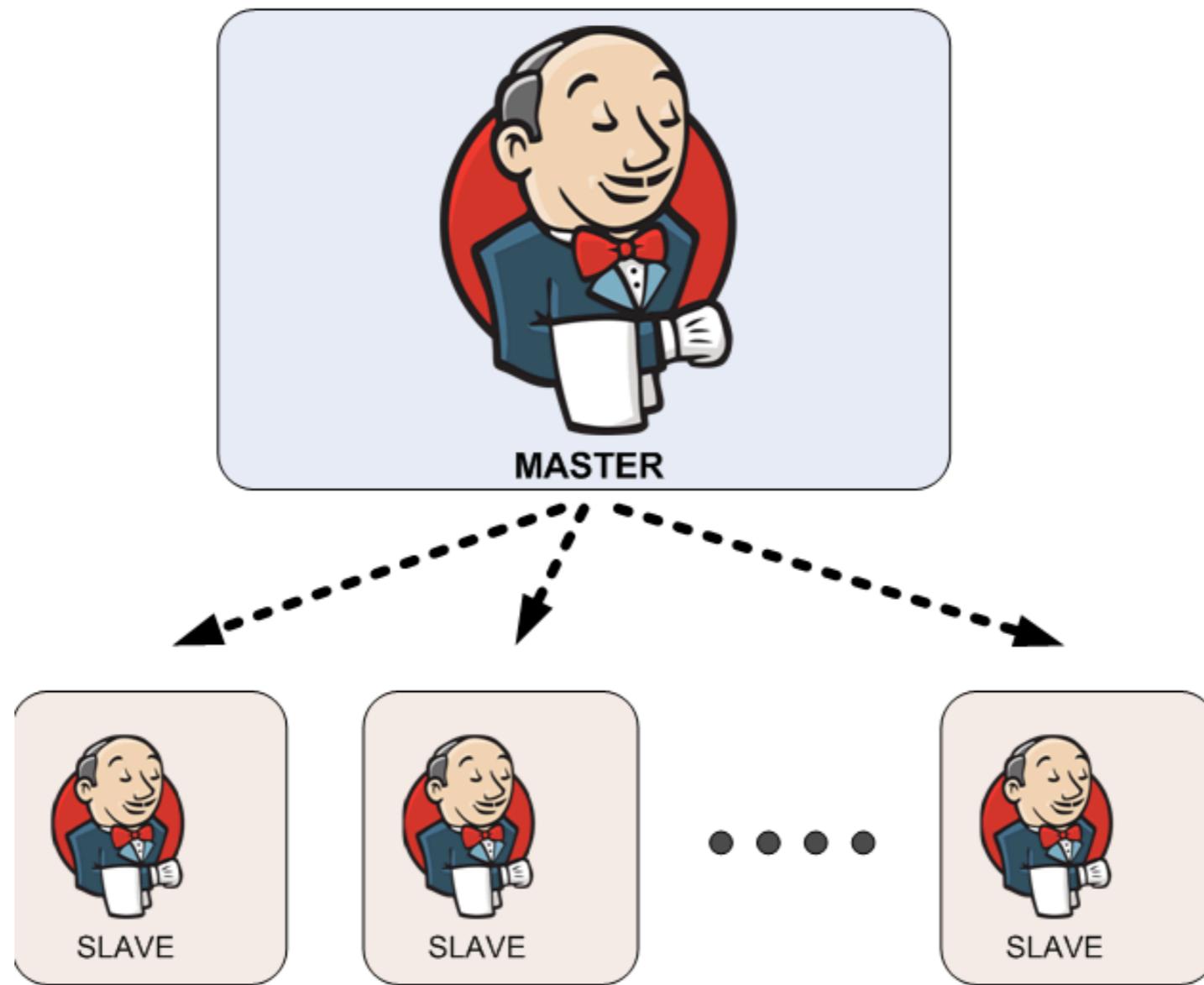
S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Mac OS X (x86_64)	In sync	5.73 GB	463.00 MB	5.73 GB	0ms
	Data obtained	36 min	36 min	36 min	36 min	36 min	36 min

Refresh status



Manage Nodes

Master-slave concept to scale Jenkins

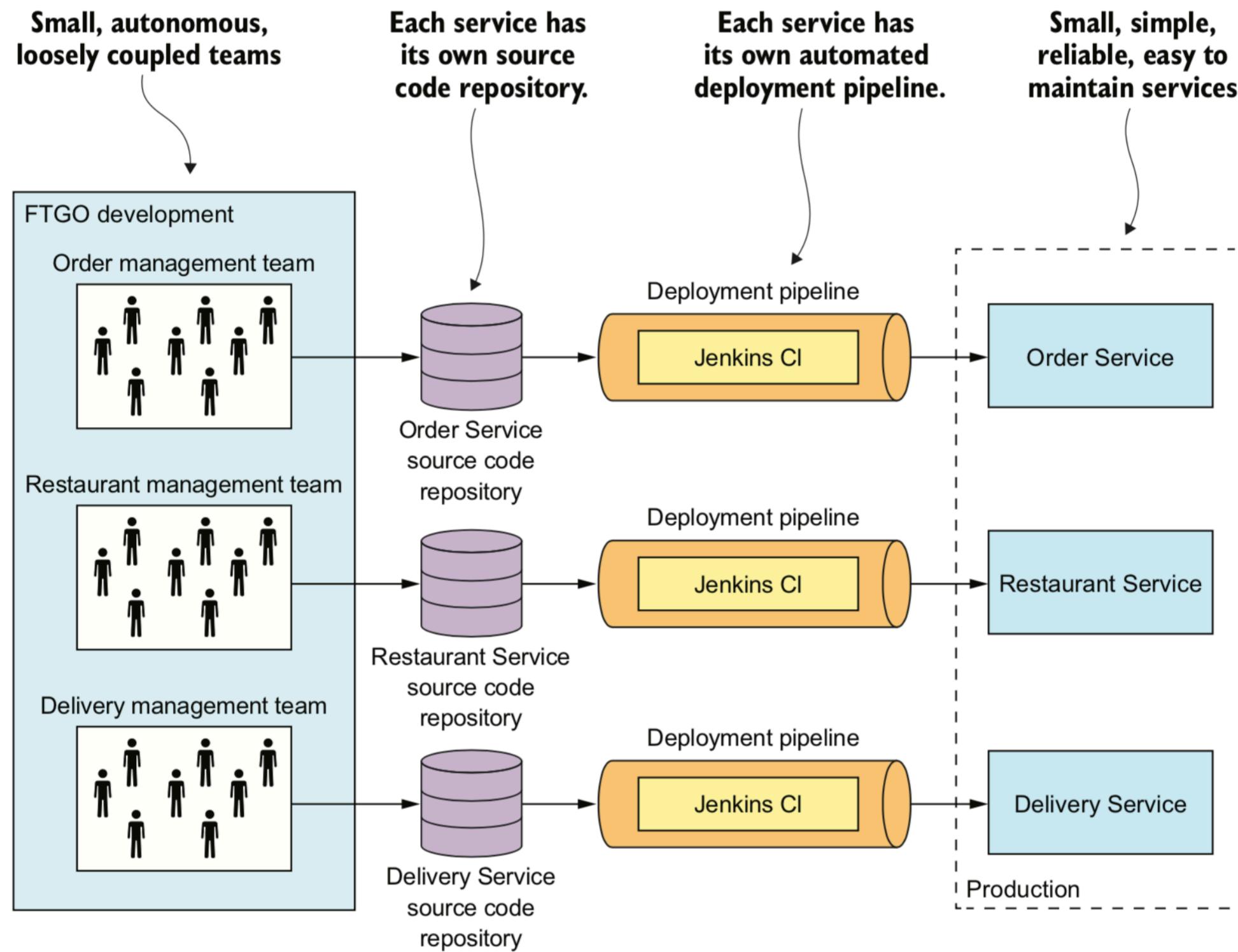


Workshop

Deployment pipeline



Deployment pipeline in each service



Twelve-Factor App



Twelve-Factor App

<https://12factor.net/>

Initial to build app for Heroku

Principles to cloud and container native app



1. Codebase

Codebase must be tracked in version control
and will have many deploy



2. Dependencies

Dependencies are explicitly declared and isolated

Use dependency management tools to shared
libraries



3. Configuration

Store configuration in the environment

Add configuration in environment variables or config files



4. Backing services

Treat backing services as an attached resource

Should easy to deploy and change



5. Build, Release, Run

Always have a build and deploy strategy

Build strategies for repeated builds, versioning of running system and rollback



6. Processes

Execute the application as a stateless process



7. Port Binding

Expose services via port bindings



8. Concurrency

Scale out with the process model



9. Disposability

Quick application startup and shutdown times
Graceful shutdown



10. Dev/prod parity

Application is treated the same way in dev, staging and production

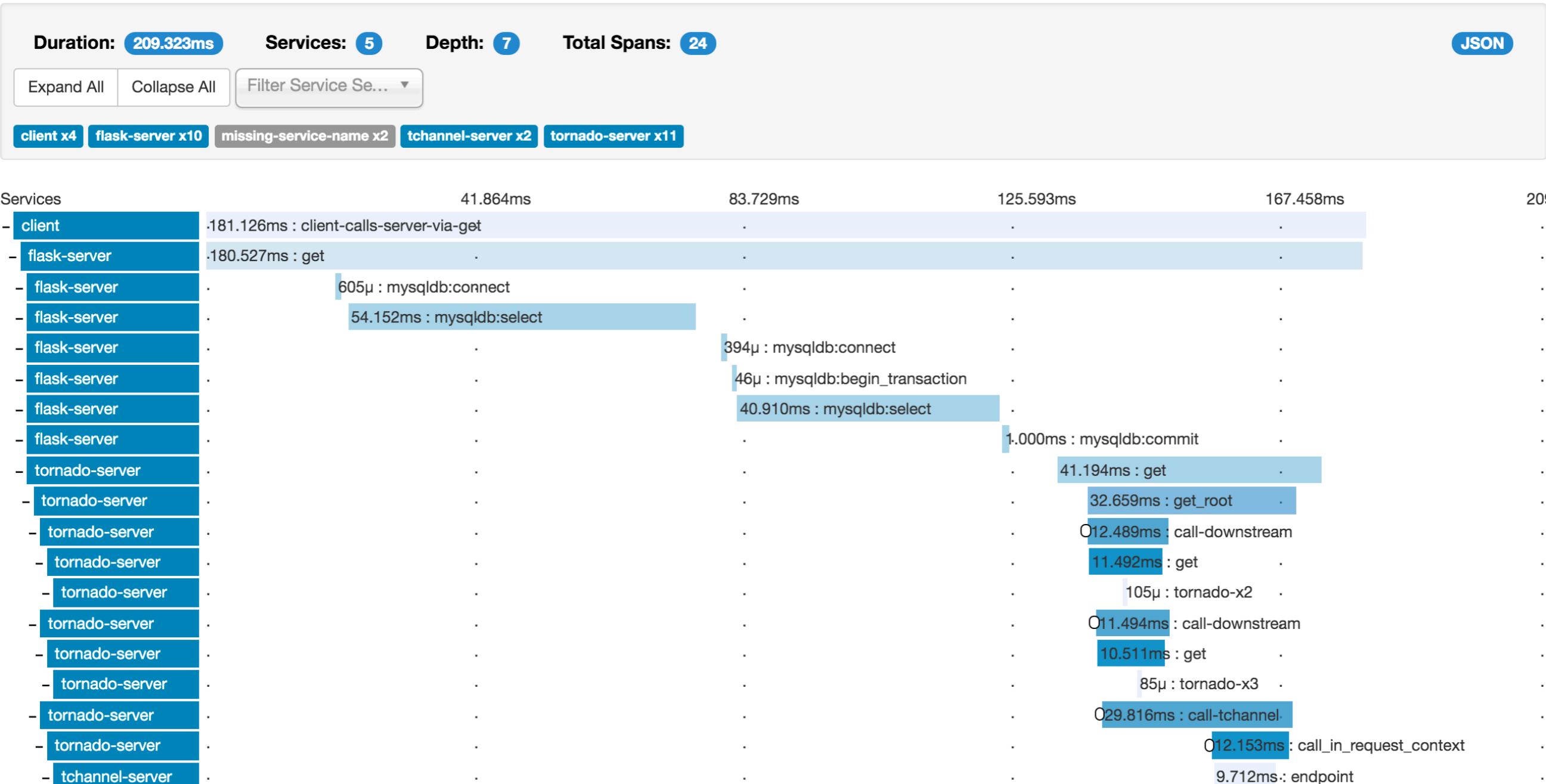


11. Log management

Treated as an event stream



Logging/Tracing



<https://zipkin.io/>



12. Admin tasks

Treated the same way like the rest of the application



Summary



Let's start with good monolith



Module 1

Module 2

Module 3

Module 4

Module 5

Module 6



Find your problem



Module 1

Module 2

Module 3

Module 4

Module 5

Module 6



Module 1

Module 2

Module 3

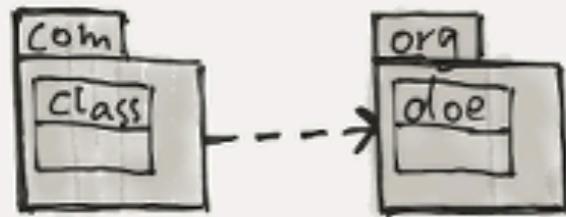
Module 5

Module 6

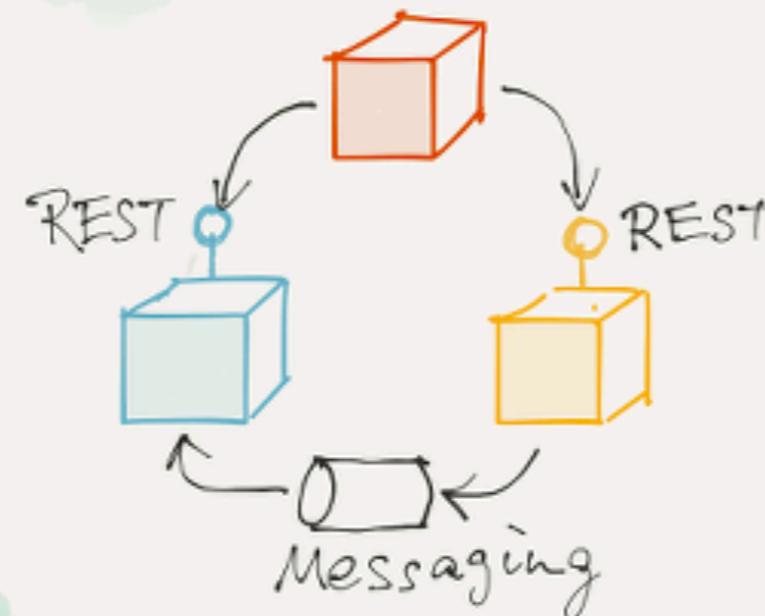
Module 4



Architecture



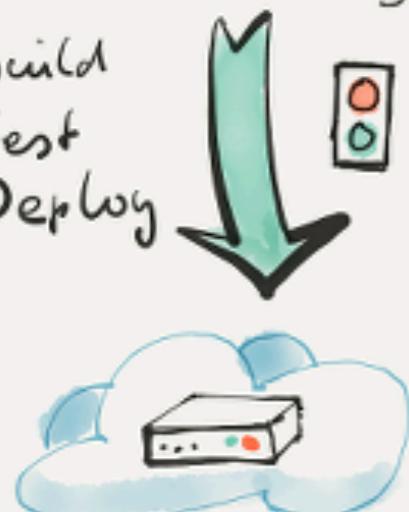
Microservices



Deployment

Continuous Delivery

`{ var i=1; }`
Build
Test
Deploy



Infrastructure

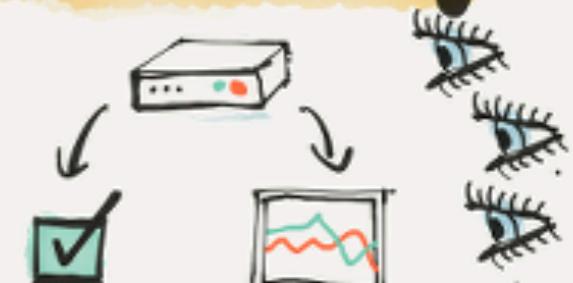


People & Teams



Communication
Collaboration

Monitoring



Features & Technology

