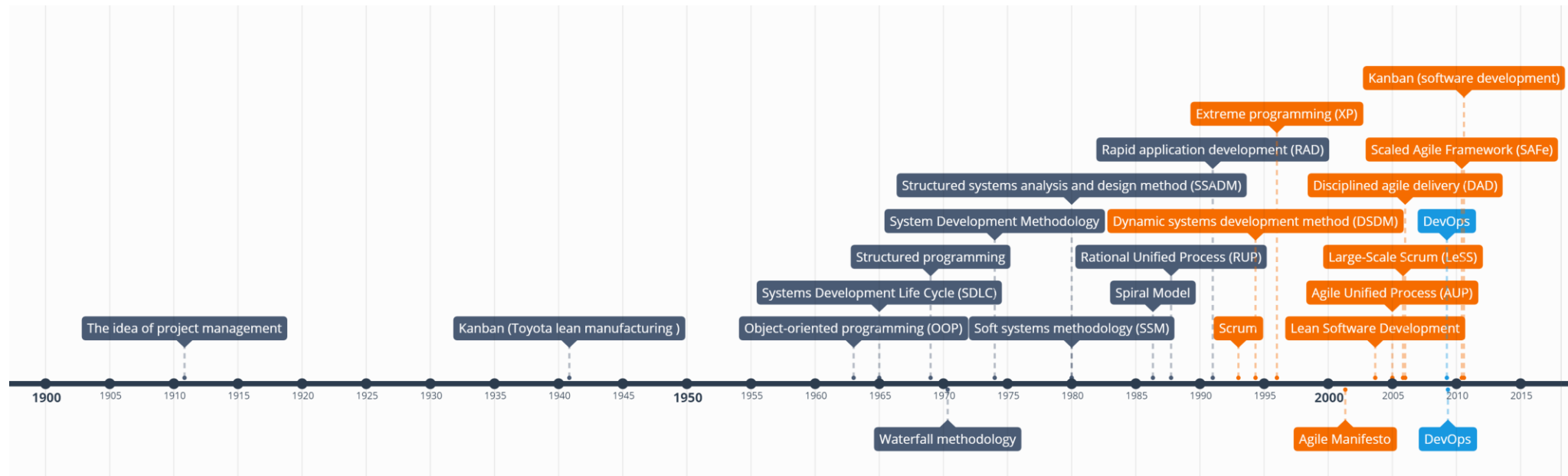


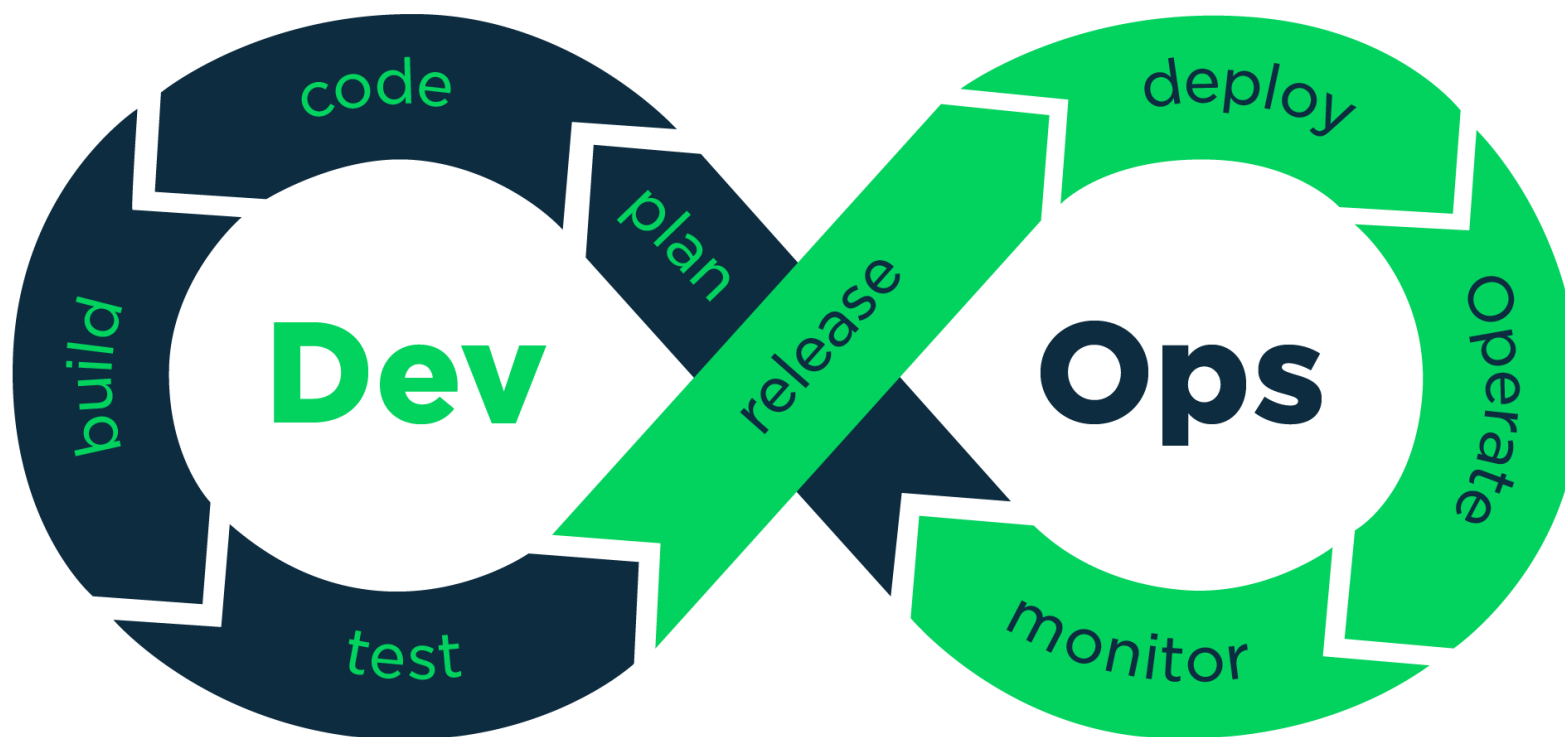
# AGILE? METHODOLOGIES?: DEVOPS

Lecturer:  
Ehsan Shoja

# SOFTWARE DEVELOPMENT PROCESS HISTORY



# DEVOPS



*“A software development method that emphasizes communication, collaboration, integration, automation, and measurement of cooperation between software developers and other IT professionals”*



# DEVOPS

- ❑ DevOps is more than just development and operations teams working together. It's more than tools and practices. DevOps is a mindset, a cultural shift, where teams adopt new ways of working.
- ❑ A DevOps culture means developers get closer to the user by gaining a better understanding of user requirements and needs. Operations teams get involved in the development process and add maintenance requirements and customer needs.
- ❑ It means adhering key principles that help DevOps teams deliver applications and services at a faster pace and higher quality than organizations using the traditional software development model.

# DEVOPS PRINCIPLES: COLLABORATION

- ❑ The key premise behind DevOps is collaboration. Development and operations teams coalesce into a functional team that communicates, shares feedback, and collaborates throughout the entire development and deployment cycle. Often, this means development and operations teams merge into a single team that works across the entire application lifecycle.
- ❑ The members of a DevOps team are responsible for ensuring quality deliverables across each facet of the product. This leads to more 'full stack' development, where teams own the complete backend-to-frontend responsibilities of a feature or product. Teams will own a feature or project throughout the complete lifecycle from idea to delivery. This enhanced level of investment and attachment from the team leads to higher quality output.

# DEVOPS PRINCIPLES: AUTOMATION

- ❑ An essential practice of DevOps is to automate as much of the software development lifecycle as possible. This gives developers more time to write code and develop new features. Automation is a key element of a CI/CD pipeline and helps to reduce human errors and increase team productivity. With automated processes, teams achieve continuous improvement with short iteration times, which allows them to quickly respond to customer feedback.

# DEVOPS PRINCIPLES: CONTINUOUS IMPROVEMENT

❑ Continuous improvement was established as a staple of agile practices, as well as lean manufacturing and Improvement Kata. It's the practice of focusing on experimentation, minimizing waste, and optimizing for speed, cost, and ease of delivery. Continuous improvement is also tied to continuous delivery, allowing DevOps teams to continuously push updates that improve the efficiency of software systems. The constant pipeline of new releases means teams consistently push code changes that eliminate waste, improve development efficiency, and bring more customer value.

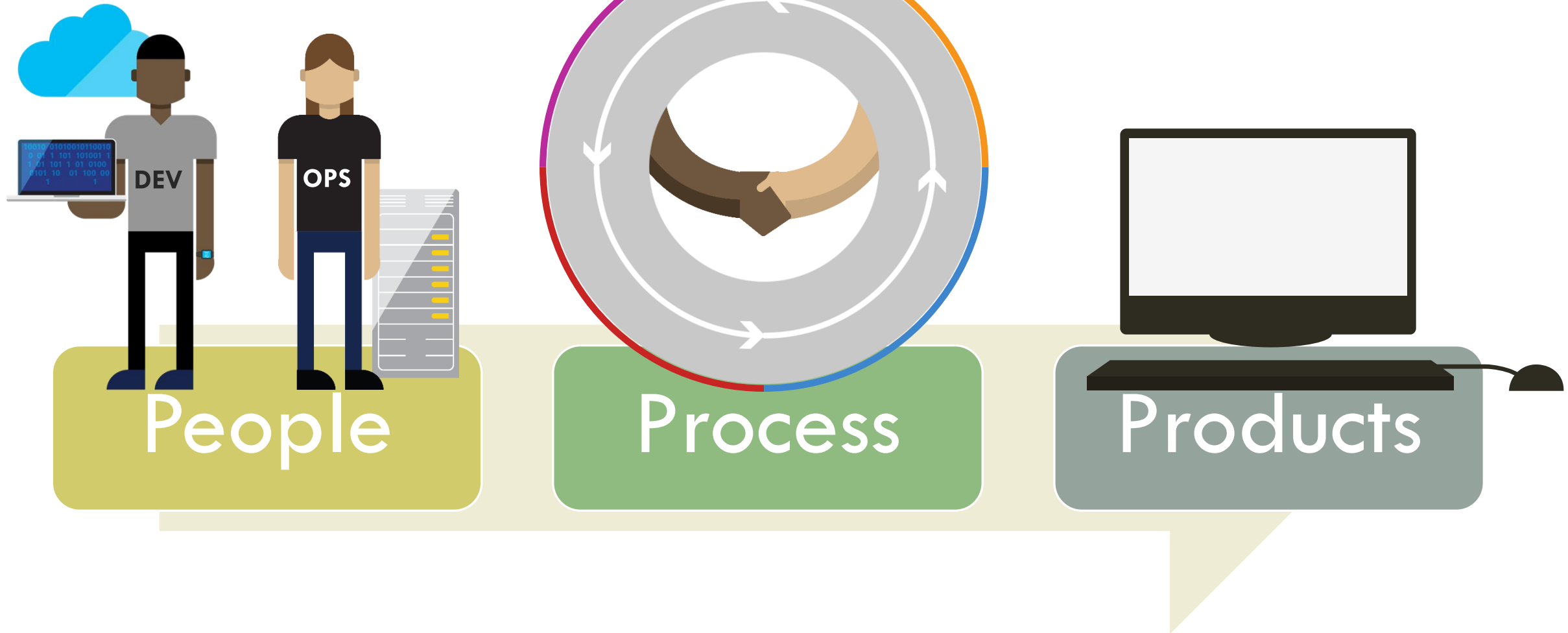
# DEVOPS PRINCIPLES: CUSTOMER-CENTRIC ACTION

- DevOps teams use short feedback loops with customers and end users to develop products and services centered around user needs. DevOps practices enable rapid collection and response to user feedback through use of real-time live monitoring and rapid deployment. Teams get immediate visibility into how live users interact with a software system and use that insight to develop further improvements.

# DEVOPS PRINCIPLES: CUSTOMER-CENTRIC ACTION

- DevOps teams use short feedback loops with customers and end users to develop products and services centered around user needs. DevOps practices enable rapid collection and response to user feedback through use of real-time live monitoring and rapid deployment. Teams get immediate visibility into how live users interact with a software system and use that insight to develop further improvements.

# LIFECYCLE OF DEVOPS



# DEVOPS PROCESS

1 Plan

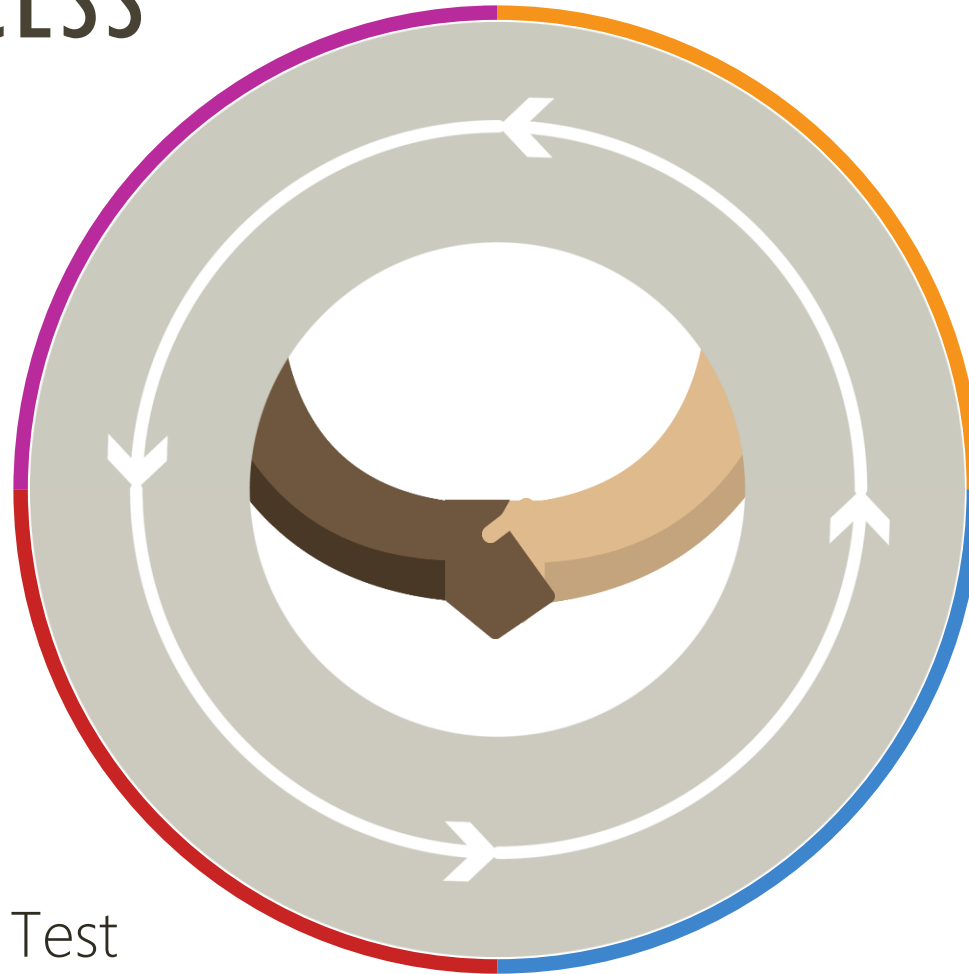
4 Monitor + Learn

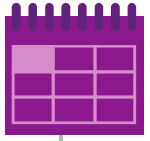
Development

Production

2 Develop + Test

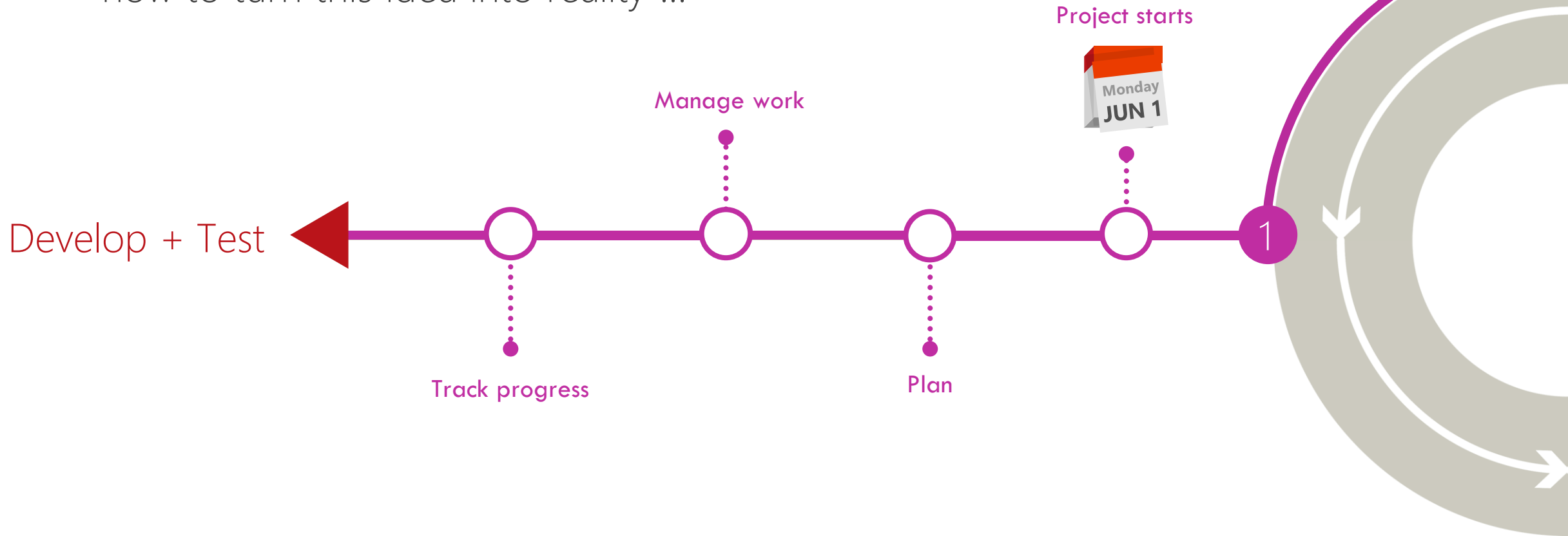
3 Release





# Plan

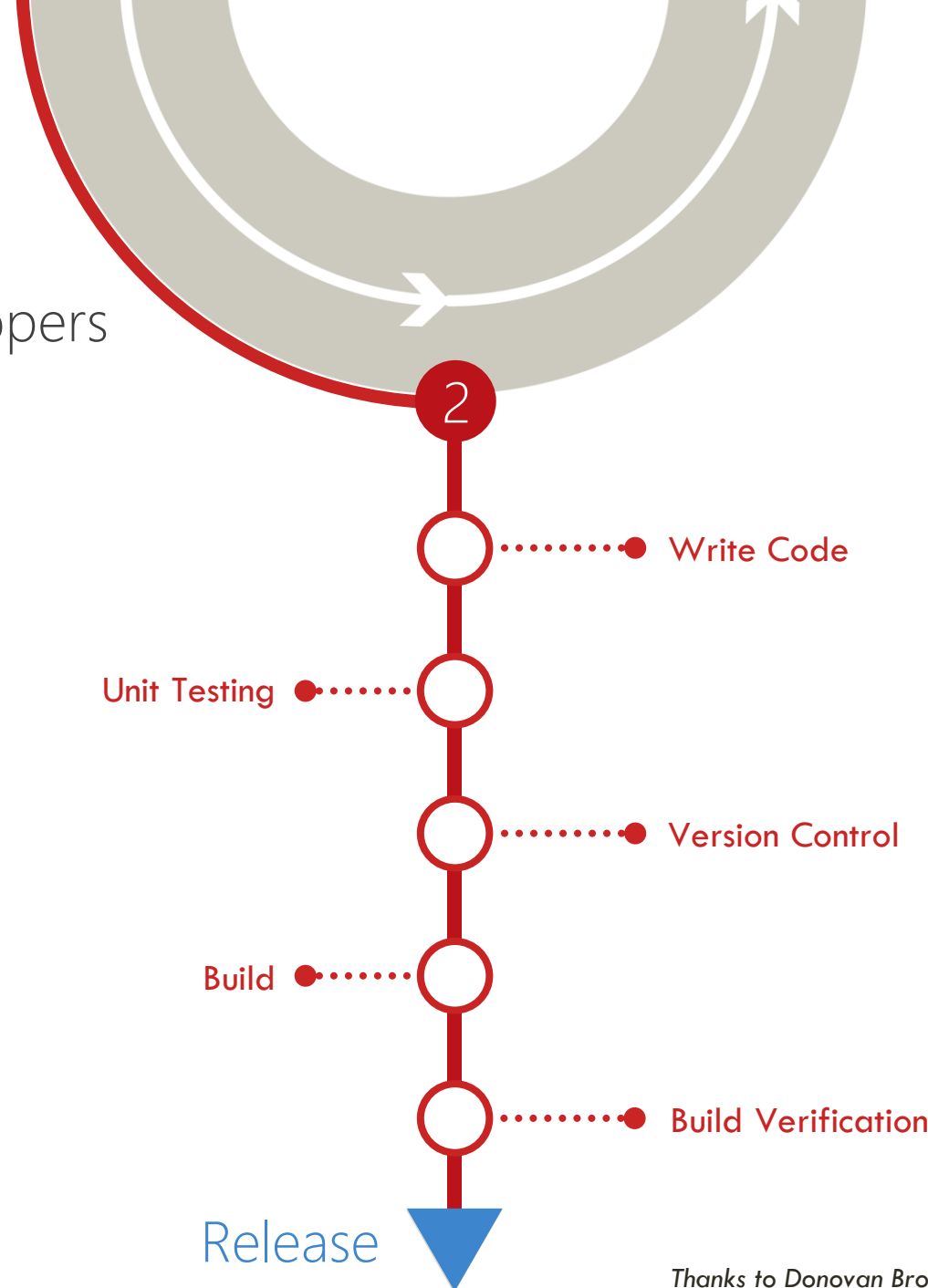
It starts with an idea – and a plan  
how to turn this idea into reality ...





## Develop + Test

Once the iteration starts, developers turn great ideas into features ...

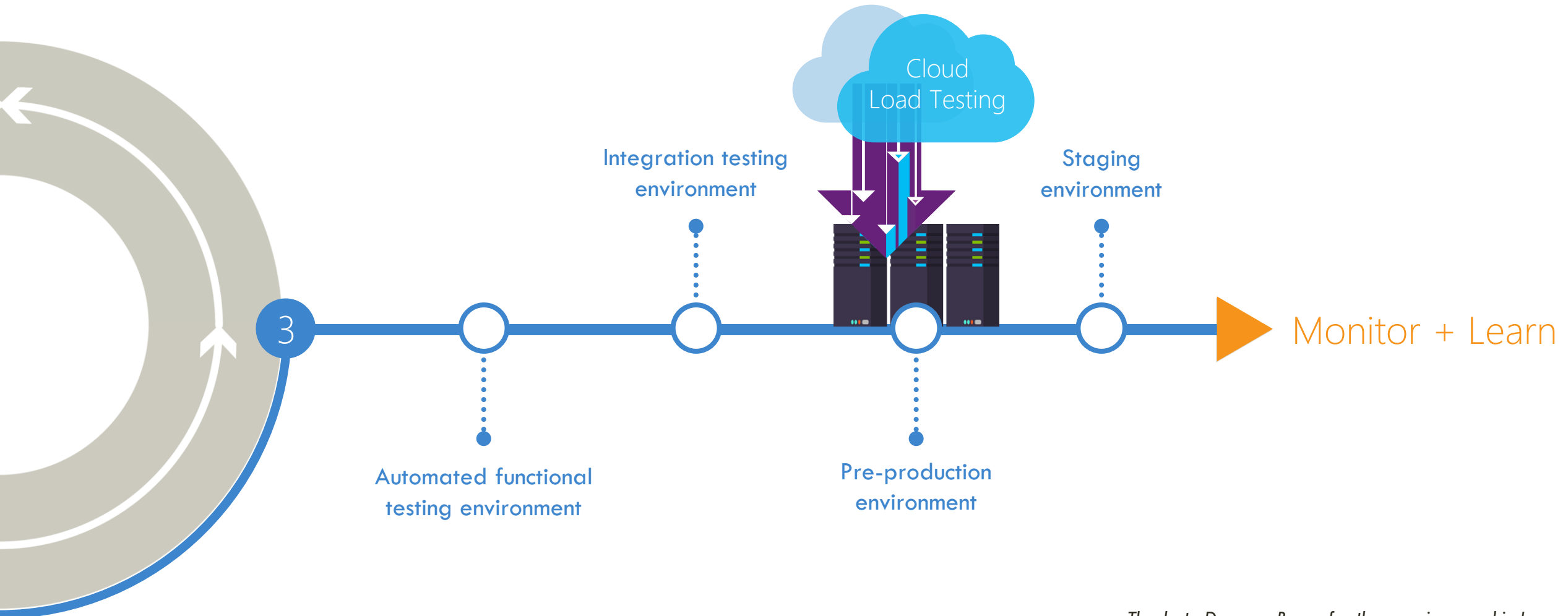


*Thanks to Donovan Brown for the amazing graphics!*



# Release

When all tests pass, the build is deployed to testing environments for each stage in the release process

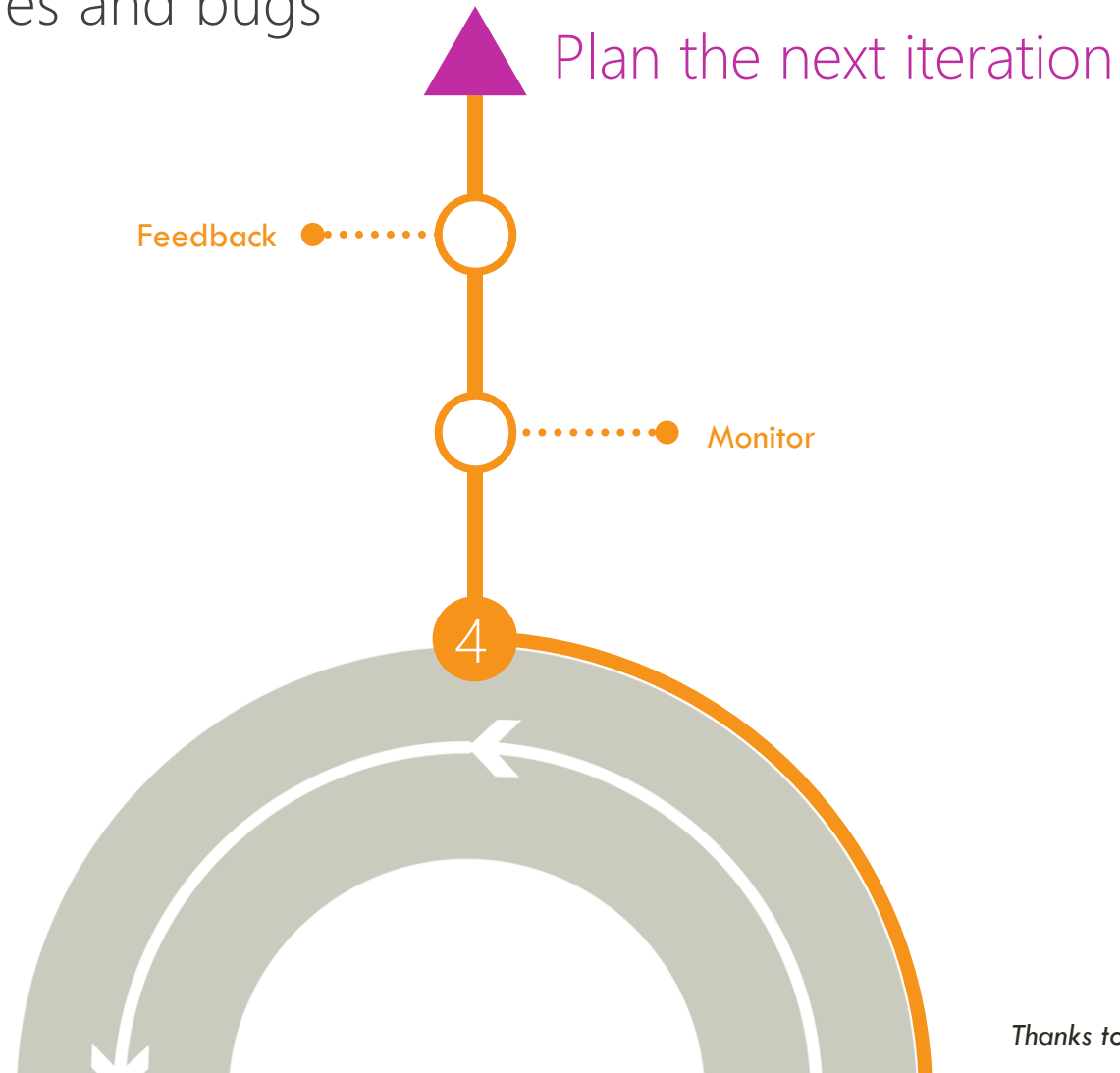


Thanks to Donovan Brown for the amazing graphics!



## Monitor + Learn

Learn and understand how users use your app, how it reacts and quickly fix issues and bugs



*Thanks to Donovan Brown for the amazing graphics!*

# DEVOPS PRACTICES

Continuous integration

Continuous Deployment / Delivery

Infrastructure as Code

Automated testing

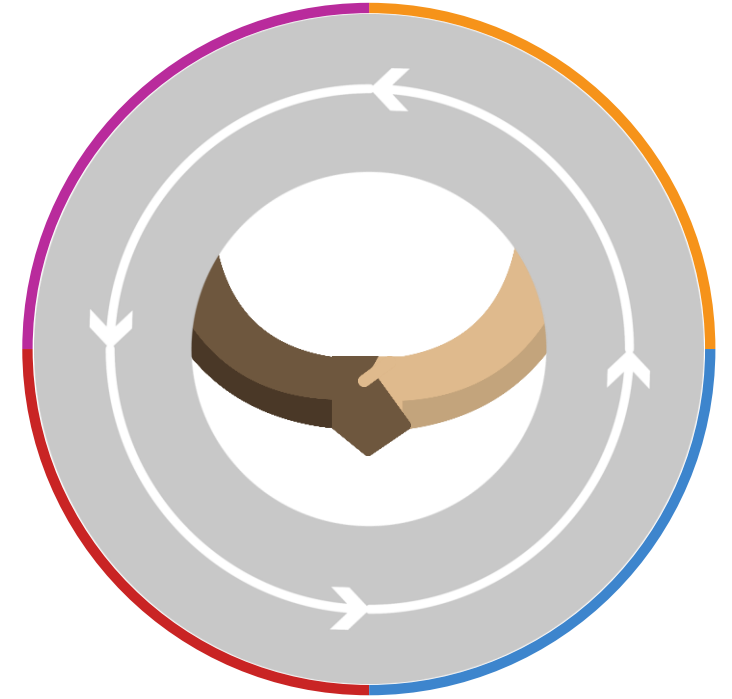
Application Performance Monitoring / Management

Release Management

Configuration Management

Doing one of these practices doesn't mean you are doing DevOps!

Source: <http://www.itproguys.com/devops-practices>



# DEVOPS PRACTICES: CONTINUOUS INTEGRATION & CONTINUOUS DELIVERY

## ❑ Continuous Integration

- ❑ Continuous integration is a software development practice where developers regularly merge their code changes into a central repository, after which automated builds and tests are run. The key goals of continuous integration are to find and address bugs quicker, improve software quality, and reduce the time it takes to validate and release new software updates.



## ❑ Continuous Delivery

- ❑ Continuous delivery is a software development practice where code changes are automatically built, tested, and prepared for a release to production. It expands upon continuous integration by deploying all code changes to a testing environment and/or a production environment after the build stage. When continuous delivery is implemented properly, developers will always have a deployment-ready build artifact that has passed through a standardized test process.



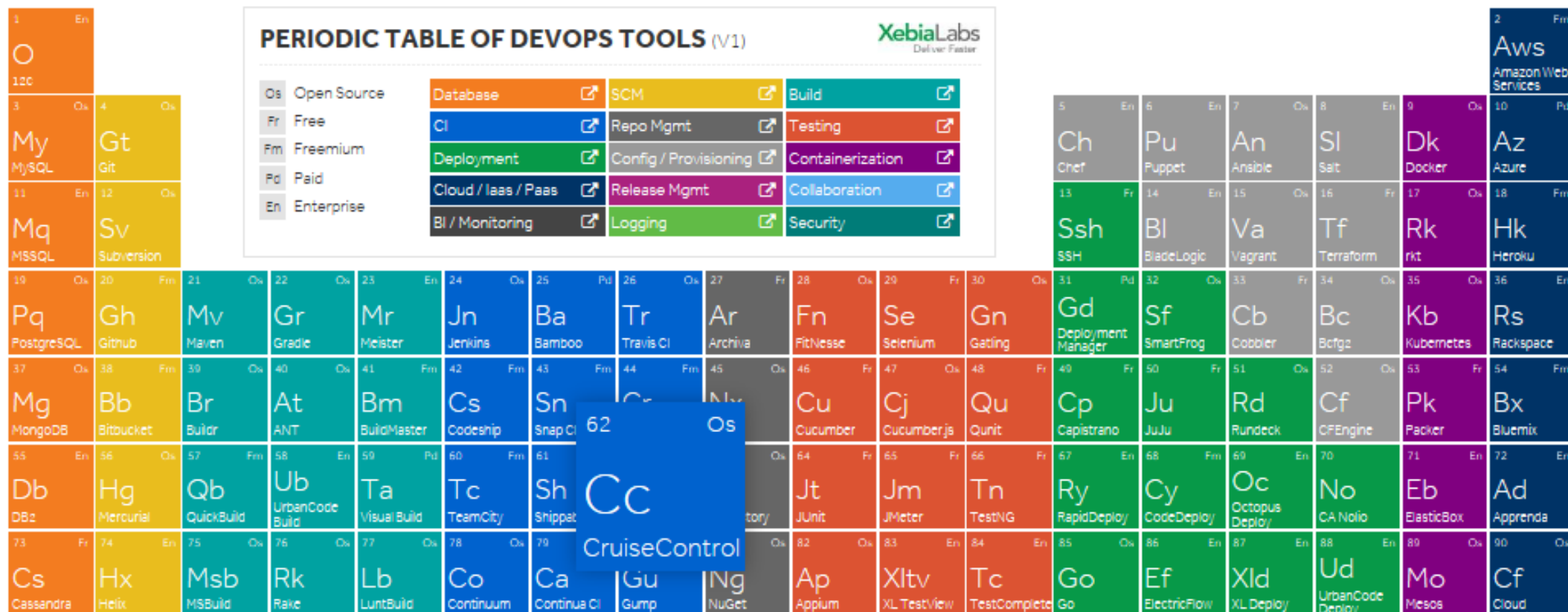
# DEVOPS PRACTICES: INFRASTRUCTURE AS CODE

- ❑ Infrastructure as code is a practice in which infrastructure is provisioned and managed using code and software development techniques, such as version control and continuous integration.
- ❑ The cloud's API-driven model enables developers and system administrators to interact with infrastructure programmatically, and at scale, instead of needing to manually set up and configure resources.
- ❑ Engineers can interface with infrastructure using code-based tools and treat infrastructure in a manner similar to how they treat application code. Because they are defined by code, infrastructure and servers can quickly be deployed using standardized patterns, updated with the latest patches and versions, or duplicated in repeatable ways.



# DEVOPS PRACTICES: INFRASTRUCTURE AS CODE

- ❑ IaC defines system topologies and resources to allow your team to manage the resources during the coding phase reliably in a controlled manner.
- ❑ You can store and version those definitions in a version control system, so your team can review and revert quickly.
- ❑ It helps reduce human errors in complex production environments.



Share

[!\[\]\(8ba916ed2d59a6c1c3c1de308d39676b\_img.jpg\)](#) [!\[\]\(1c230ae3e1bfef740ede90dde75d6544\_img.jpg\)](#) [!\[\]\(8c28abcc812daa7358c8fce8c8d0920d\_img.jpg\)](#) [!\[\]\(9ce535ae3f041d7c0eb1883b286082dc\_img.jpg\)](#) [!\[\]\(c2a4ef8aba0fe183743185a538adffc6\_img.jpg\)](#)

Embed

[!\[\]\(692b05d24fc6c4ac356cdd4368041ef8\_img.jpg\)](#) [!\[\]\(6e2fcb89970d41a52ce6628b942539cc\_img.jpg\)](#) [!\[\]\(40d1ccff93542053394ee29c2468e1f4\_img.jpg\)](#) [!\[\]\(1a7f2397350dd6fc353f5fb2f9363a0a\_img.jpg\)](#)

Become Excellent!

[!\[\]\(9b4d7e8189187fd24fbcf14371004c2c\_img.jpg\) Subscribe here!](#)

91 En Xlr XL Release	92 En Ur UrbanCode Release	93 En Ls CA Service Virtualization	94 En Bm BMC Release Process	95 En Hp HP Codar	96 Pd Ex Excel	97 En Pl Plutora Release	98 En Sr Serena Release	99 Fm Tr Trello	100 Pd Jr Jira	101 Fm Rf HipChat	102 Fm Sl Slack	103 Fm Fd Flowdock	104 Pd Pv Pivotal Tracker	105 En Sn ServiceNow
106 OX Ki Kibana	107 Fm Nr New Relic	108 OX Ni Nagios	109 OX Gg Ganglia	110 OX Ct Cacti	111 OX Gr Graphite	112 Fm Le Logentries	113 En Sp Splunk	114 Fm Sl Sumo Logic	115 OX Ls Logstash	116 Fm Lg Loggly	117 OX Gr Graylog	118 OX Sn Snipe-IT	119 OX Tr Terraform	120 En Cy CyberArk

**Contact us!**

