SOFTWARE ENGINEERING

Automated Software Development Process

Asst. Prof. Dr. Rathachai Chawuthai

Department of Computer Engineering
Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang

Agenda

- DevOps
- CI/CD
- CI/CD Tools

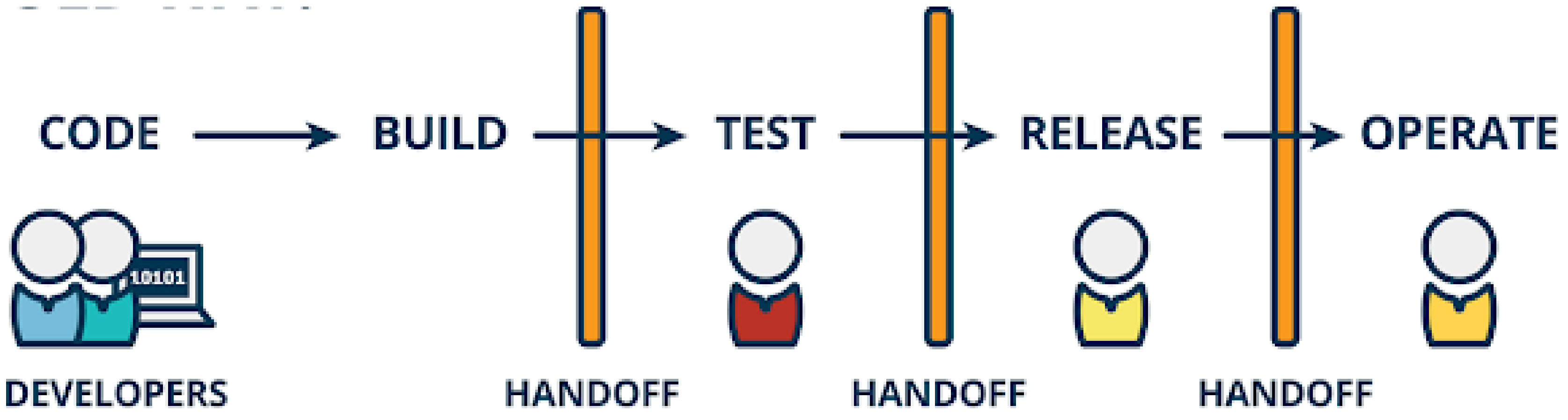
DevOps

Why DevOps Matters

- Software and the Internet have transformed the world and its industries, from shopping to entertainment to banking.
- Software no longer merely supports a business; rather it becomes an integral component of every part .
- Companies interact with their customers through software delivered as online services of a business. or applications and on all sorts of devices.
- They also use software to increase operational efficiencies by transforming every part of the value chain, such as logistics, communications, and operations.
- In a similar way that physical goods companies transformed how they design, build, and deliver products using industrial automation throughout the 20th century, companies in today's world must transform how they build and deliver software.

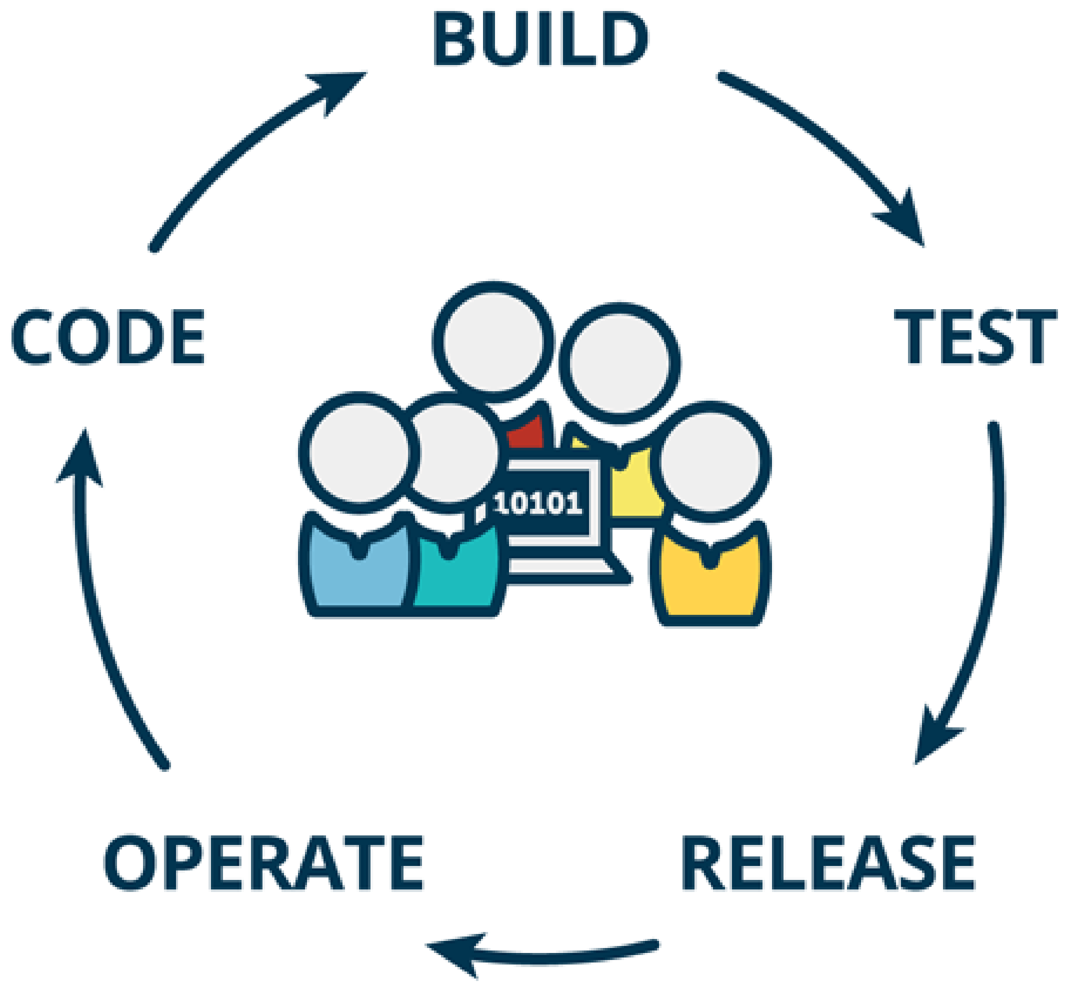
Ref: • https://aws.amazon.com/devops/what-is-devops/

Old Way



Ref: • http://www.mindtheproduct.com/2016/02/what-the-hell-are-ci-cd-and-devops-a-cheatsheet-for-the-rest-of-us/

New Way



Ref: • http://www.mindtheproduct.com/2016/02/what-the-hell-are-ci-cd-and-devops-a-cheatsheet-for-the-rest-of-us/

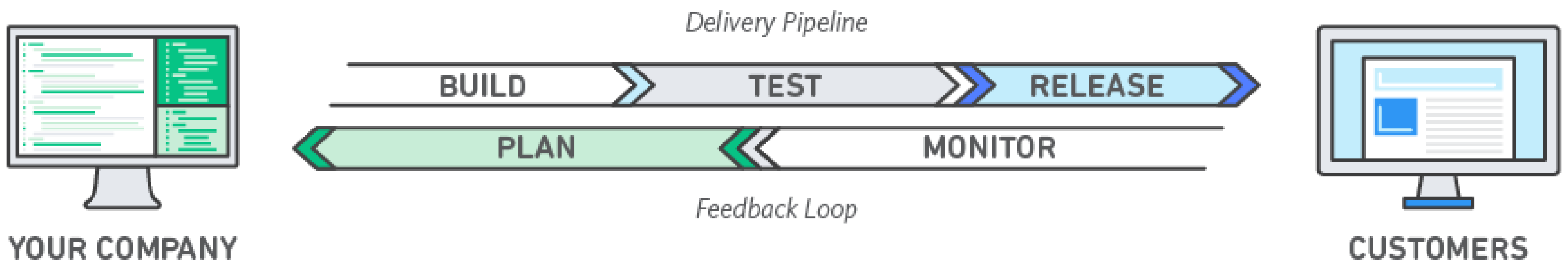
DevOps Model Defined

- DevOps is the combination of
  - cultural philosophies,
  - practices, and
  - tools

  that increases an organization's **ability to deliver** applications and services at **high velocity**: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes.
- This speed enables organizations to better serve their customers and compete more effectively in the market.

Ref: • https://aws.amazon.com/devops/what-is-devops/

Flow



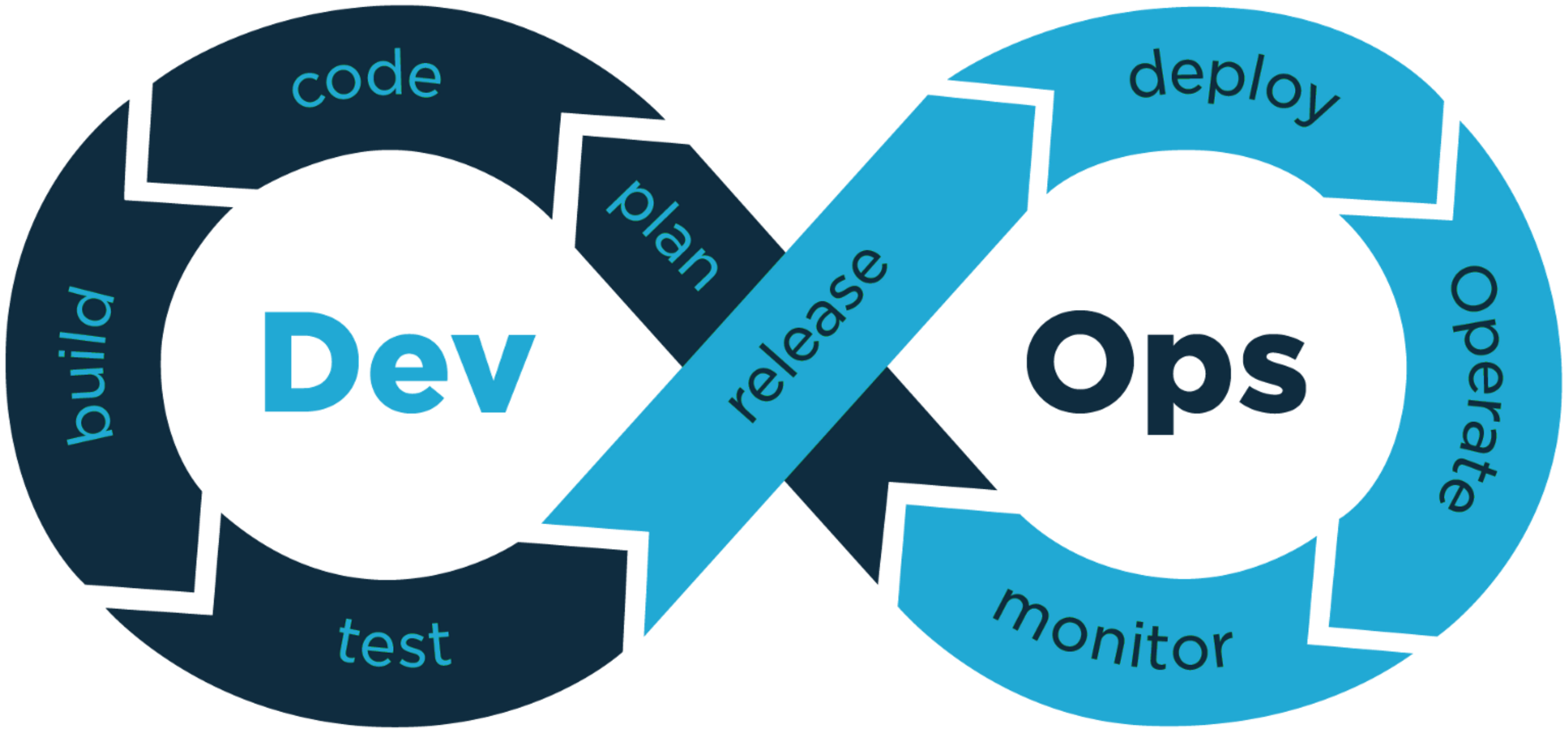
Ref: • https://aws.amazon.com/devops/what-is-devops/

How DevOps Works

- Under a DevOps model, development and operations teams are no longer "siloed." Sometimes, these two teams (dev & ops) are merged into a single team where the engineers work across the entire application lifecycle, from development and test to deployment to operations, and develop a range of skills not limited to a single function.
- In some DevOps models, quality assurance and security teams may also become more tightly integrated with development and operations and throughout the application lifecycle. When security is the focus of everyone on a DevOps team, this is sometimes referred to as DevSecOps.
- These teams use practices to automate processes that historically have been manual and slow. They use a technology stack and tooling which help them operate and evolve applications quickly and reliably. These tools also help engineers independently accomplish tasks (for example, deploying code or provisioning infrastructure) that normally would have required help from other teams, and this further increases a team's velocity.

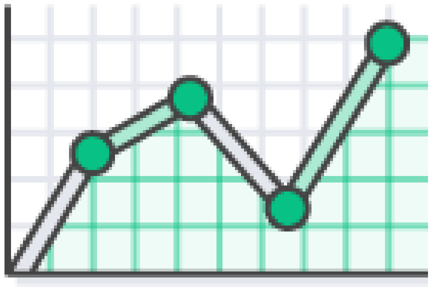
Ref: • https://aws.amazon.com/devops/what-is-devops/

DevOps



Ref: • https://tech.treebo.com/how-to-create-a-devops-roadmap-treebos-1-year-devops-journey-27c072001a28

Benefit of DevOps

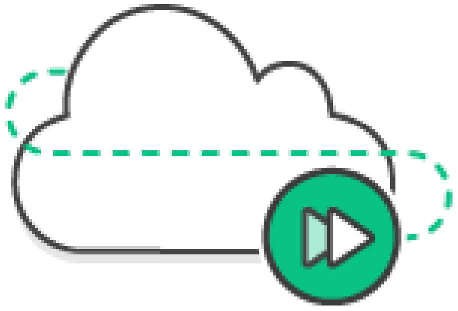


Speed

- Move at high velocity so you can innovate for customers faster, adapt to changing markets better, and grow more efficient at driving business results.
- The DevOps model enables your developers and operations teams to achieve these results. For example, microservices and continuous delivery let teams take ownership of services and then release updates to them quicker.

Ref: • https://aws.amazon.com/devops/what-is-devops/

Benefit of DevOps

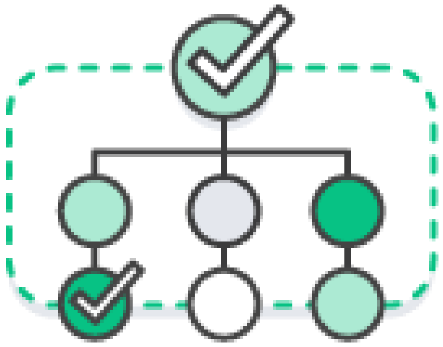


Rapid Delivery

- Increase the frequency and pace of releases so you can innovate and improve your product faster.
- The quicker you can release new features and fix bugs, the faster you can respond to your customers' needs and build competitive advantage.
- Continuous integration and continuous delivery are practices that automate the software release process, from build to deploy.

Ref: • https://aws.amazon.com/devops/what-is-devops/

Benefit of DevOps

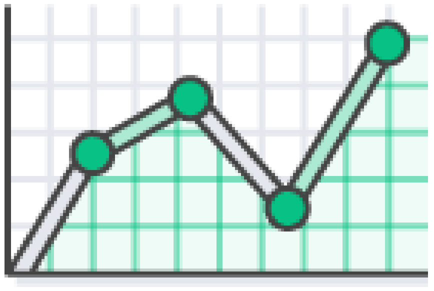


Reliability

- Ensure the quality of application updates and infrastructure changes so you can reliably deliver at a more rapid pace while maintaining a positive experience for end users.
- Use practices like continuous integration and continuous delivery to test that each change is functional and safe.
- Monitoring and logging practices help you stay informed of performance in real-time.

Ref: • https://aws.amazon.com/devops/what-is-devops/

Benefit of DevOps



Scale

- Operate and manage your infrastructure and development processes at scale.
- Automation and consistency help you manage complex or changing systems efficiently and with reduced risk.
- For example, infrastructure as code helps you manage your development, testing, and production environments in a repeatable and more efficient manner.

Ref: • https://aws.amazon.com/devops/what-is-devops/

Benefit of DevOps

Improved Collaboration

- Build more effective teams under a DevOps cultural model, which emphasizes values such as ownership and accountability.
- Developers and operations teams collaborate closely, share many responsibilities, and combine their workflows.
- This reduces inefficiencies and saves time (e.g. reduced handover periods between developers and operations, writing code that takes into account the environment in which it is run).

Ref: • https://aws.amazon.com/devops/what-is-devops/

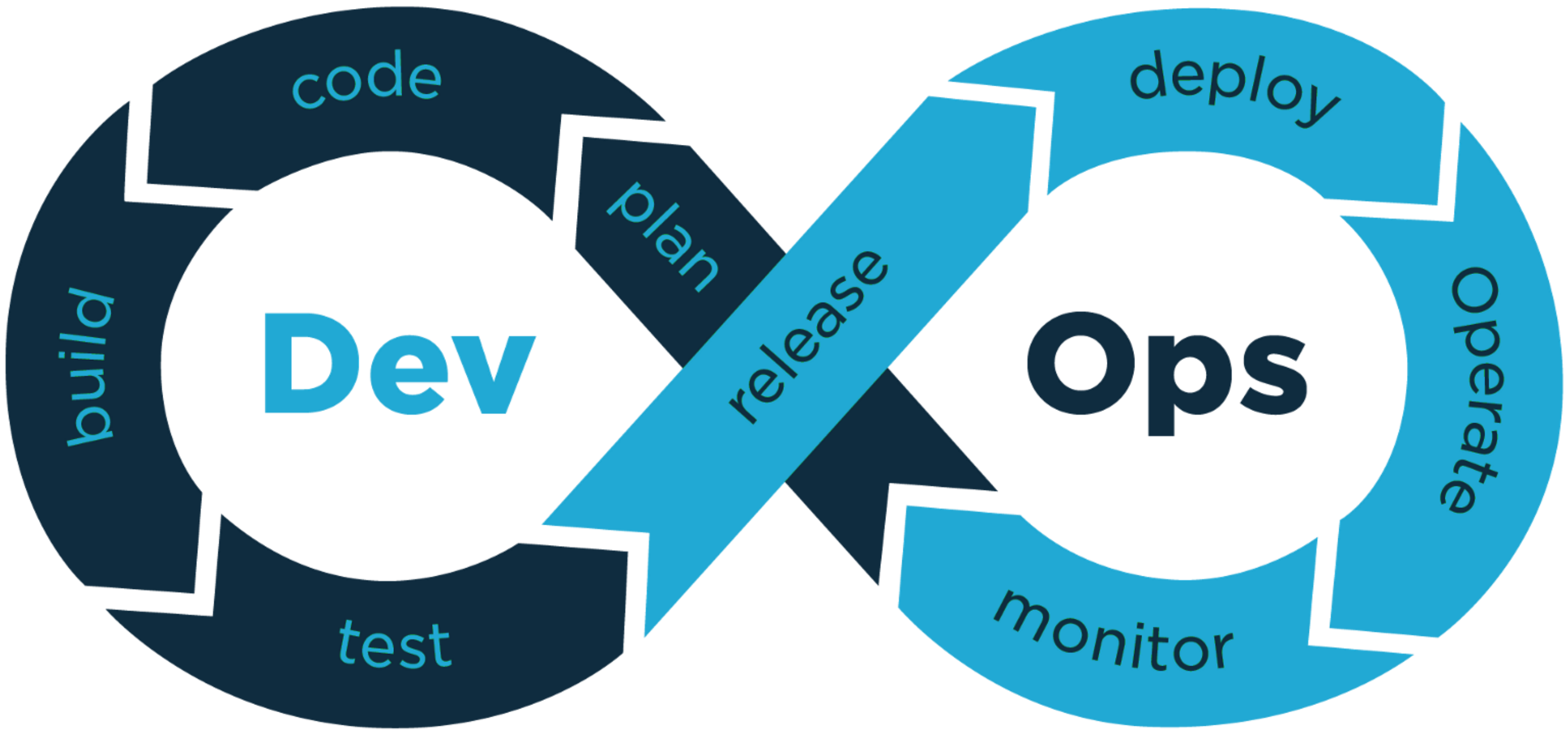
DevOps Cultural Philosophy

- Transitioning to DevOps requires a change in culture and mindset. At its simplest, DevOps is about removing the barriers between two traditionally siloed teams, development and operations. In some organizations, there may not even be separate development and operations teams; engineers may do both.
- With DevOps, the two teams work together to optimize both the productivity of developers and the reliability of operations. They strive to communicate frequently, increase efficiencies, and improve the quality of services they provide to customers.
- They take full ownership for their services, often beyond where their stated roles or titles have traditionally been scoped by thinking about the end customer's needs and how they can contribute to solving those needs.
- Quality assurance and security teams may also become tightly integrated with these teams. Organizations using a DevOps model, regardless of their organizational structure, have teams that view the entire development and infrastructure lifecycle as part of their responsibilities.

Ref: • https://aws.amazon.com/devops/what-is-devops/

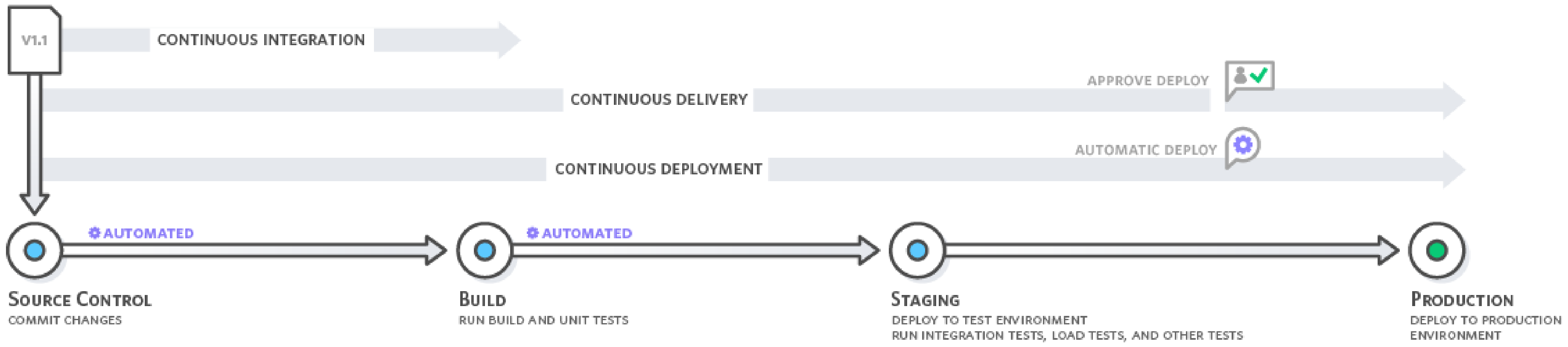
CI/CD

DevOps



Ref: • https://tech.treebo.com/how-to-create-a-devops-roadmap-treebos-1-year-devops-journey-27c072001a28

CI/CD



Ref:
- https://aws.amazon.com/devops/continuous-integration/

CI : Continuous integration

- Continuous integration is a DevOps software development practice where developers regularly merge their code changes into a central repository, after which automated builds and tests are run.
- Continuous integration most often refers to the build or integration stage of the software release process and entails both an automation component (e.g. a CI or build service) and a cultural component (e.g. learning to integrate frequently).
- 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.

Ref: • https://aws.amazon.com/devops/continuous-integration/

CI: How

- With continuous integration, developers frequently commit to a shared repository using a version control system such as Git. Prior to each commit, developers may choose to run local unit tests on their code as an extra verification layer before integrating.
- A continuous integration service automatically builds and runs unit tests on the new code changes to immediately surface any errors.
- Continuous integration refers to the build and unit testing stages of the software release process. Every revision that is committed triggers an automated build and test.
- With continuous delivery, code changes are automatically built, tested, and prepared for a release to production. Continuous delivery expands upon continuous integration by deploying all code changes to a testing environment and/or a production environment after the build stage.

Ref: • https://aws.amazon.com/devops/continuous-integration/

CD: Continuous Delivery

- Continuous delivery is a software development practice where code changes are automatically prepared for a release to production. A pillar of modern application development, continuous delivery expands upon continuous integration by deploying all code changes to a testing environment and/or a production environment after the build stage. When properly implemented, developers will always have a deployment-ready build artifact that has passed through a standardized test process.
- Continuous delivery lets developers automate testing beyond just unit tests so they can verify application updates across multiple dimensions before deploying to customers. These tests may include UI testing, load testing, integration testing, API reliability testing, etc. This helps developers more thoroughly validate updates and pre-emptively discover issues. With the cloud, it is easy and cost-effective to automate the creation and replication of multiple environments for testing, which was previously difficult to do on-premises.

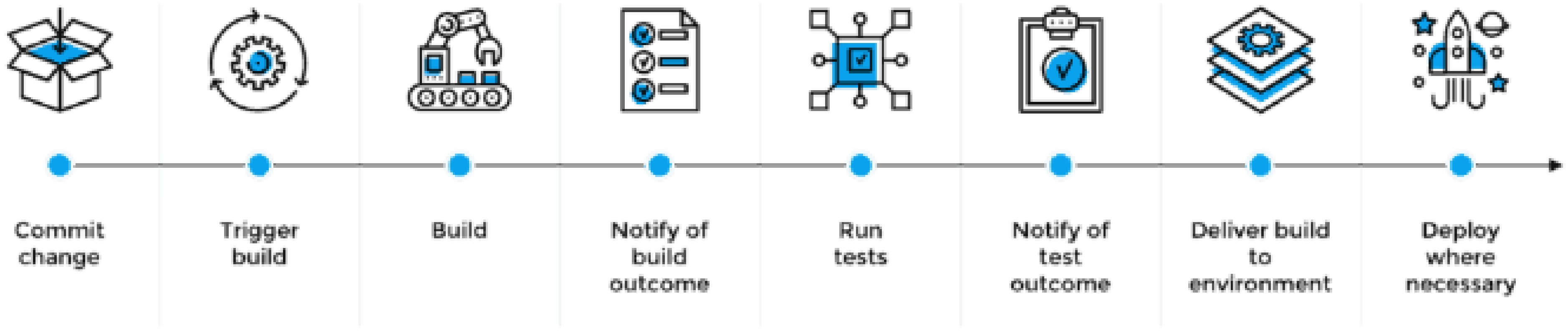
Ref: • https://aws.amazon.com/devops/continuous-delivery/

CD: Continuous Delivery

- Continuous delivery automates the entire software release process. Every revision that is committed triggers an automated flow that builds, tests, and then stages the update. The final decision to deploy to a live production environment is triggered by the developer.

Ref: • https://aws.amazon.com/devops/continuous-delivery/

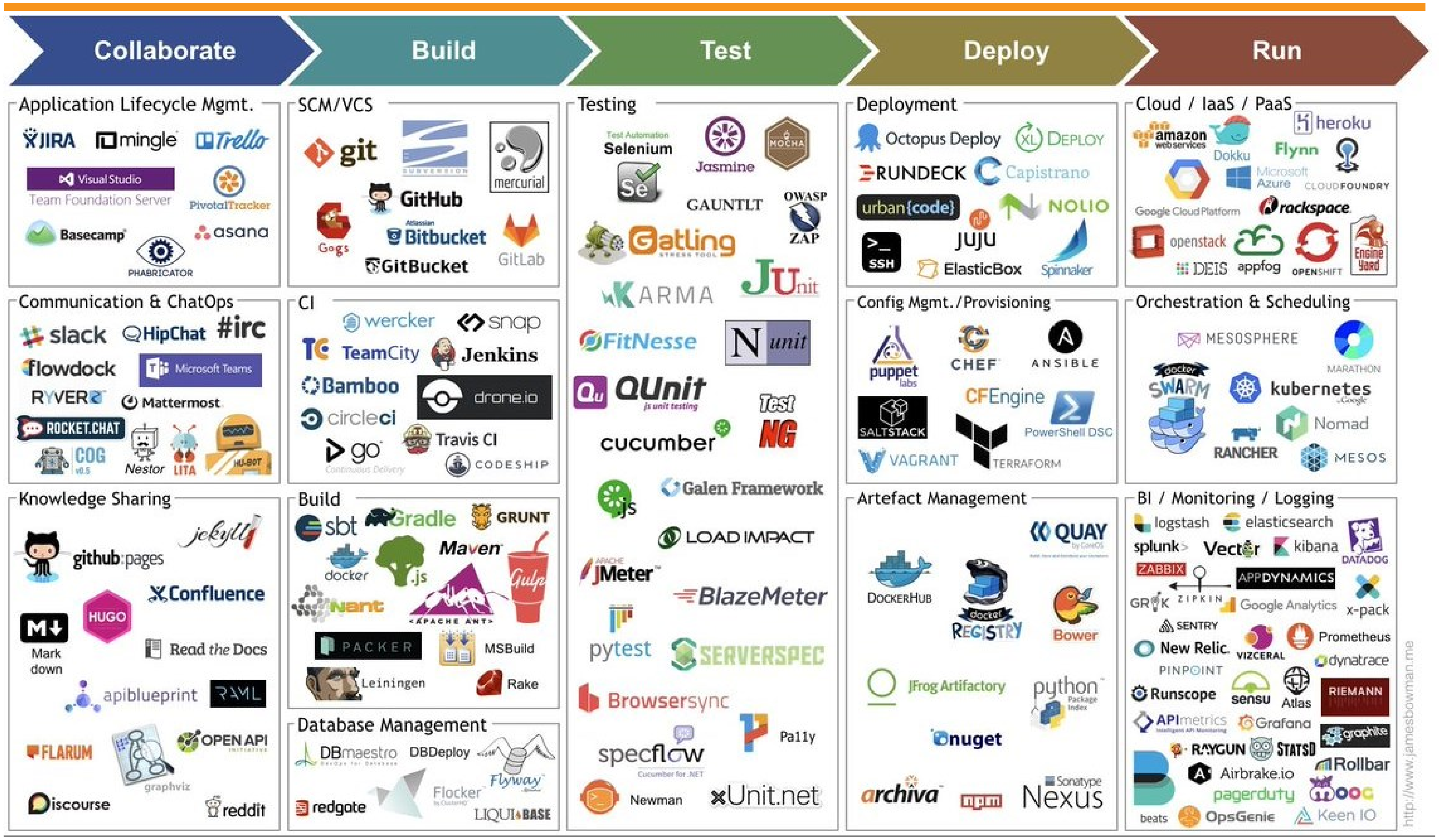
CI/CD Pipeline



Ref: • https://hackernoon.com/understanding-the-basic-concepts-of-cicd-fw4k32s1

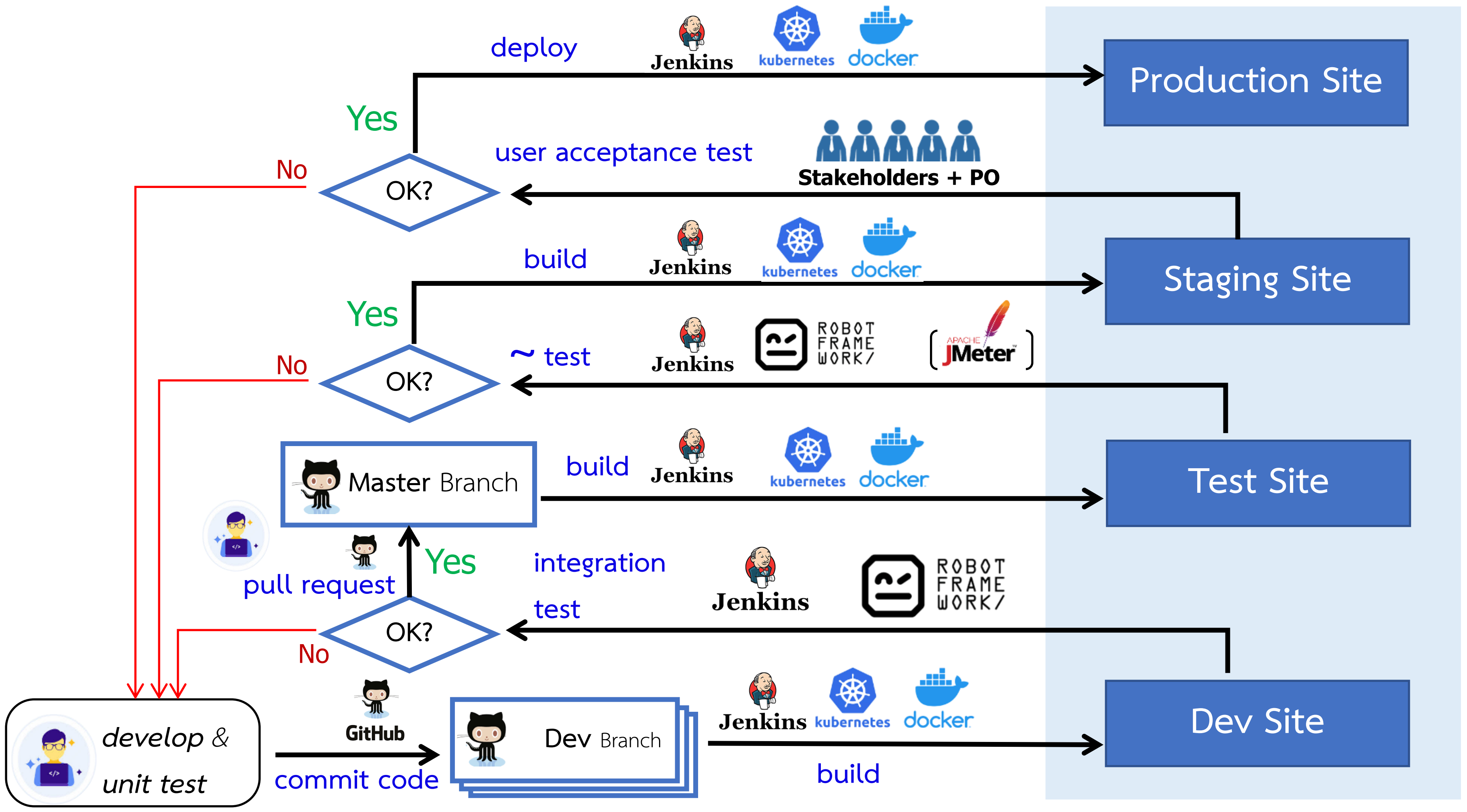
CI/CD Tools

CI/CD Tools



Ref: • https://aws.amazon.com/devops/continuous-integration/

Example CI/CD Flow & Tools



Example CI/CD Skills

Linux

GitHub

Jenkins

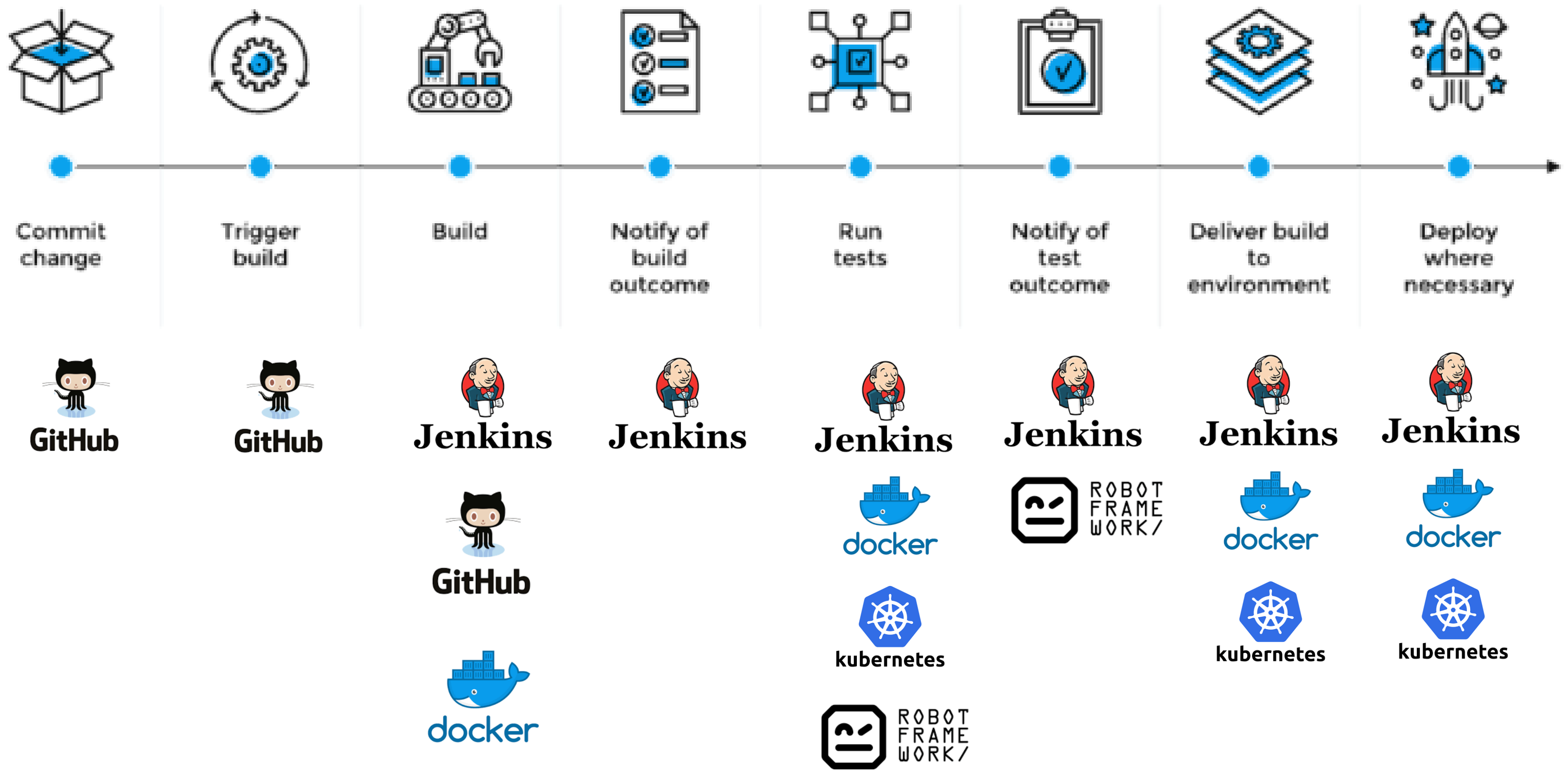
docker

kubernetes

ROBOT FRAMEWORK

Summary

Key Points



Ref: • (image) https://ton.packetlove.com/blog/life-style/devops-ci-cd.html

“

To successfully implement continuous delivery, you need to change the culture of how an entire organization views software development efforts.

”

Tommy Tynjä

つづく