



Cloud, Infrastructure and DevOps

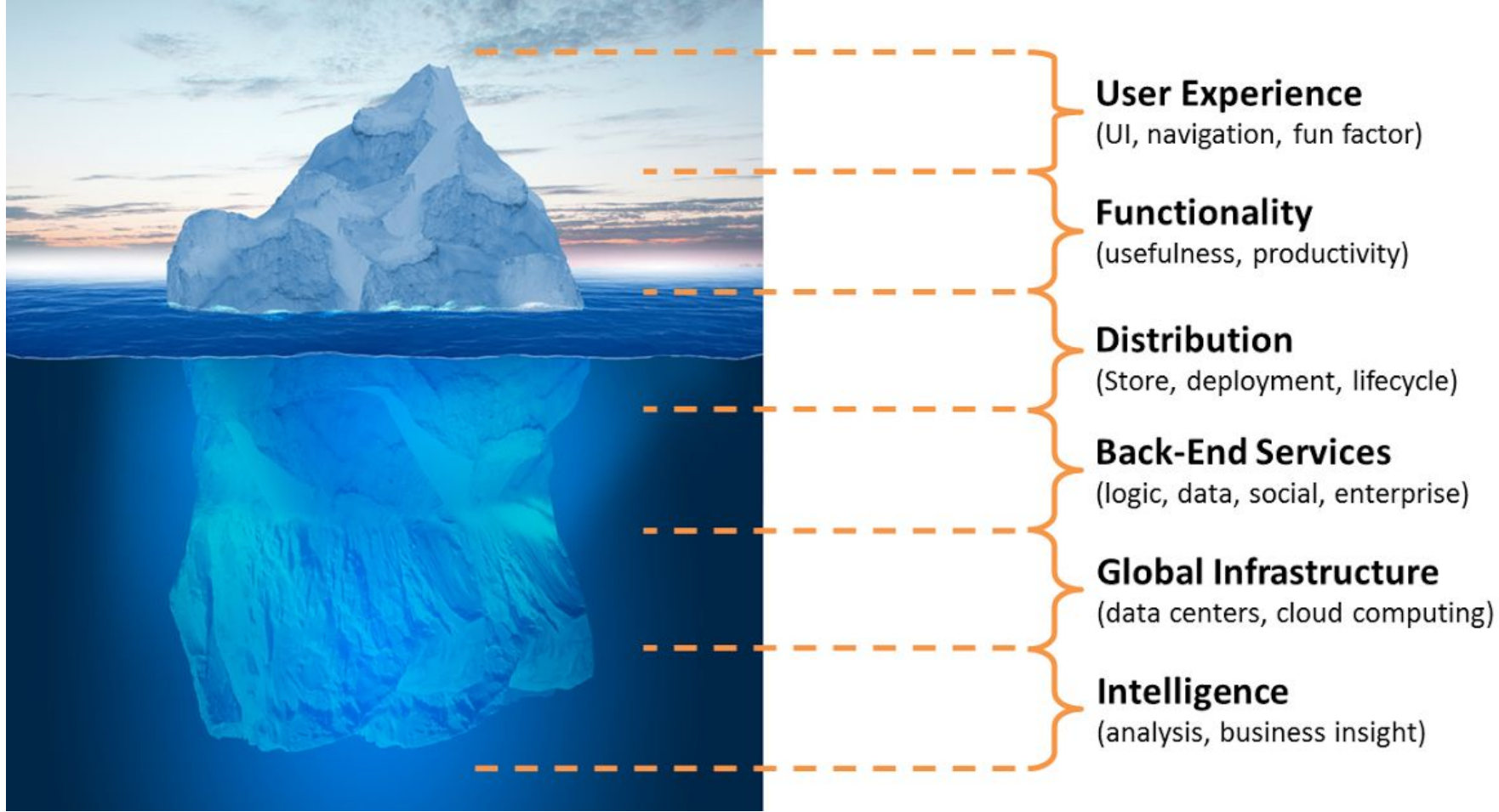
Yulia Tenincheva

Senior Cloud Engineer, MentorMate

Agenda

- IT Infrastructure, *on premise* vs *Cloud*
- The Five pillars of Cloud Computing
 - (Well-Architected Framework)
- Cloud Native Applications, Serverless
- DevOps Culture, DevSecOps
- DevOps Tools, Pipelines, CI / CD, Demo





Infrastructure

On-Premises



Ongoing Costs

- Apply patches, upgrades
- Downtime
- Performance tuning
- Rewrite customizations
- Rewrite integrations
- Upgrade dependent applications
- Ongoing burden on IT (hardware)
- Maintain/upgrade network
- Maintain/upgrade security
- Maintain/upgrade database

Cloud Computing



Ongoing Costs

- Subscription fees
- Training
- Configuration
- System Administration

"Cloud is about how you do computing, not where you do computing."

~ Paul Maritz, CEO of VMware

"Cloud computing is empowering, as anyone in any part of world with internet connection and a credit card can run and manage applications in the state of the art global data-centers; companies leveraging cloud will be able to innovate cheaper and faster."

~ Jamal Mazhar, Founder and CEO of Kaavo

**There is
no cloud**



**it's just
someone else's
computer**

On-Premises

Infrastructure (as a Service)

Platform (as a Service)

Software (as a Service)

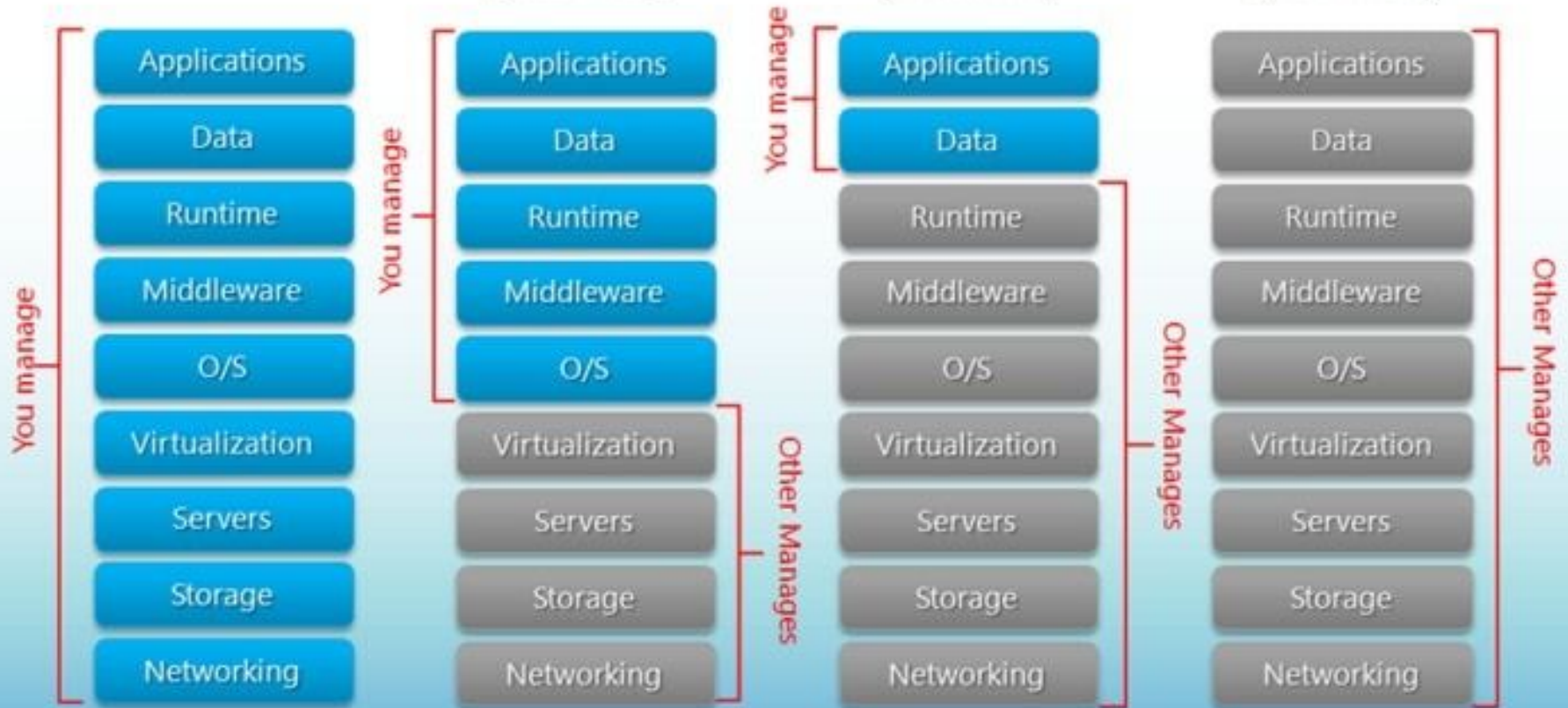


Figure 1. Magic Quadrant for Cloud Infrastructure and Platform Services



← [Gartner Report](#)



Analytics



Application Integration



AR & VR



AWS Cost Management



Blockchain



Business Applications



Compute



Containers



Customer Engagement



Database



Developer Tools



End User Computing



Front-End Web & Mobile



Game Tech



Internet of Things



Machine Learning



Management & Governance



Media Services



Migration & Transfer



Networking & Content
Delivery



Quantum Technologies



Robotics



Satellite



Security, Identity &
Compliance



Storage

A world map showing the global presence of Mentormate. Orange circles with numbers indicate the number of offices in each region. The distribution is as follows:

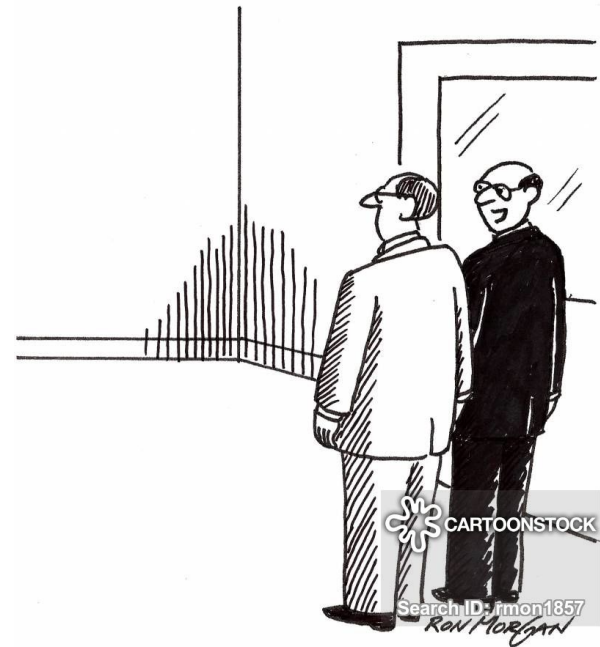
- North America:** 4 offices in the US West, 3 in the US East, 3 in Mexico, and 3 in Brazil.
- Europe:** 3 offices in the UK, 3 in France, 3 in Germany, 3 in Italy, 3 in Spain, and 3 in Sweden.
- Asia:** 3 offices in India, 3 in China, 2 in Japan, 3 in South Korea, 4 in Taiwan, 1 in Hong Kong, 3 in Singapore, 3 in Malaysia, 3 in Indonesia, and 3 in Australia.
- Other Regions:** 3 offices in Africa, 3 in the Middle East, and 3 in Southeast Asia.

Cloud Native Applications

Cloud native is an **approach** to building and running applications that fully exploit the **advantages** of the **cloud computing** model.

Cloud-Native Apps Are Different

Services	<ul style="list-style-type: none">• All functionality is published and consumed via web services
Handling Failures	<ul style="list-style-type: none">• Every Integration point will eventually fail one time or another• Be prepared to handle all kind of failures
Horizontal Scalability	<ul style="list-style-type: none">• Design for Scale Out
Asynchronous Processing	<ul style="list-style-type: none">• Break down the task, process requests asynchronously• Use queues to decouple functionality• Eventual consistency model
Stateless Model	<ul style="list-style-type: none">• Build stateless services that can be scaled out and load balanced
Minimize Human Intervention	<ul style="list-style-type: none">• Go DevOps/NoOps



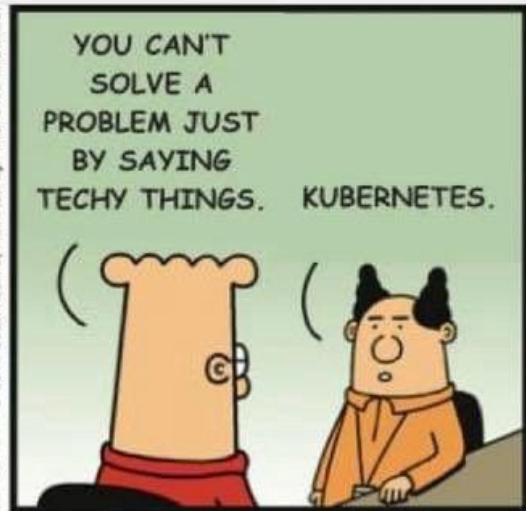
"Everything is in the Cloud."



@ScottAdamsSays
Dilbert.com



11-08-17 © 2017 Scott Adams, Inc. All rights reserved. McMeel



Serverless

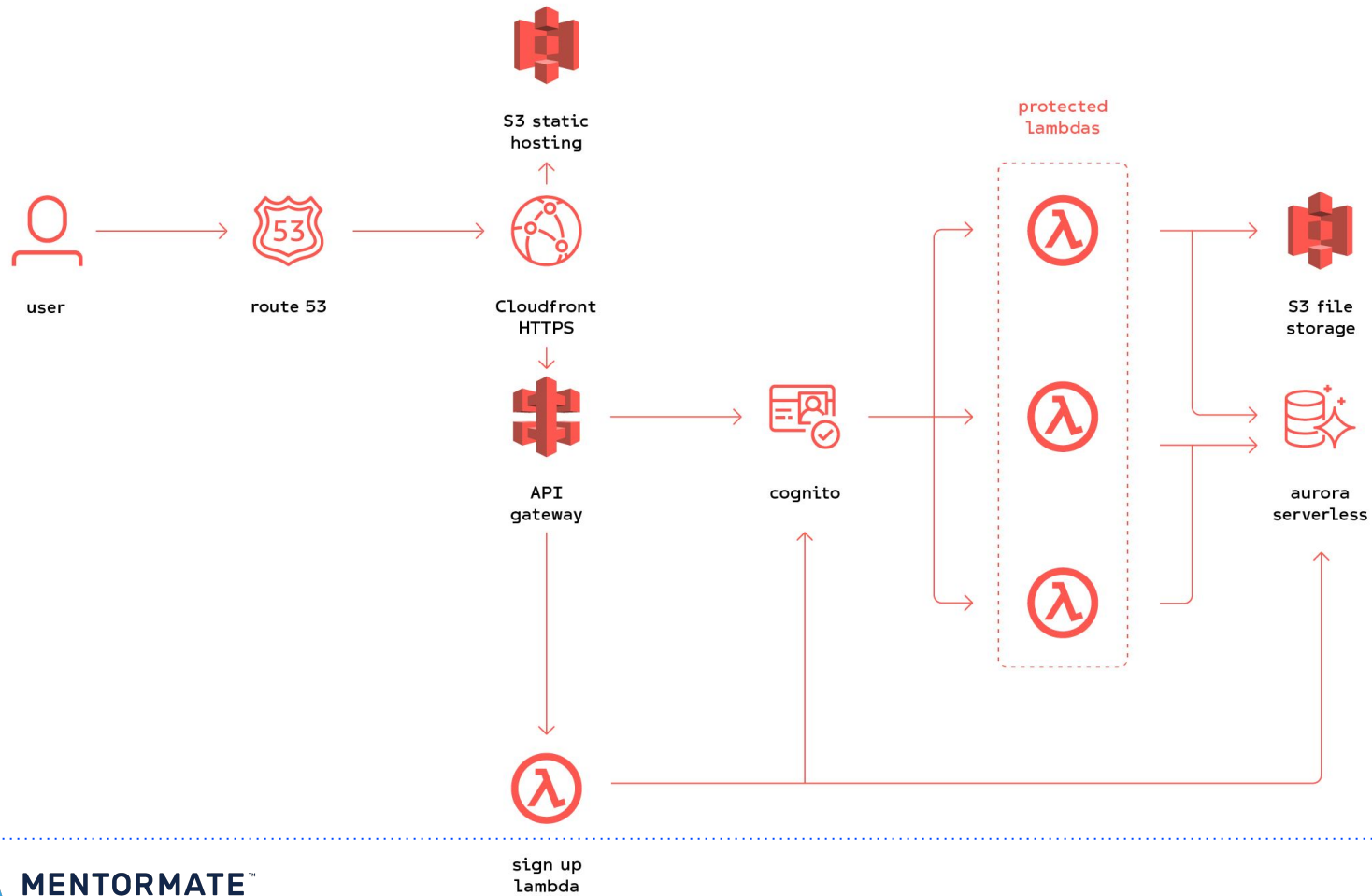
Serverless is a way to focus on
business value

Remember:

Days of programming can
save hours of configuration

Remember:

Hours of troubleshooting can save
you 5 minutes of reading the
documentation



Well-Architected Framework

The 5 Pillars of the AWS Well-Architected - [Link](#)



Security

Project information, systems, and assets while delivering business value through risk assessments and mitigation strategies



Reliability

Recover from infrastructure or service failures, dynamically acquire computer resources to meet demand, mitigate disruptions



Performance Efficiency

Use computing resources efficiently to meet system requirements, and to maintain efficiency as demand changes and technologies evolve



Cost Optimization

Avoid and eliminate unneeded cost or suboptimal resources with cost effective resources, matched supply and demand, and expenditure awareness



Operational Excellence

Run and monitor systems to deliver business value and continually improve supporting processes and procedures

When architecting technology solutions on AWS, if you neglect the **five pillars**, it can become challenging to build a **system that delivers** on your expectations and requirements.

Prepare your application for deployment

- Stateless!
- Version control, git branches and tags
- External configuration, secrets, multi-env config
- Comprehensive logging
- Health-check endpoint (/healthz)
- Security, SSL
- Documentation - how to run the application



DevOps

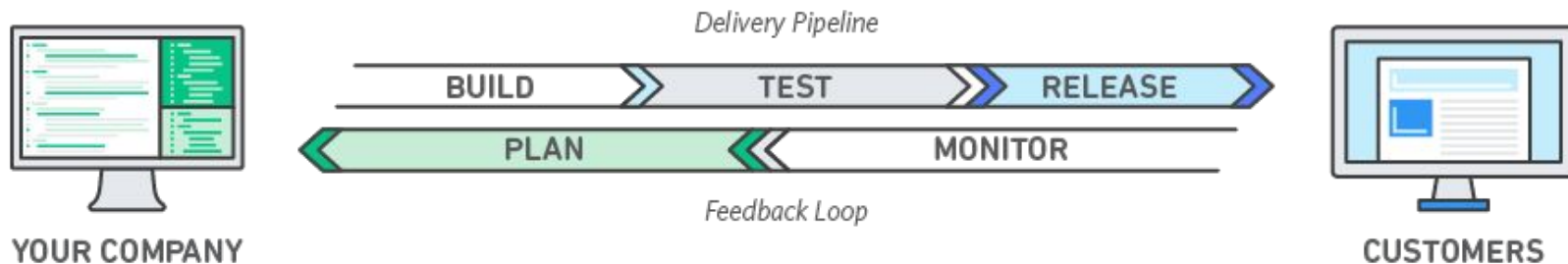
**“Philosophy of the efficient
development, deployment and
operation of the highest quality
software possible”**

—A friend from RackSpace

DevOps Model Defined

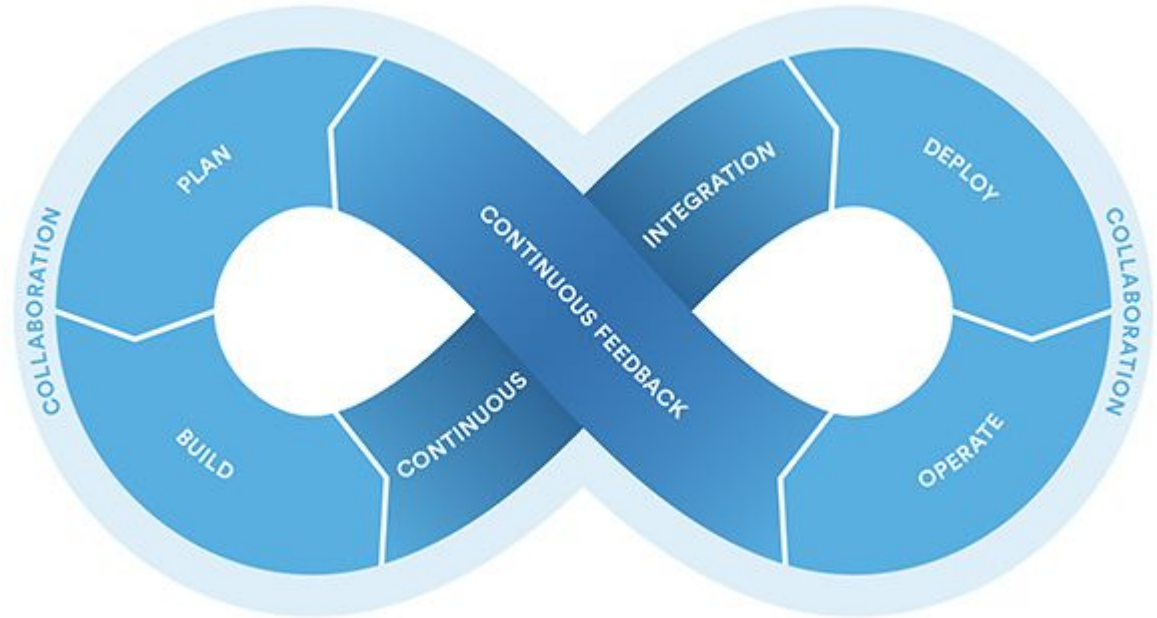
DevOps is the combination of cultural philosophies, practices, and tools that **increases an organization's ability to deliver** applications and services at high velocity.

This speed enables organizations to better serve their customers and compete more effectively in the market.



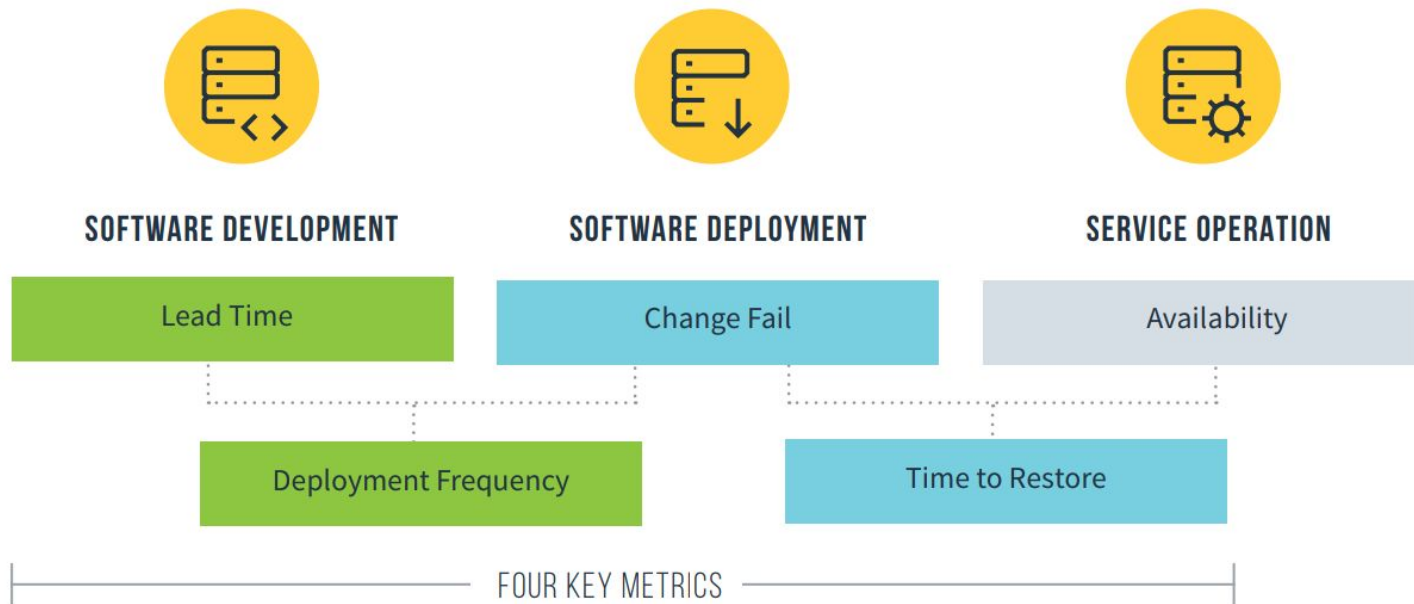
Benefits of DevOps

- Speed
- Rapid Delivery
- Reliability
- Scale
- Improved Collaboration
- Security (DevSecOps)



The State of DevOps

PERFORMANCE METRICS



The actual metric of success:

To what degree (from 1 to 7)

do we fear doing
deployments?

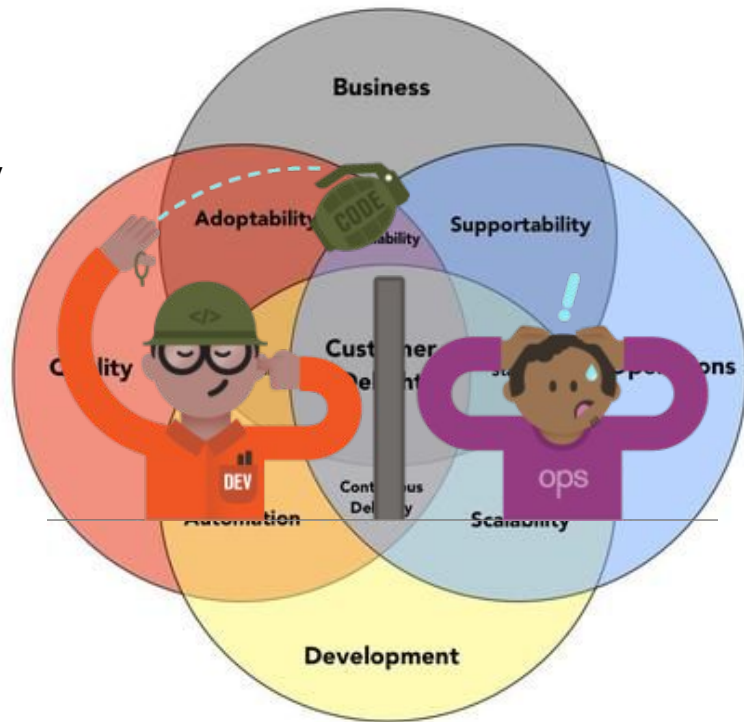


“Delivering software **quickly, reliably,**
and **safely** is at the heart of
technology transformation and
organizational performance.”

—State of DevOps 2019

DevOps Cultural Philosophy

- DevOps is about removing the barriers
- Teams work together to optimize both the productivity of developers and the reliability of operations
- Communicate frequently and improve the quality of services they provide to customers
- Take full ownership for their services
- Thinking about the end customer's needs and how YOU can contribute to solving those needs



“You build it, you run it”

—Werner Vogels, CTO, Amazon

WHAT DO WE WANT?



TO DELIVER BETTER SOFTWARE!



WHEN DO WE WANT IT?



CONTINUOUSLY!!!



DevOps Practices

- Continuous Integration (CI)

Very frequent but small updates is the key

- Continuous Delivery (CD)

Always have a deployment-ready build artifact

- Microservices

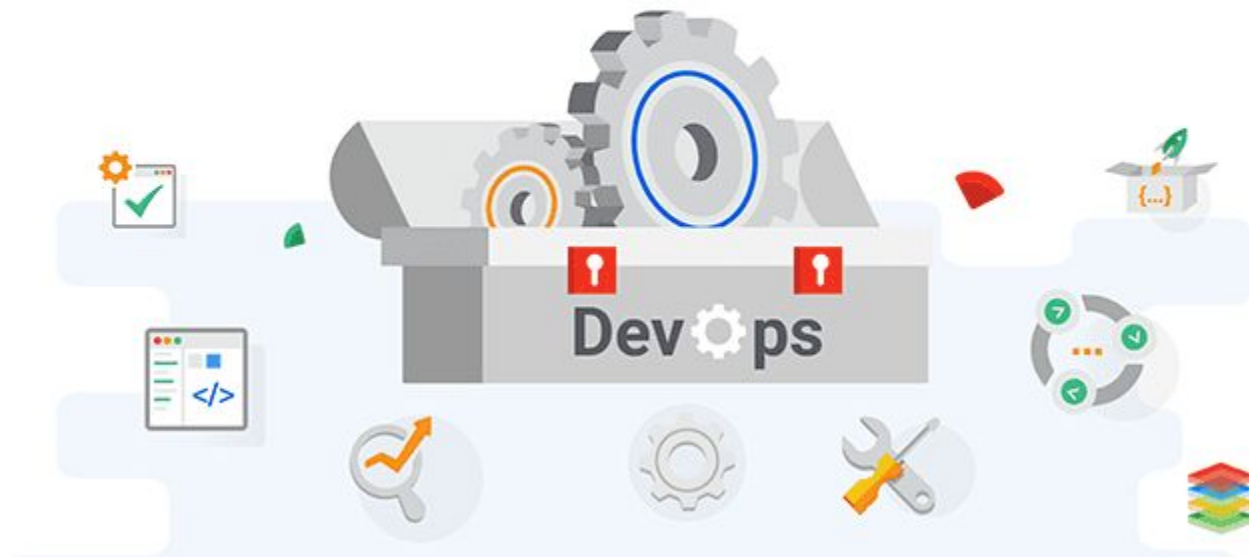
- Infrastructure as Code (IaC)

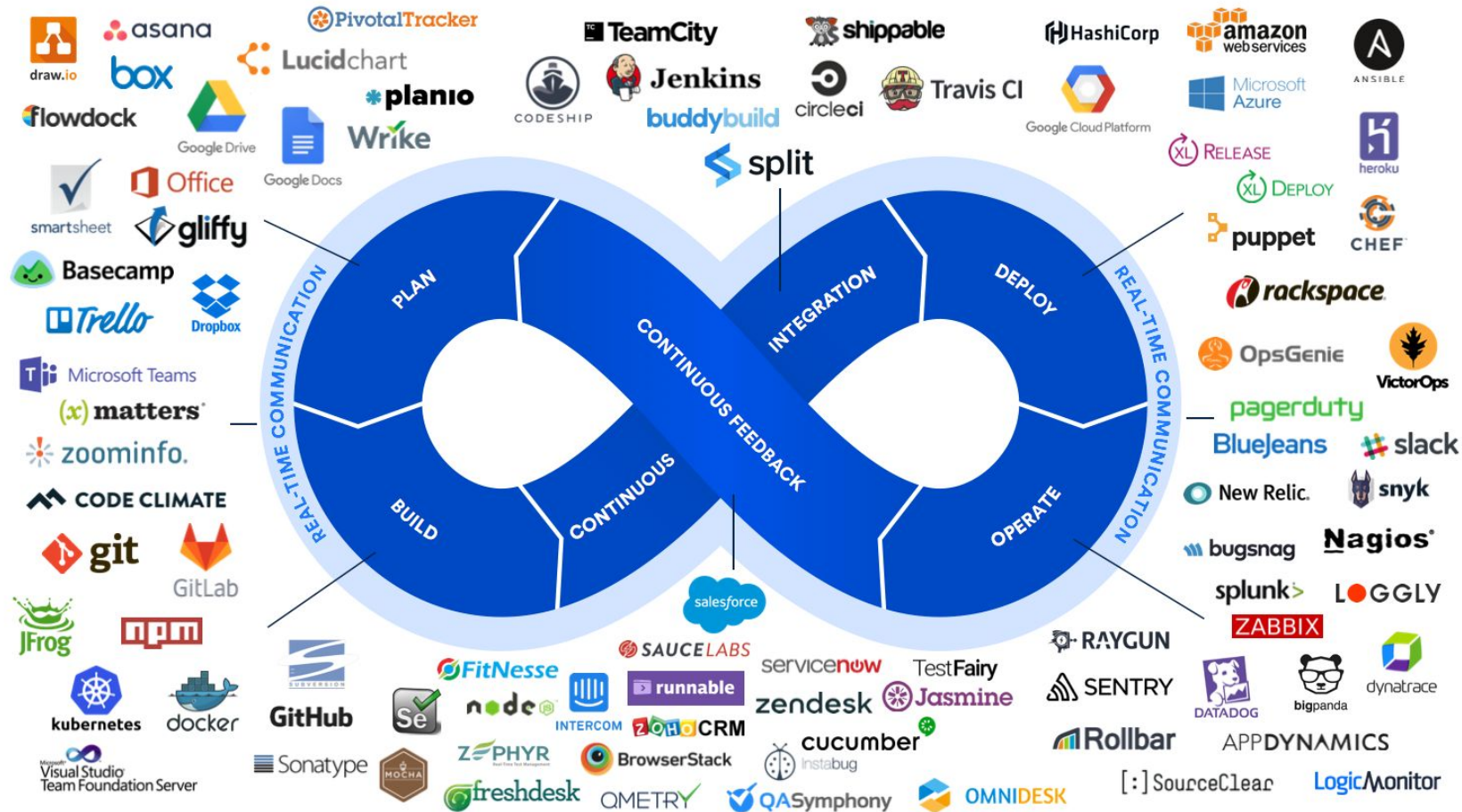
- Monitoring & Logging

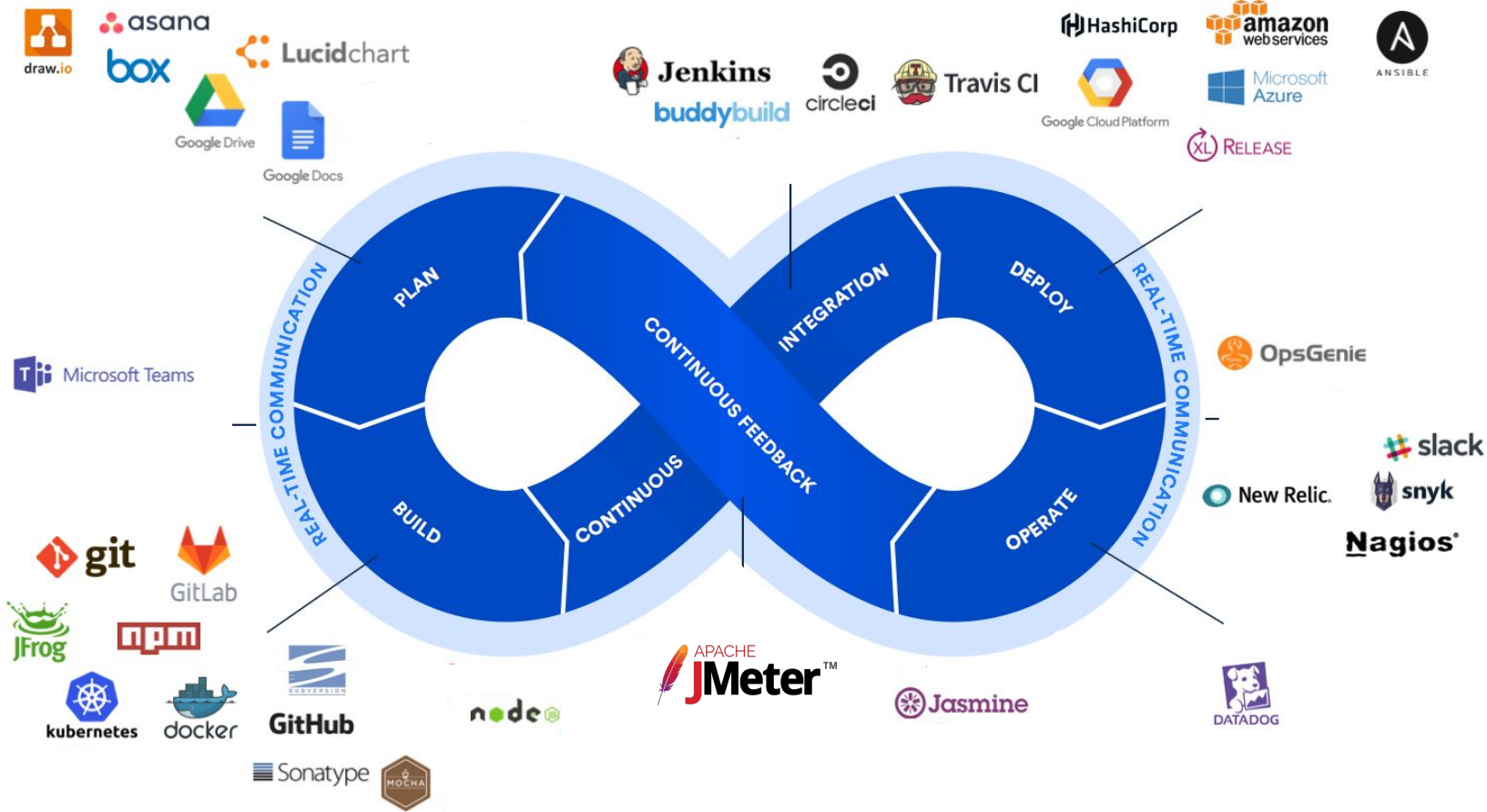
- Communication & Collaboration



List of Best **DevOps** Tools



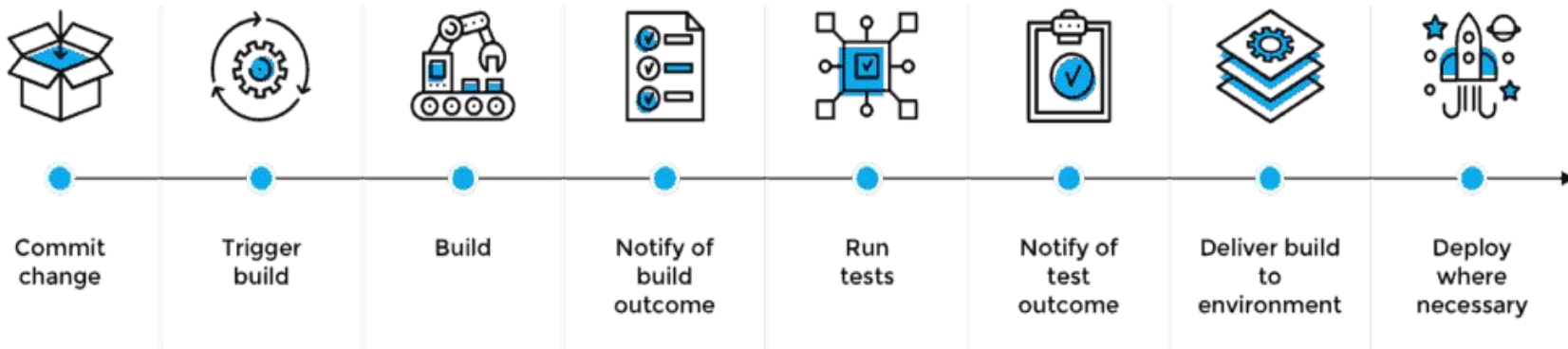




CI/CD Pipelines

Abstract

CI/CD Pipeline



The most satisfying thing in the career?

“It’s when I wrote the code, pushed the button to deploy it, watched the metrics to see if it actually worked in production, and fixed it if it broke”

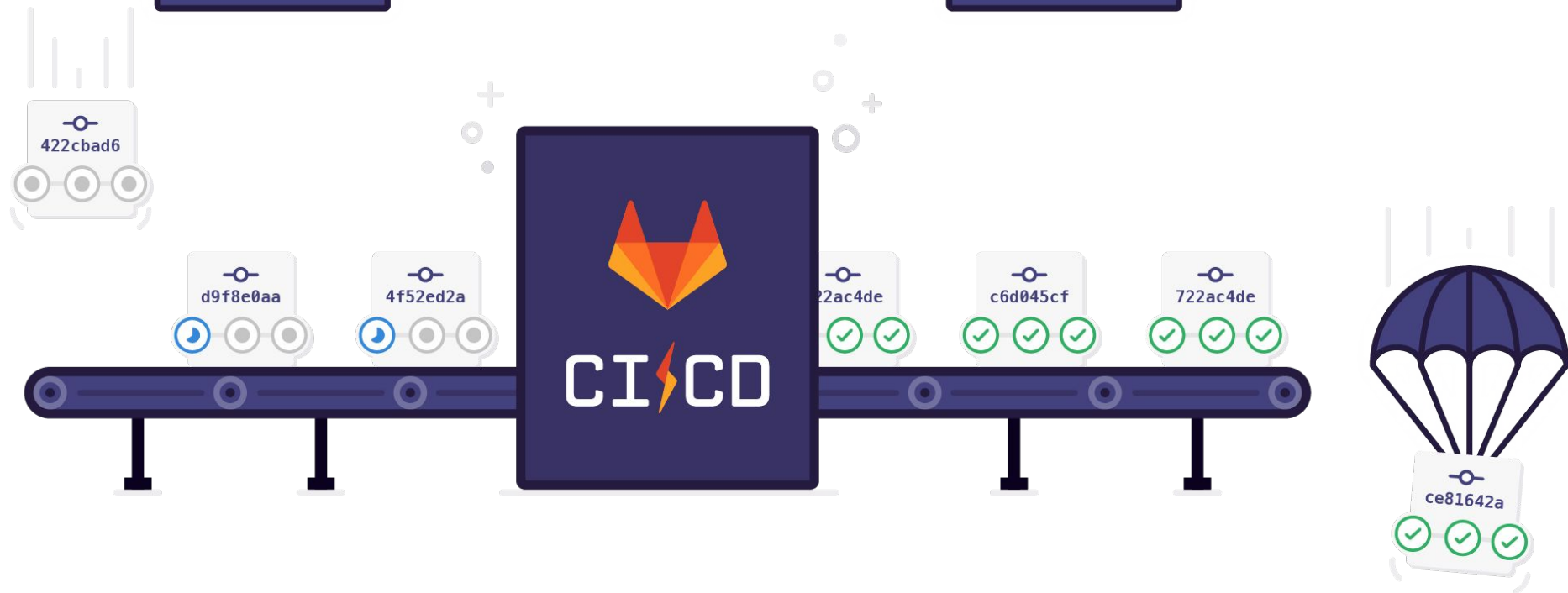
—Director of operations engineering Nike, Inc.



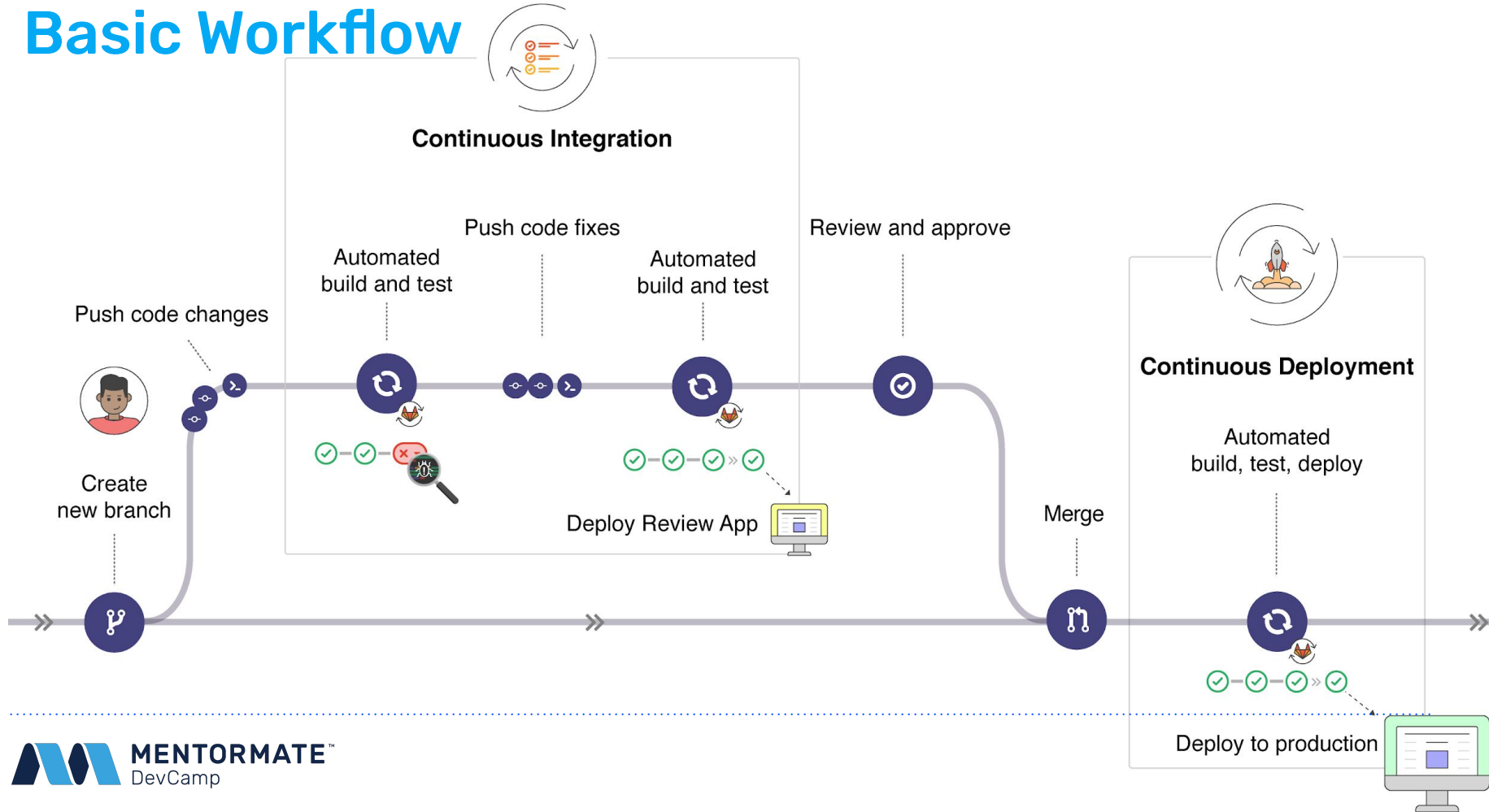
Gitlab CI

TEST ➡

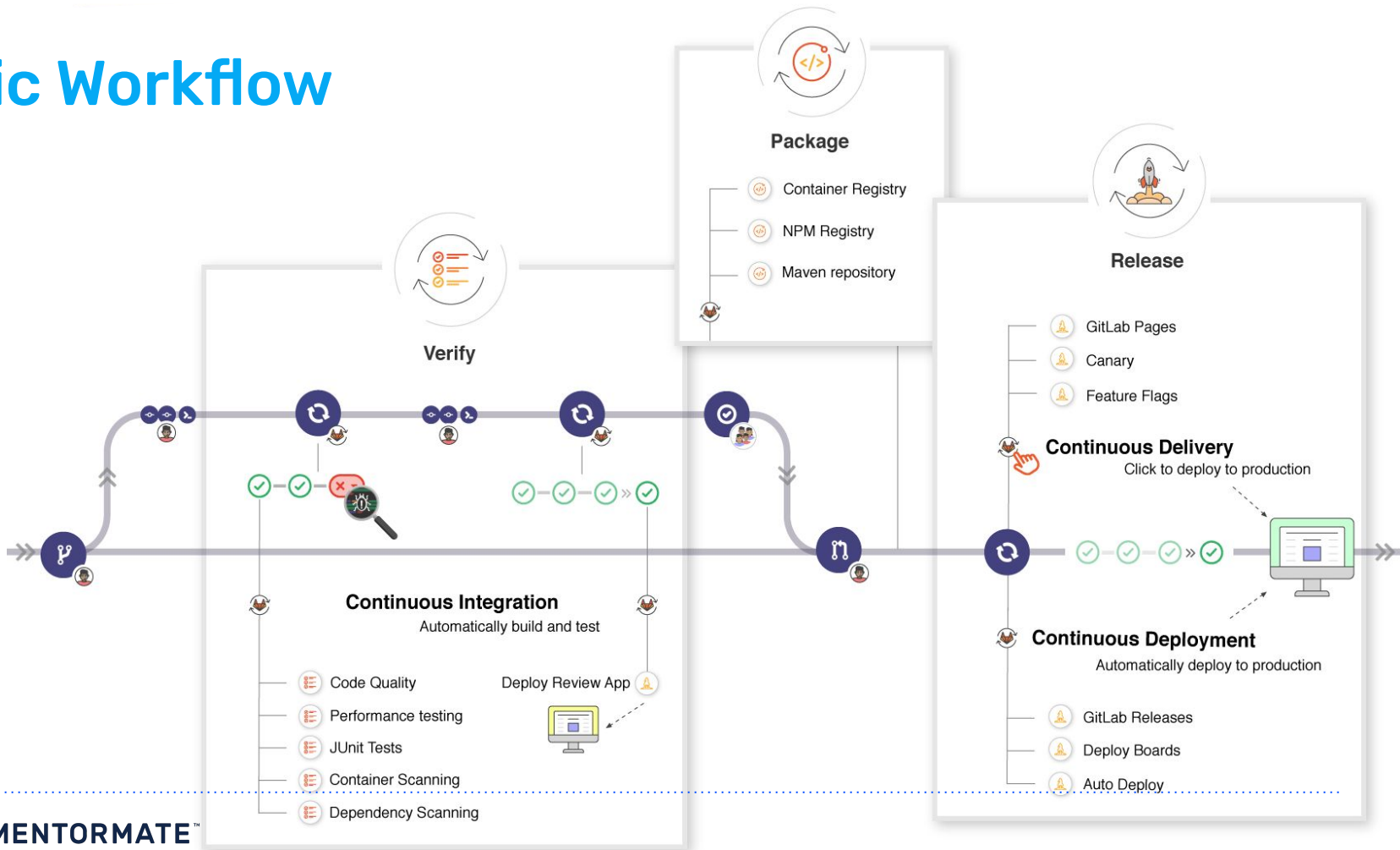
DEPLOY ➡



Basic Workflow



Basic Workflow



Example .gitlab.ci file

```
image: node:latest

stages:
  - lint

eslint:
  stage: lint
  script:
    # Install ESLint in this docker container
    - npm install -g eslint
    # Configure ESLint (will read your .eslintrc file)
    - eslint --init
    # Run ESLint
    - eslint <your_js_file>
```

Live Demo

Resources

- The Lean Startup
by Eric Ries
- The DevOps Handbook
by Gene Kim, Patrick Debois, John Willis and Jez Humble
- The Phoenix Project
- Accelerate
- [The State of DevOps](#) (2019)