
DevOps - Week 1 - Intro to DevOps

Muhammad Ali Kahoot
Dice Analytics

About Me - M. Ali Kahoot



/kahootali



/kahootali

- Senior SRE at PawaPay, a UK based Fintech (Remote)
- DevOps Consultant, Daleel (Remote)
- Former Senior DevOps Engineer, Tarabut Gateway (Remote)
- DevOps Trainer, Dice Analytics
- DevOps Consultant
- Ex-Team Lead, DevOps Engineer, Stakater, Aurora Solutions
- Ex-Software Engineer, Bentley Systems
- Avid Open Source Contributor



About Me - M. Ali Kahoot



/kahootali



/kahootali

- CKAD, Oracle Cloud Certified & Certified in RedHat Delivery Specialist: Container Platform Deployment
- Blog on Kubernetes with more than 163k views and 2.3k claps on Medium
- Have trained more than 500 resources on DevOps & Kubernetes
- Certified Expert in DevOps, Docker, Kubernetes from Pluralsight
- Public Speaker at DevOpsCon Singapore, Google Devfest, S&P DevOps Week, Data on Kubernetes & conducted more than 10 Bootcamps on Kubernetes for worldwide professionals

About You

- Name
- Field
- Experience
- Technologies you have worked on
- What are your expectations from this course?

Course Agenda

<https://diceanalytics.pk/school/courses-and-workshops/devops-live-training/>

COMMUNICATION

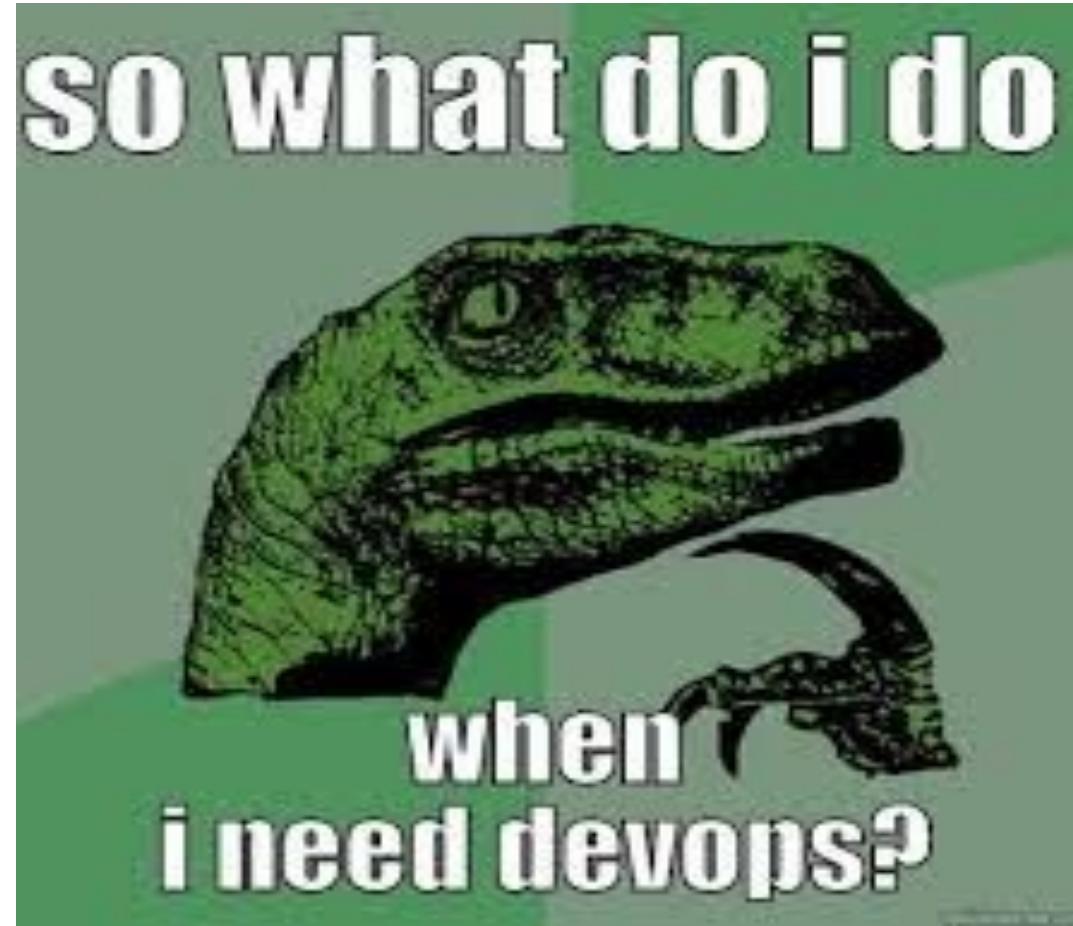
- LMS
- QUESTIONS
- LABS
- SLACK
- ASSIGNMENTS

Disclaimer

These slides are made with a lot of effort, so it is a humble request not to share it with any one or reproduce in any way.

All content including the slides is the property of Muhammad Ali
Kahoot & Dice Analytics

What is DevOps?



Problem

- Everything has/needs Software nowadays
- Software has to run on a server
- A complete cycle of how code in your repo gets deployed to the server
- A bottleneck in deploying new features frequently
- Errors in service, then have to complete the lifecycle again
- Diagnose Issues on servers
- Shifting Blames

Problem

- Delays when deploying to Dev, waiting for confirmation, moving to QA, again manual confirmation then moving to Prod.
- Can take days to release a new version of software
- Then customer feedback, and again make changes and whole process is repeated

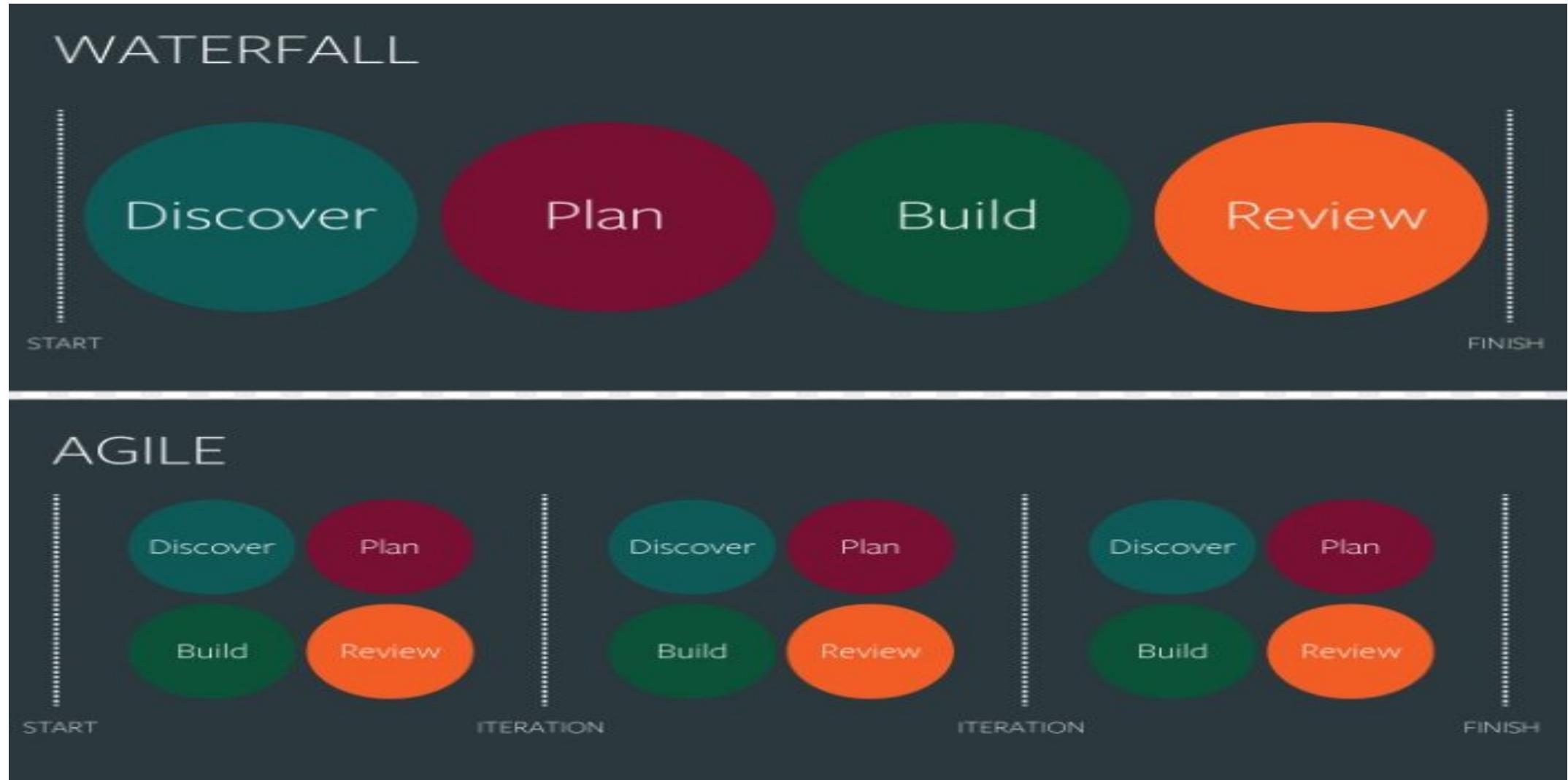
Problem



Result

- Product delivery cycles continue to move slower and slower
- Fewer and less ambitious projects are undertaken
- The organization is no longer able to provide stable, reliable service to customers.

SDLC



SDLC

From <https://analyze.co.za/the-transition-to-devops/>

PROJECT EXECUTION METHODOLOGIES – THE CHANGE

WATERFALL



AGILE

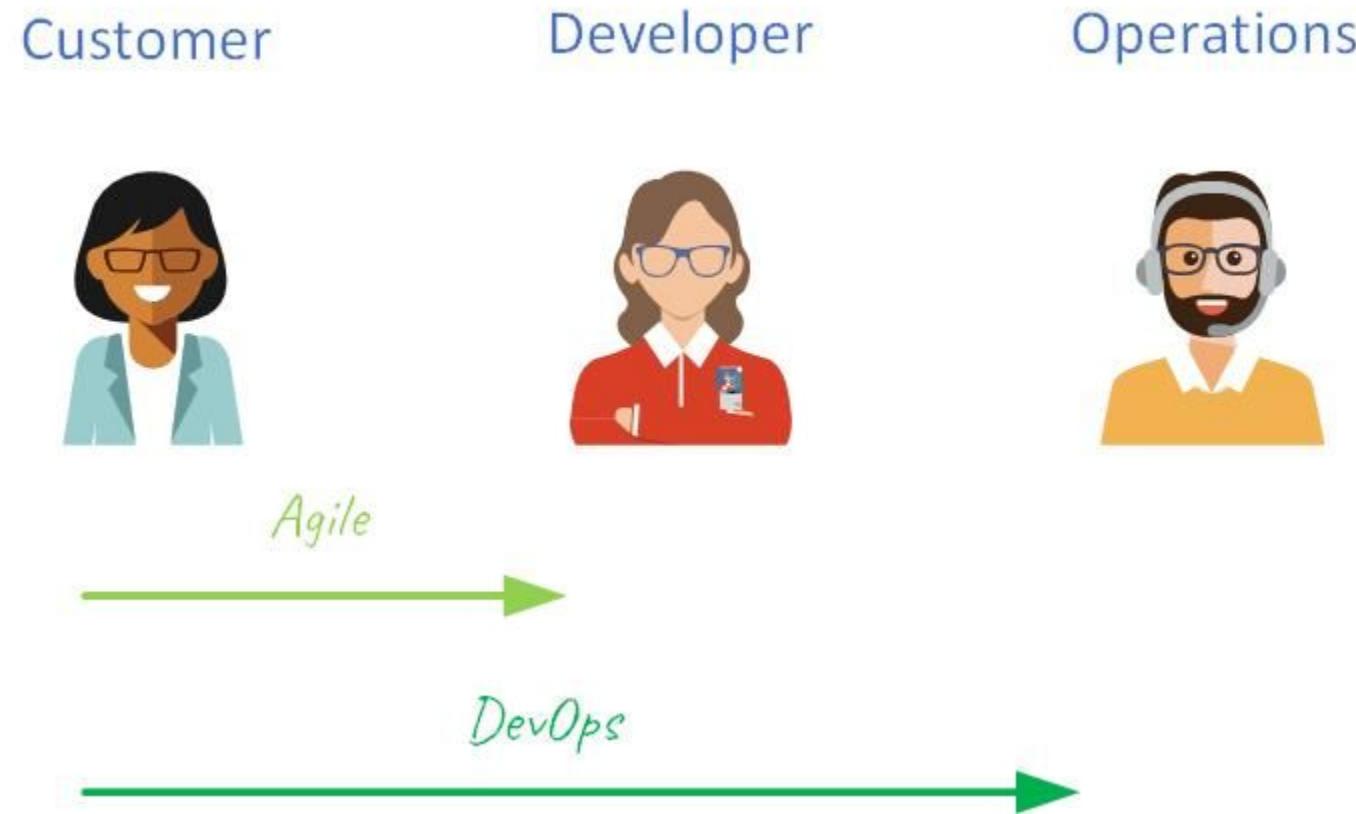


DEVOPS



SDLC

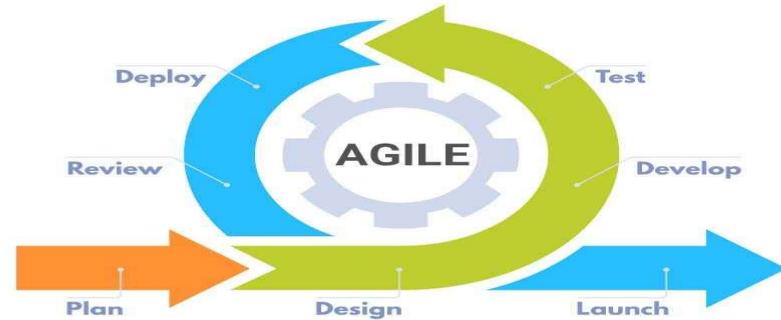
From <https://www.smexdigital.com/blog/agile/agile-vs-devops/>



Agile vs DevOps

From <https://reqtest.com/agile-blog/agile-vs-devops/>

AGILE VS DEVOPS - UNDERSTAND THE DIFFERENCE!



VS



DevOps

Continuous feedback and quick release/fail process

If code is correct, release and take feedback from stakeholder

If error, fail fast and give feedback to developer

Evolution of DevOps

THE AGILE MANIFESTO

Created in 2001 as lightweight set of values and principles against heavyweight software development processes

Relevant key principles:

- small batch sizes
- incremental releases instead of large, waterfall releases
- small, self-motivated teams.

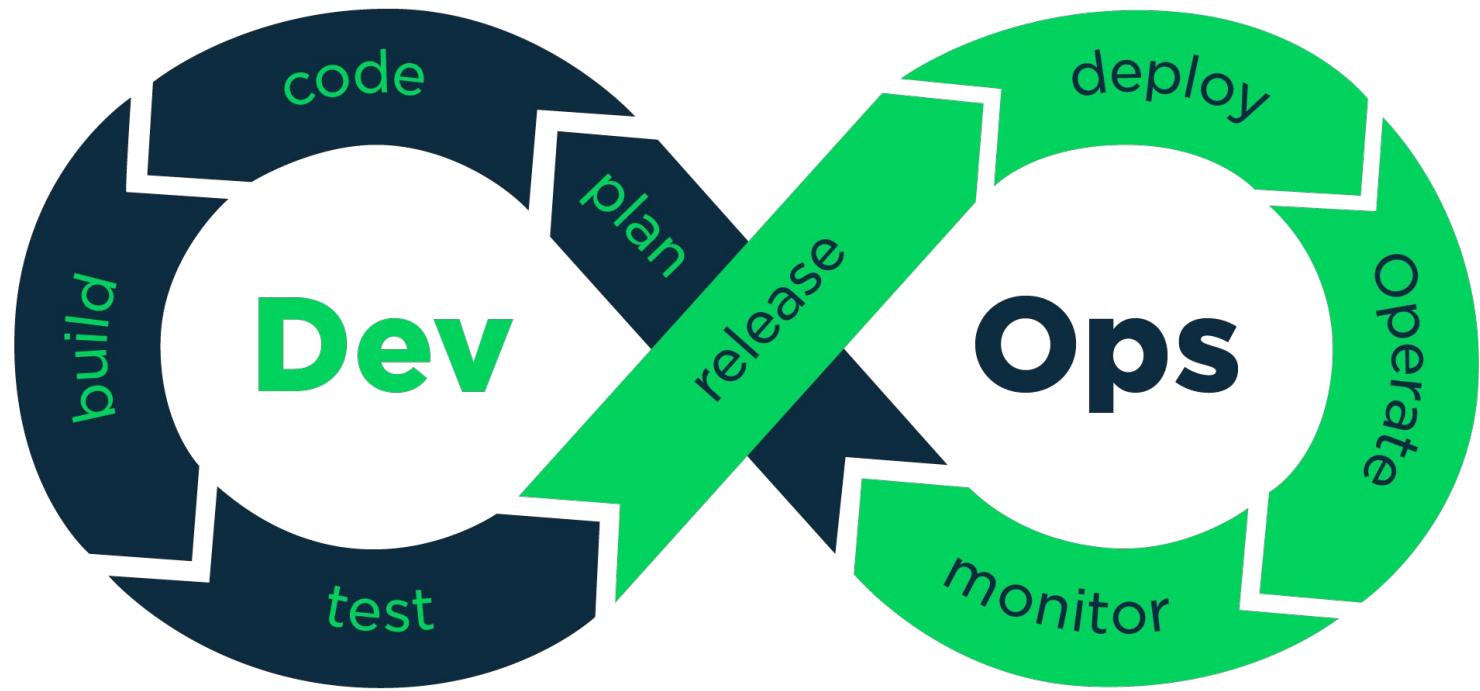
Evolution of DevOps

THE CONTINUOUS DELIVERY MOVEMENT

2006 - 2009

- Extending the concepts of continuous build, test, and integration, to continuous delivery
- “deployment pipeline”. Ensure that code and infrastructure are always in a deployable state
- code checked in to trunk can be safely deployed into production.

What is DevOps?



DevOps

DevOps is a combination of software development (Dev) and information technology operations (Ops). DevOps is a set of software development practices that aim to shorten the systems development life cycle while delivering features, fixes, and updates frequently in close alignment with business objectives.

Wikipedia

DevOps

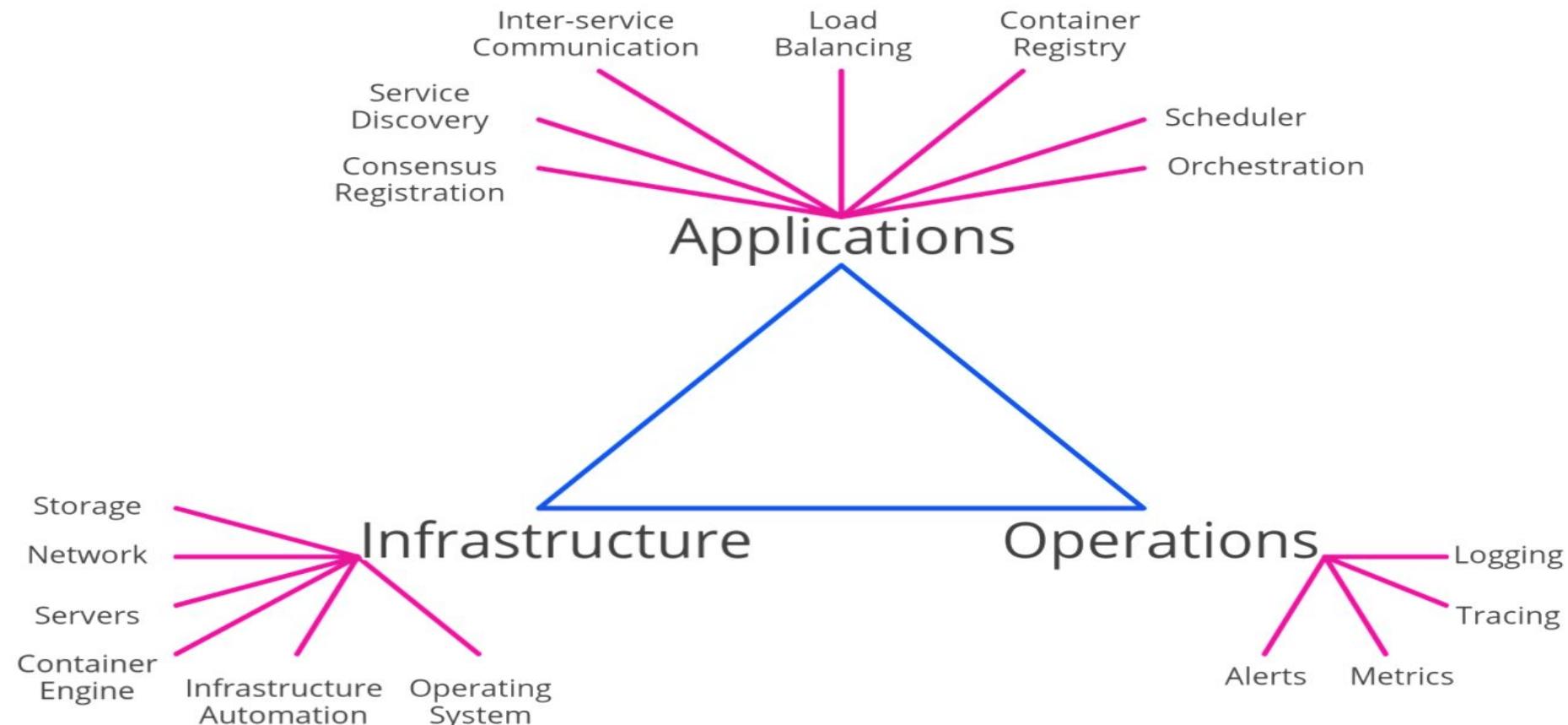
DEVOPS IS THE COMBINATION OF CULTURAL PHILOSOPHIES, PRACTICES, AND TOOLS THAT INCREASES AN ORGANIZATION'S ABILITY TO DELIVER APPLICATIONS AND SERVICES AT HIGH VELOCITY: EVOLVING AND IMPROVING PRODUCTS AT A FASTER PACE THAN ORGANIZATIONS USING TRADITIONAL SOFTWARE DEVELOPMENT AND INFRASTRUCTURE MANAGEMENT PROCESSES. THIS SPEED ENABLES ORGANIZATIONS TO BETTER SERVE THEIR CUSTOMERS AND COMPETE MORE EFFECTIVELY IN THE MARKET.

AWS

DevOps - My Definition

DevOps ultimately means deploying and maintaining Correct & Error-Free code from a developer's machine, on a Scalable & Secure Infrastructure, to revenue generating prod awesomeness!

DevOps Engineer Job Responsibilities



Benefit of DevOps

Speed

- Fast feedback loops at every step of the process

Rapid Delivery

- Increase the frequency and pace of releases and bug fixes

Reliability

- Fast automated tests are run in production-like environments
- No firefighting for days or weeks

Delegation

- Teams take Responsibility and ownership

Benefit of DevOps

Scale

- Operate infrastructure and development processes at scale

Improved Collaboration

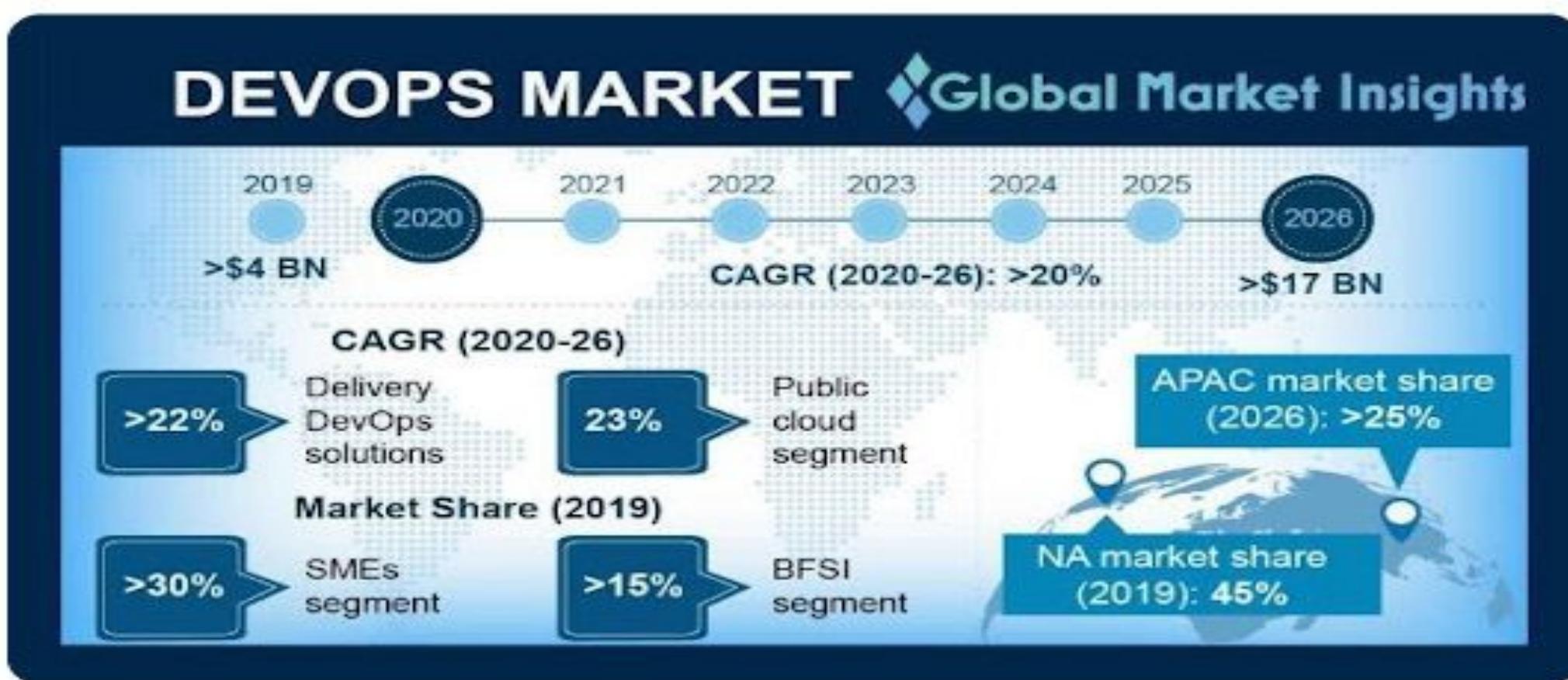
- More effective teams emphasizing values such as ownership and accountability.
- Developers and operations teams collaborate closely reducing inefficiencies and saving time

Security

- Compliant with policies and configuration management

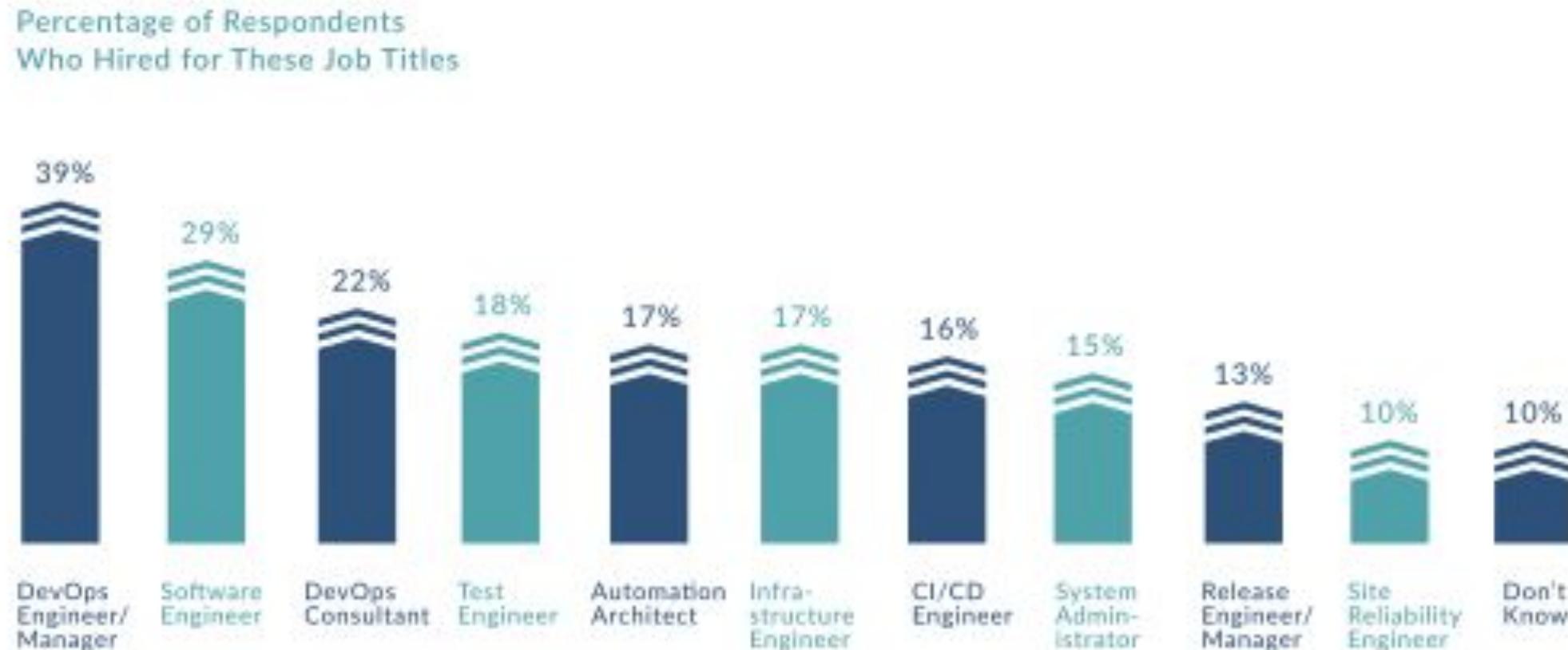
Why Should One Move to DevOps ?

DevOps Trends

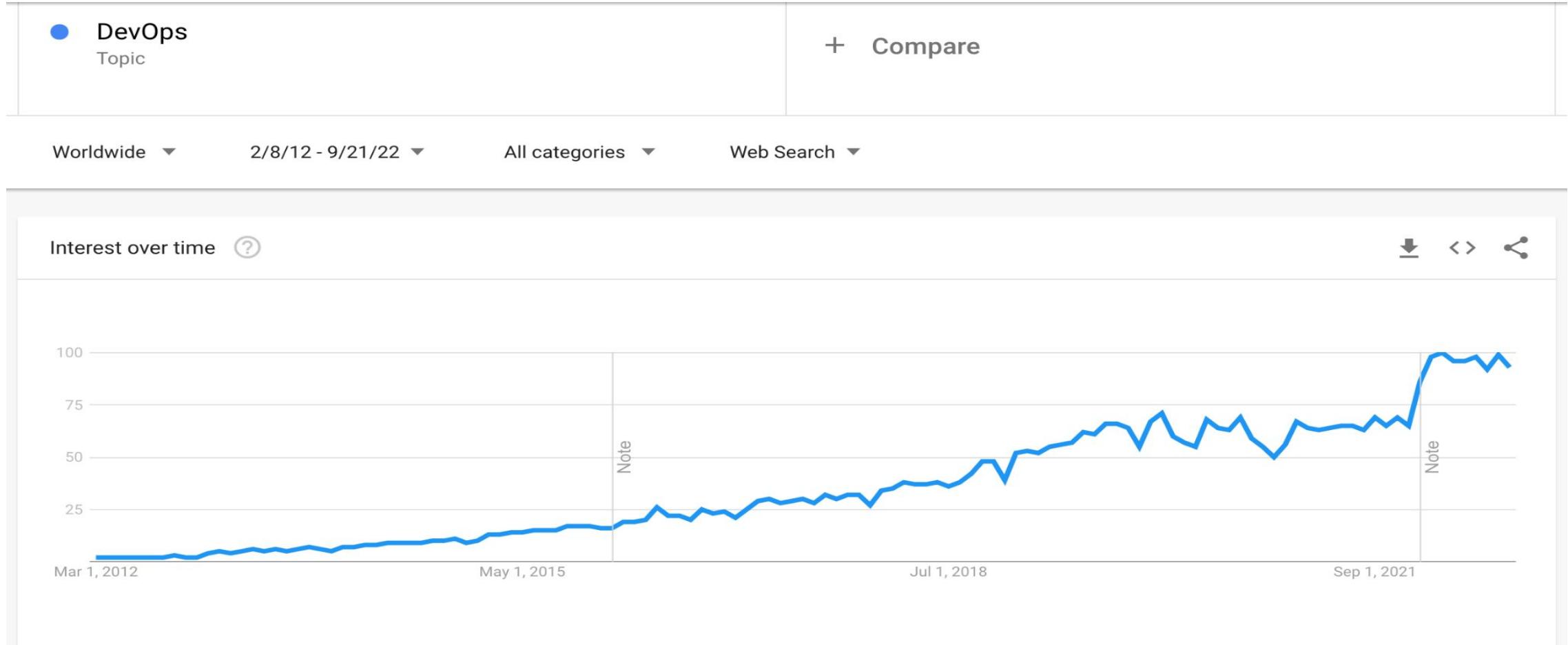


DevOps Trends

From: DevOps Institute | Upskilling: Enterprise DevOps Skills Report 2019



DevOps Trends



DevOps Course By M. Ali Kahoot - Dice Analytics

DevOps Trends

From: <https://www.datacareer.de/blog/devops-engineer-salaries-in-europe-in-2018/>

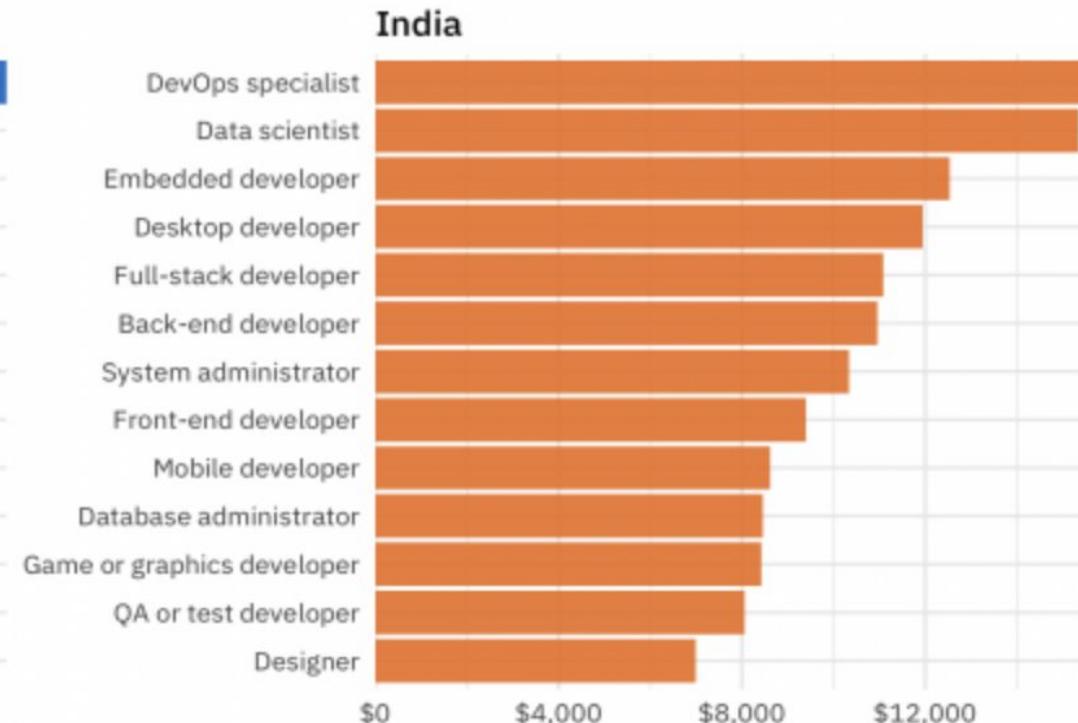
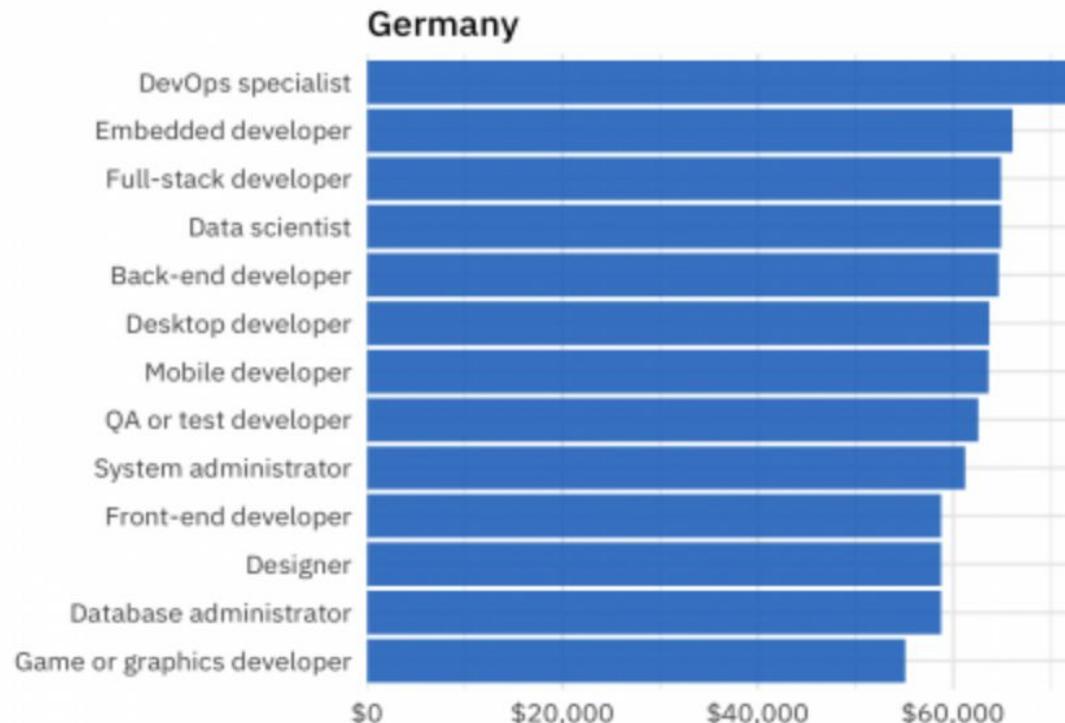
Country	Annual Salary in 1000 EURO	OECD Price level	Adjusted Salary
Italy	26.3	91	28.9
Belgium	34.2	101	33.8
France	42.7	101	42.3
Spain	36.8	83	44.4
Ireland	47.5	102	46.6
United Kingdom	52.4	108	48.5
Netherlands	52.5	103	50.9
Germany	55.3	98	56.5
Switzerland	83.5	142	58.8
Austria	60	101	59.4

DevOps Trends - 2018

From: <https://stackoverflow.blog/2018/09/05/developer-salaries-in-2018-updating-the-stack-overflow-salary-calculator/>

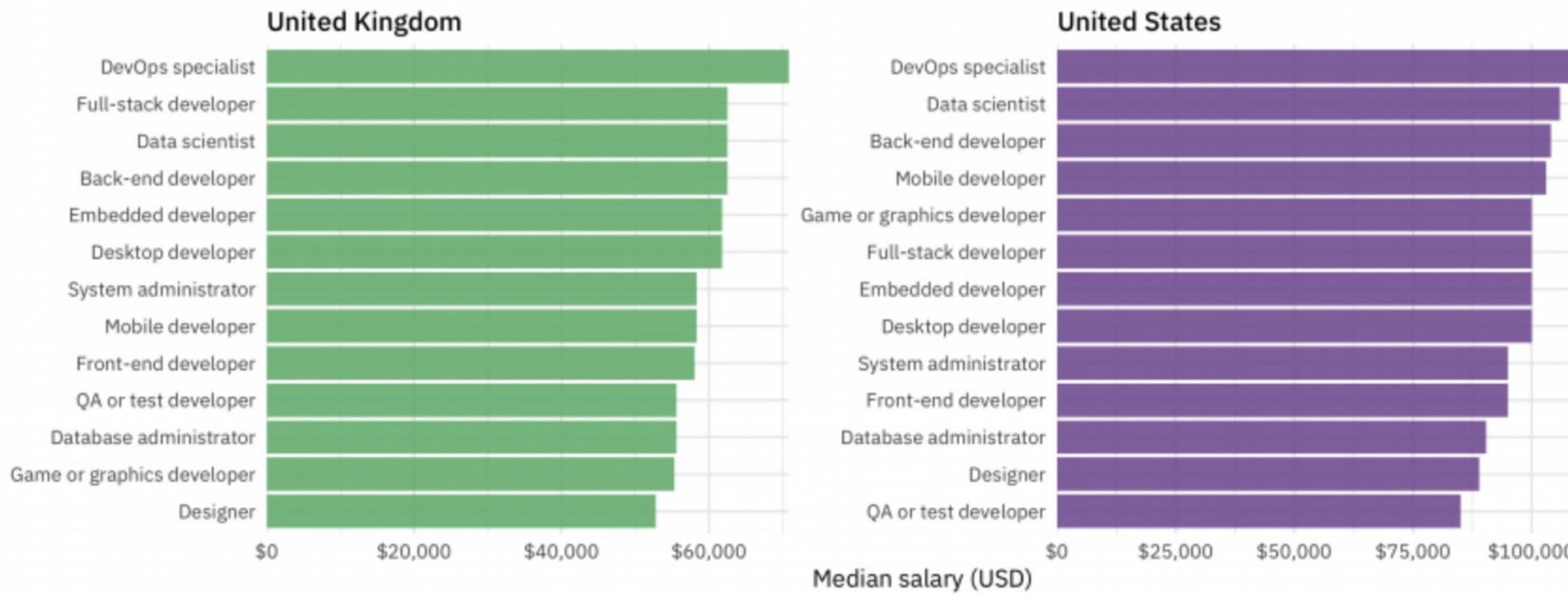
Median salary for different types of developers

Developers working in DevOps are the highest earners



DevOps Trends - 2018

From: <https://stackoverflow.blog/2018/09/05/developer-salaries-in-2018-updating-the-stack-overflow-salary-calculator/>

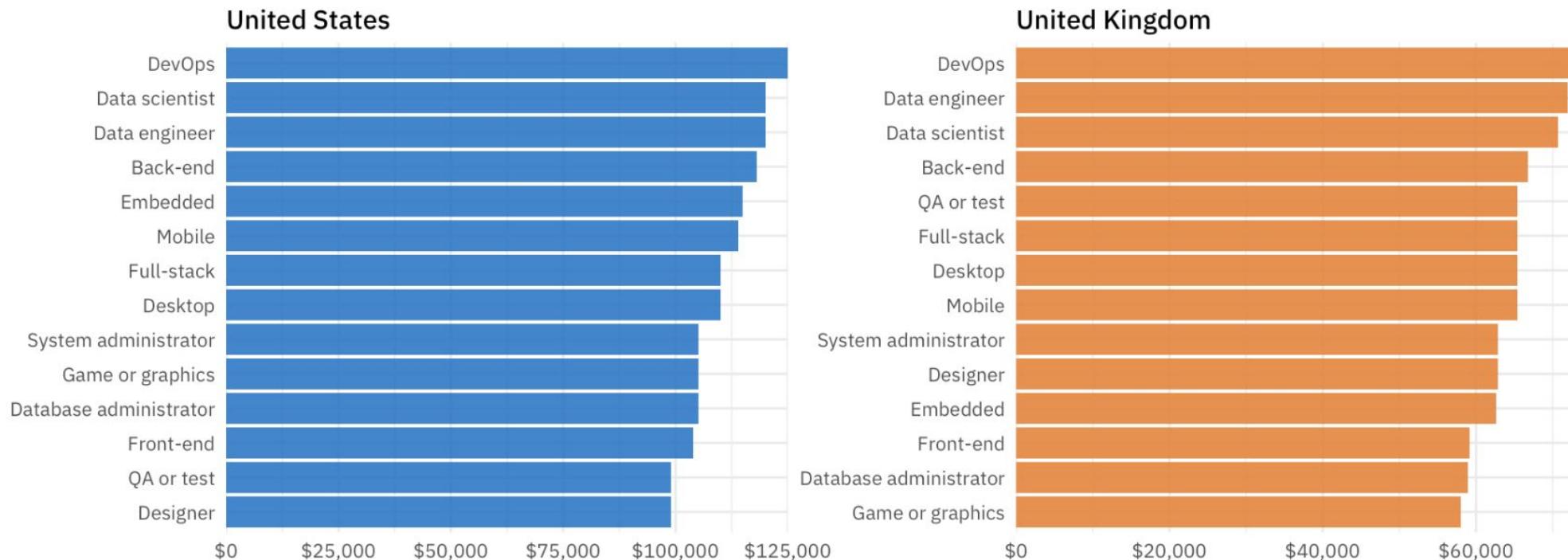


DevOps Trends - 2019

From: <https://stackoverflow.blog/2019/10/16/coding-salaries-in-2019-updating-the-stack-overflow-salary-calculator/>

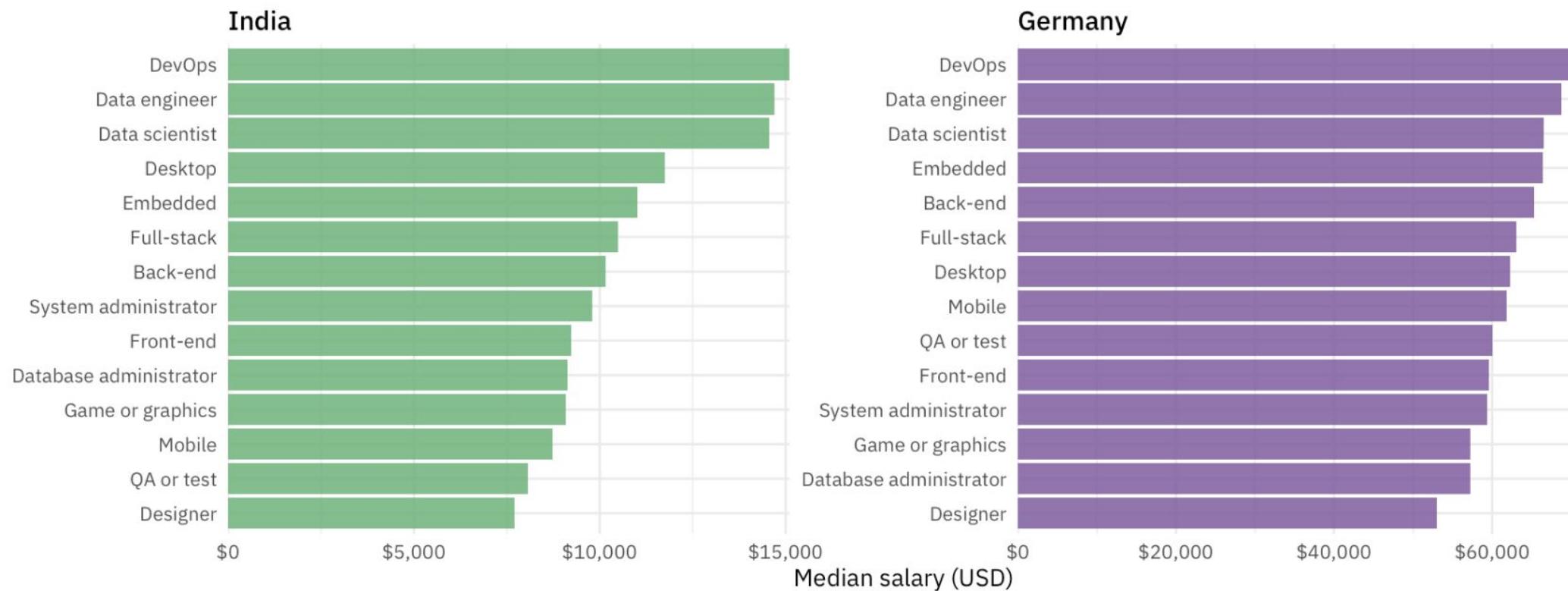
Median salary for different types of developers

Developers working in DevOps are the highest earners



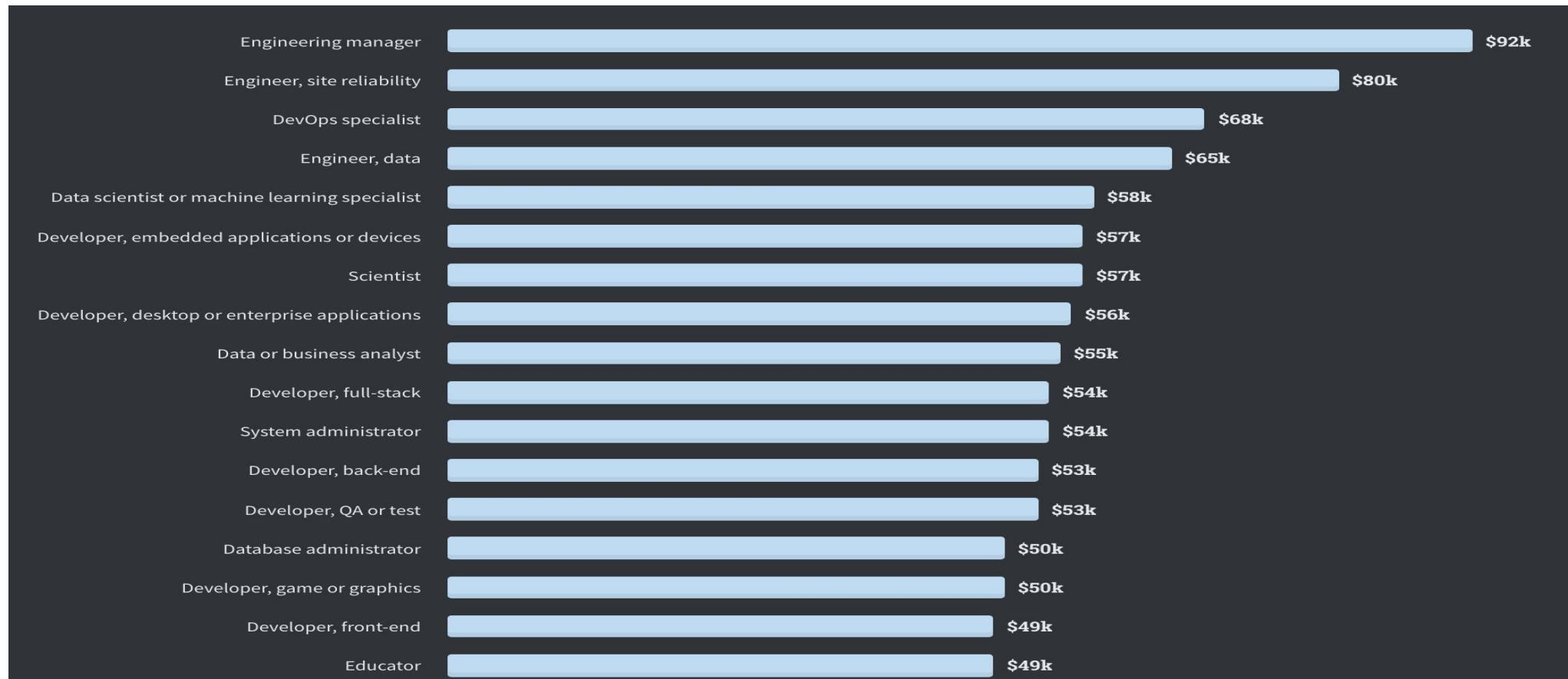
DevOps Trends - 2019

From: <https://stackoverflow.blog/2019/10/16/coding-salaries-in-2019-updating-the-stack-overflow-salary-calculator/>



DevOps Trends - 2020

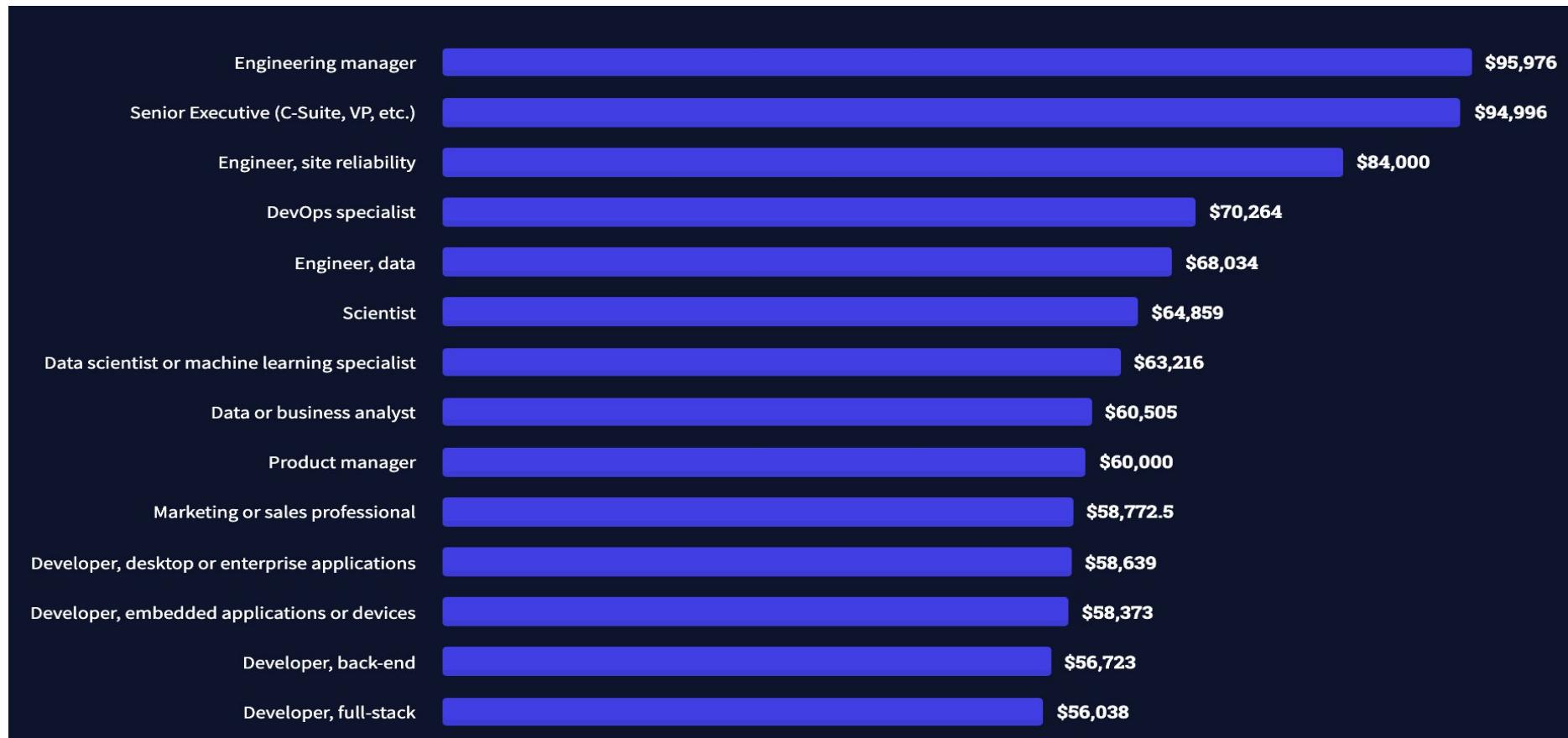
From: <https://insights.stackoverflow.com/survey/2020#work-salary-by-developer-type>



DevOps Trends - 2021

From:

https://insights.stackoverflow.com/survey/2021?_ga=2.181119271.1887520914.1628768285-1278854634.1628351206#work-salary



DevOps Trends

From: https://www.glassdoor.com/Salaries/devops-engineer-salary-SRCH_KO0,15.htm

How much does a Devops Engineer make?

Experience

All years of Experience

Industry

All industries

\$131,600 /yr

Total Pay

\$103,537 /yr

Base Pay

\$28,062 /yr

Additional Pay

\$131,600 /yr

\$104K

\$168K

\$85K \$208K

■ Most Likely Range ■ Possible Range



Total Pay Trajectory

For Devops Engineer

\$131,600 /yr
Devops Engineer

\$159,317 /yr
Senior Devops Engineer

\$157,894 /yr
Lead Devops Engineer

[See Full Career Path >](#)

[Download as data table](#)

Related Job Titles

For Devops Engineer

Salary Benchmarking: Technical Roles

734 Other Technical Senior Level Professionals



DevOps Trends - Hardest Job to Fill

DevOps vacancies will be the 'hardest to fill' in 2021

By [Sead Fadilpašić](#) published January 15, 2021

The abrupt move to remote working triggered a sudden rise in demand for DevOps experts.

DevOps Engineer: #1 Hardest Job to Fill

DevOps

References:

- <https://www.information-age.com/devops-positions-hardest-fill-2021-say-hr-professionals-123493389/>
- <https://www.logicworks.com/blog/2016/06/devops-engineer-hardest-job-find-skills-shortage/>
- <https://www.itproportal.com/news/devops-vacancies-will-be-the-hardest-to-fill-in-2021/>

DevOps positions to be hardest to fill in 2021, say HR professionals

The 4th annual Developer Report from CodinGame has revealed that DevOps positions could be the hardest to fill in 2021, after surveying HR professionals

DevOps Trends

Recent Acquisitions

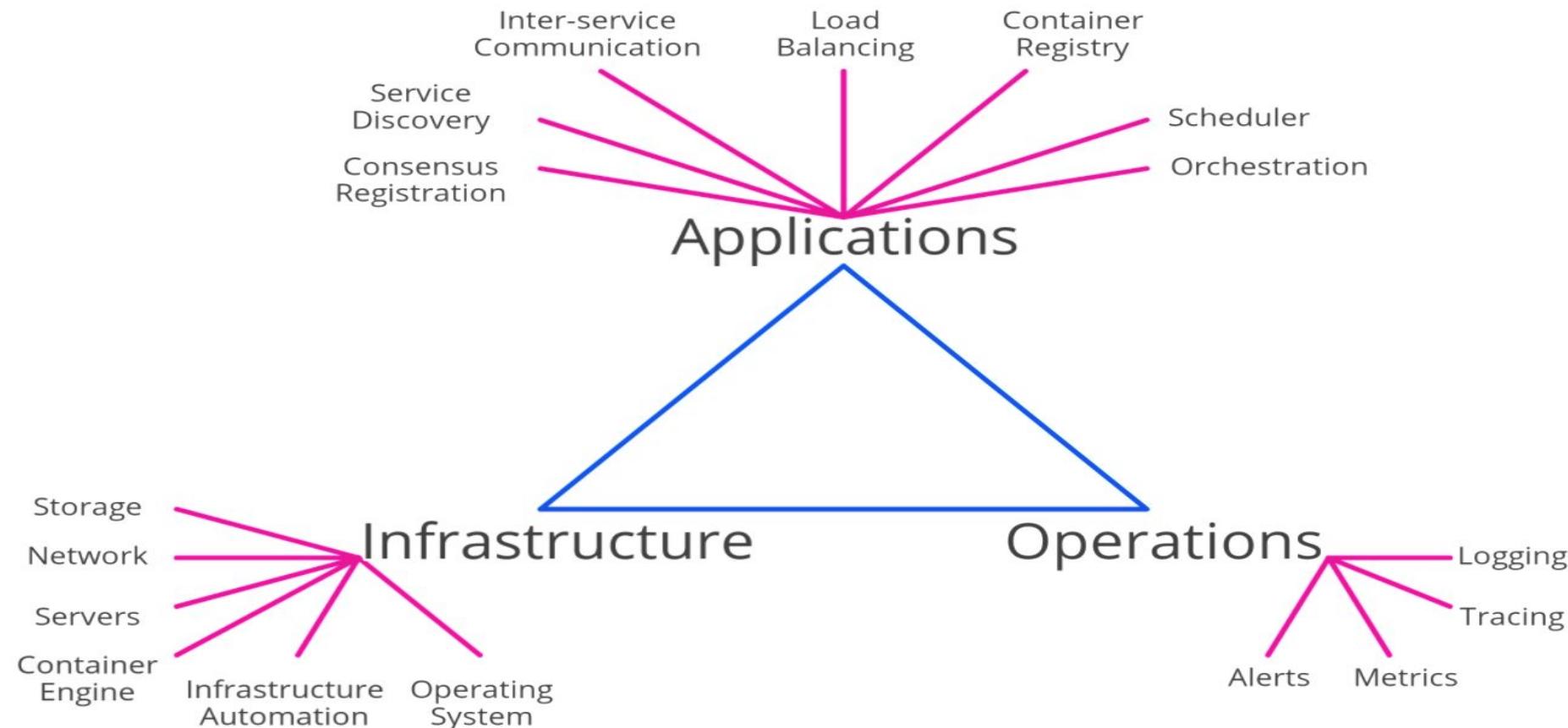
- Microsoft acquired Github
- IBM acquired Redhat
- Mirantis acquired Docker

How to Become a DevOps Hackerman



DevOps Course By M. Ali Kahoot - Dice Analytics

DevOps Engineer Job Responsibilities



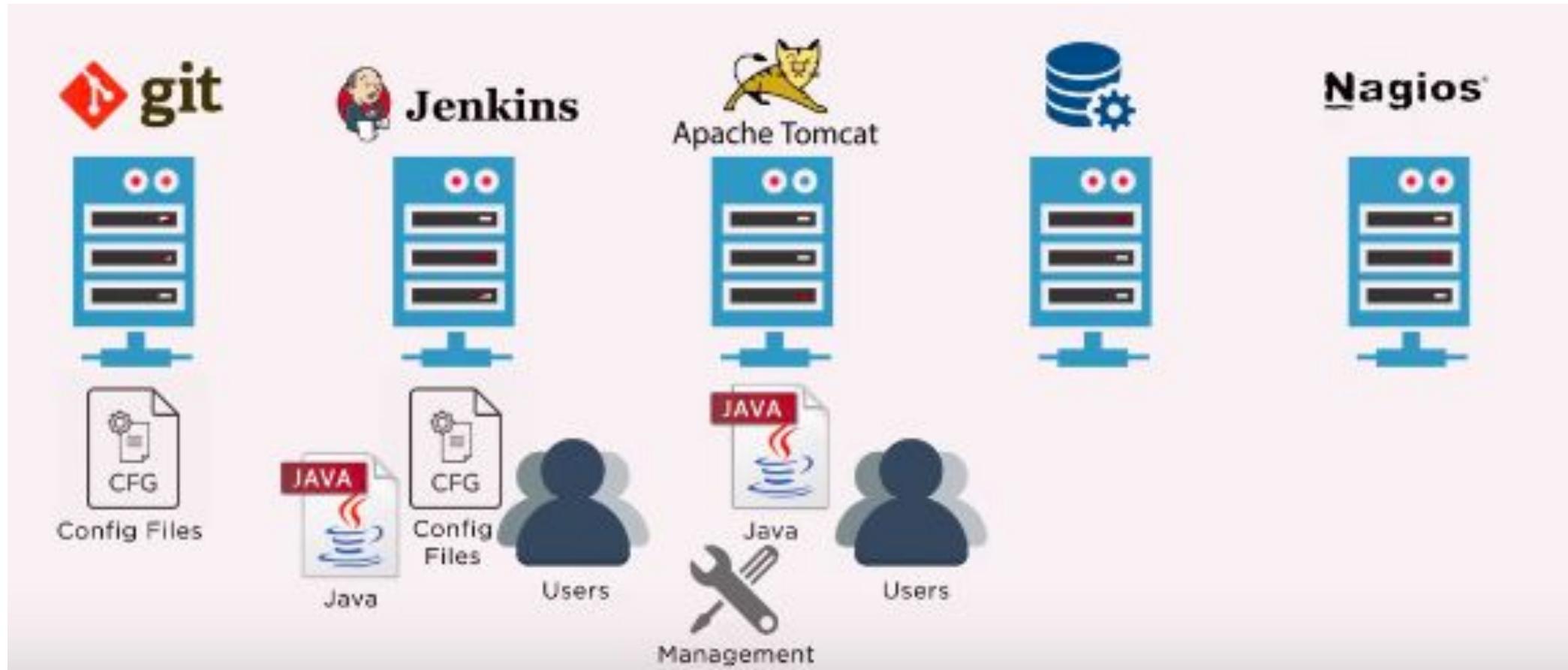
DevOps Best Practices

- Infrastructure as Code
- Cloud Computing
- Microservices
- Continuous Integration
- Continuous Delivery/Continuous Deployment
- Monitoring and Logging
- Communication and Feedback

Infrastructure as Code

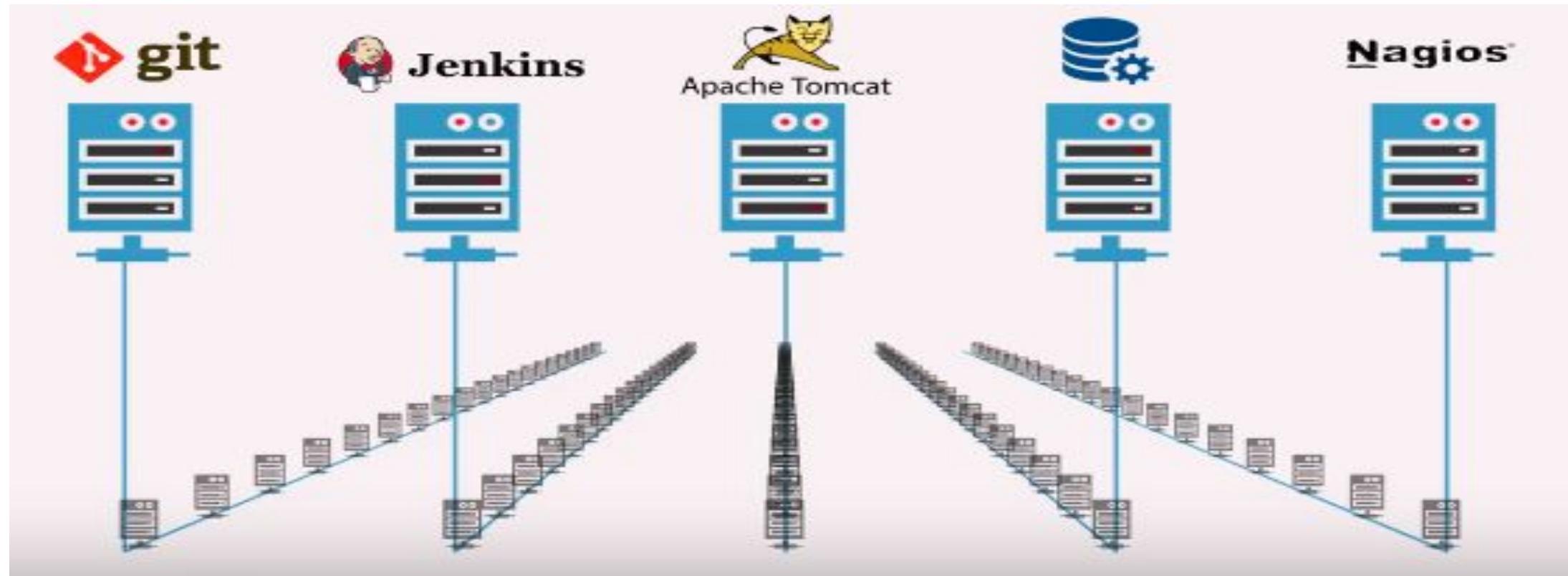
- Need infrastructure to deploy applications
- Can be on-prem or cloud
- How to create 100s of VMs
- Infrastructure and servers can quickly be created or recreated using standardized patterns.
- Easier to govern changes in infrastructure resources

Infrastructure as Code



Infrastructure as Code

- Automation across multiple envs



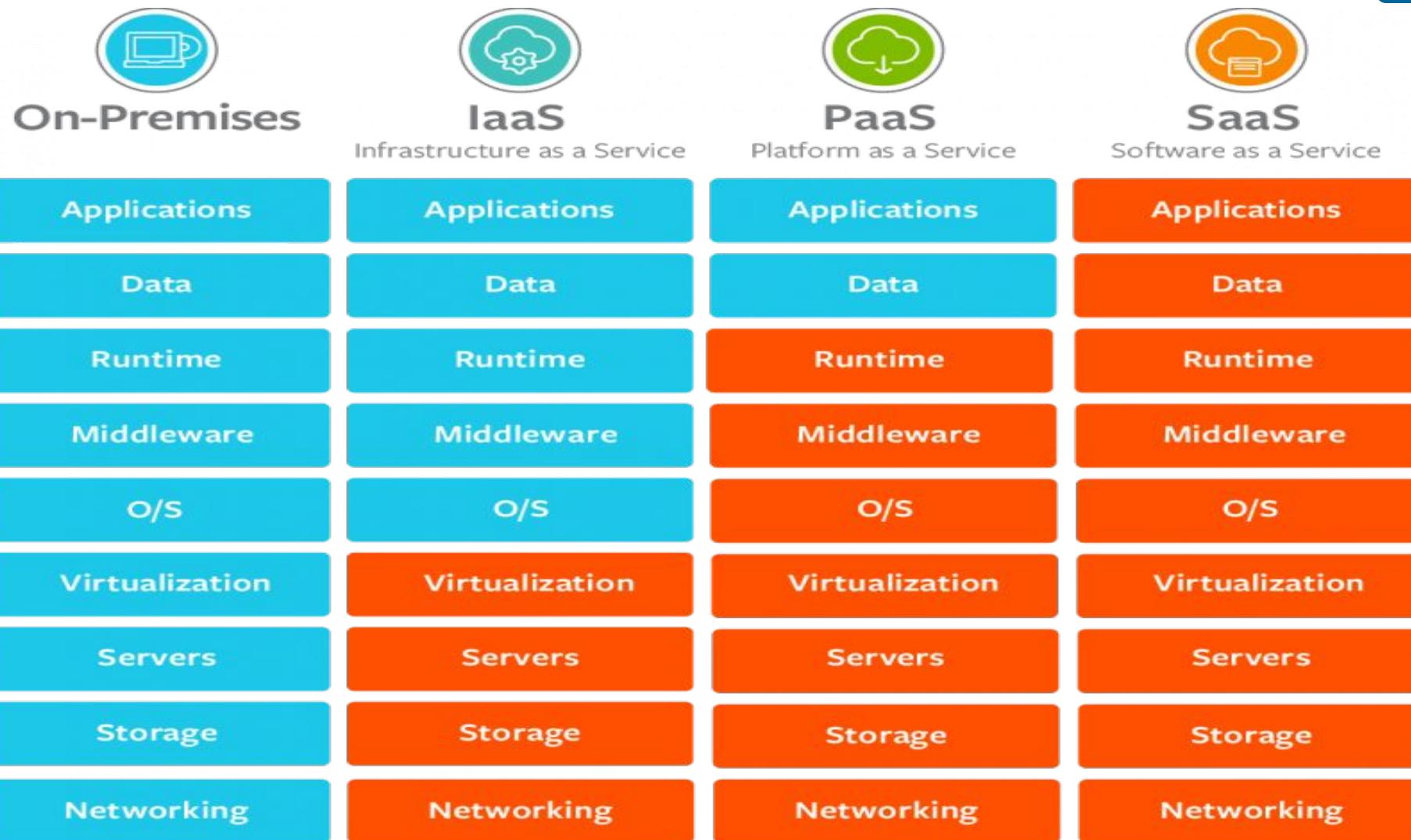
Infrastructure as Code

- Recreation of environment
- Deploy everything in an automated way or through code
- Main purpose is to have state & deployment manifests in any central place
- Terraform, Ansible, Puppet, Chef
- Lead to GitOps

Cloud Computing

- On-demand computing resources
- Don't have to buy resources
- Elastic resources - Scale up or down quickly and easily to meet demand
- Pay for what you use only
- Self service - All the IT resources you need with self-service access

From <https://www.bmc.com/blogs/saas-vs-paas-vs-iaas-whats-the-difference-and-how-to-choose/>



You Manage

Other Manages

DevOps Course By M. Ali Kahoot - Dice Analytics

Examples

YOUR OWN CAR

On-premises solution



LEASED CAR

IaaS



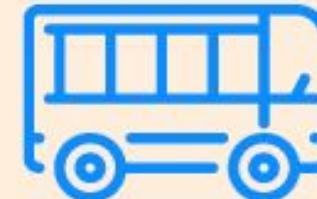
TAXI

PaaS



BUS

SaaS



Examples

- IAAS: Google Compute Engine, Amazon EC2, Rackspace, etc
- PaaS: Google App Engine, GKE Cloud Foundry, Engine Yard, EKS, AKS etc.
- SaaS: Salesforce, Google Docs, Office 365, Basecamp, Facebook etc

FaaS/Serverless

FaaS: Function as a Service

Can run a function in cloud

Don't need to worry of platform

Platform scaling and other issues managed by Provider

Google Cloud Functions, Azure Functions, AWS lambda, AWS Fargate, Azure Container Instance etc

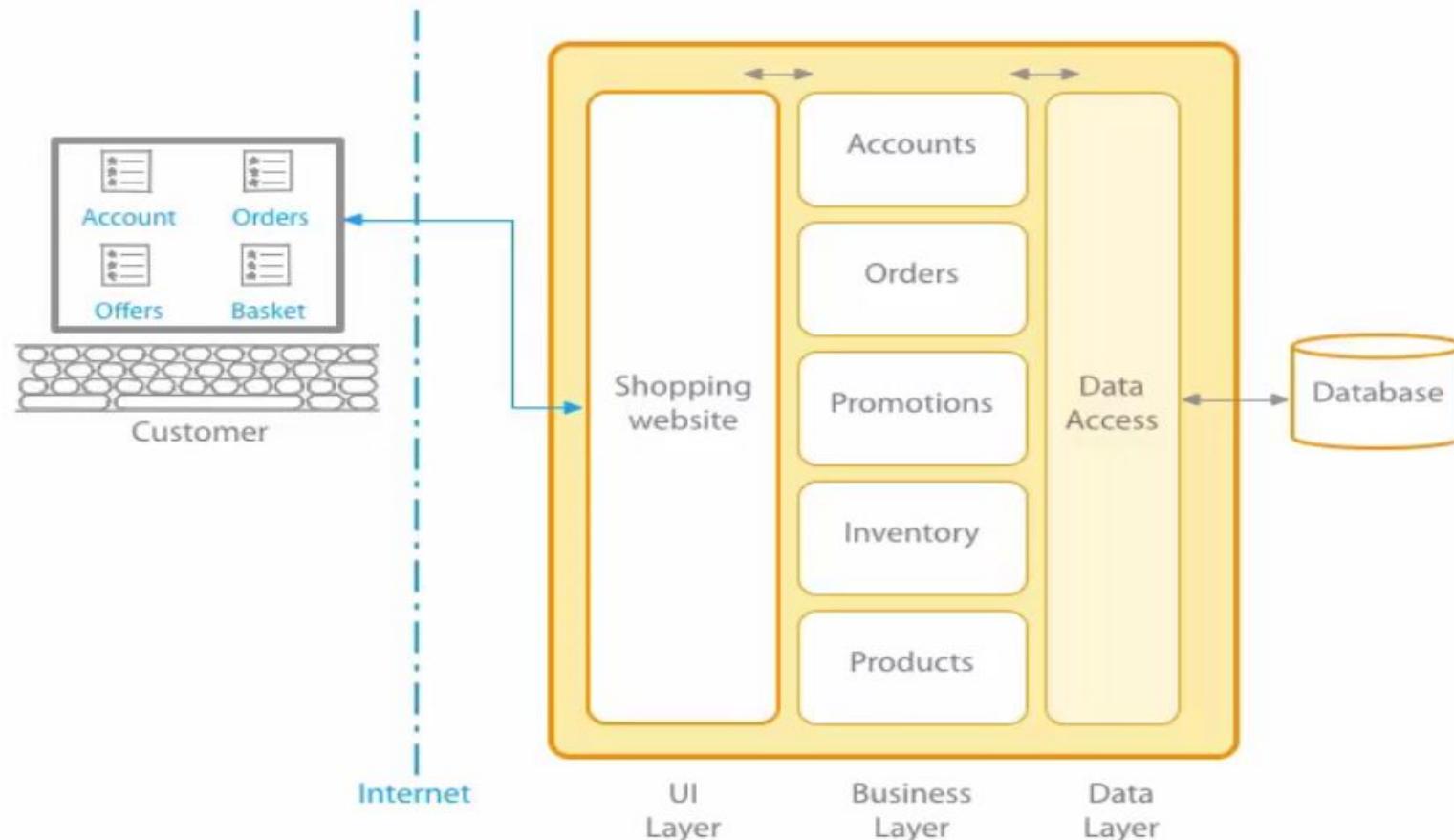
Clouds

- Google Cloud
- AWS
- Azure
- Oracle
- IBM Cloud
- Digital Ocean
- Openstack

Microservices

- Design approach to build a single application as a set of small services.
- Each service runs in its own process and communicates with other services through a well-defined interface.
- Each service can be updated and deployed independently, decreasing risk of update and impact of errors.
- Separate services meaning
 - Independent CI/CD pipelines
 - Independent Ownership
 - Independent Responsibility

Monolithic Service



Concerns of Monolithic

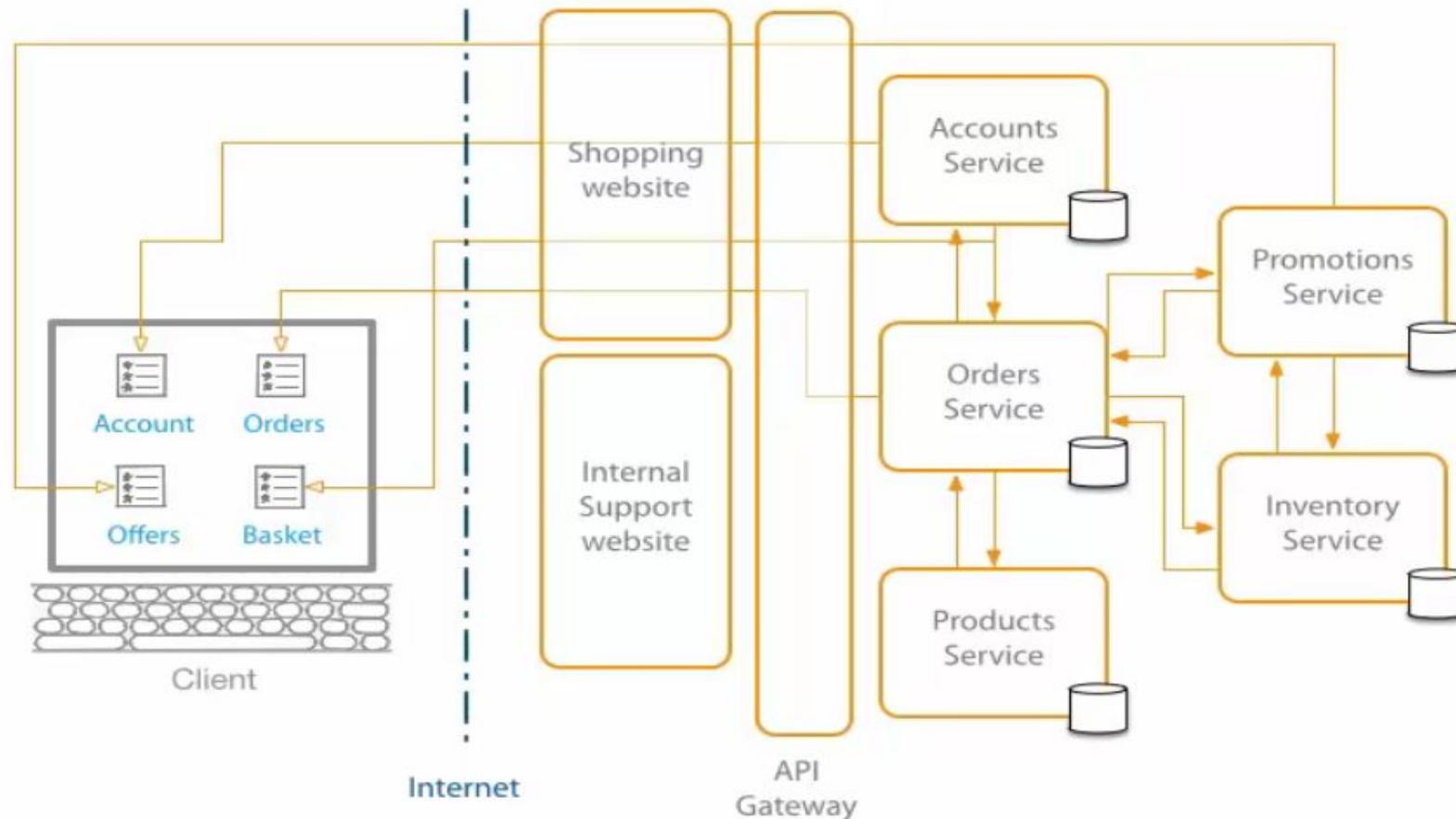
- Difficult to Scale
- Long Time to Ship
- Complexities of Growing Applications
- No Clear Ownership
- Failure Cascade
- Wall Between Dev and Ops
- Stuck in a Technology/Language

Deciding a Microservice?

Factors deciding if it should be a Microservice

- Multiple Rates of Change
- Independent Life Cycles
- Independent Scalability
- Isolated Failure
- Technology/Language stack
- Separate Team/Ownership

Microservices



Microservices

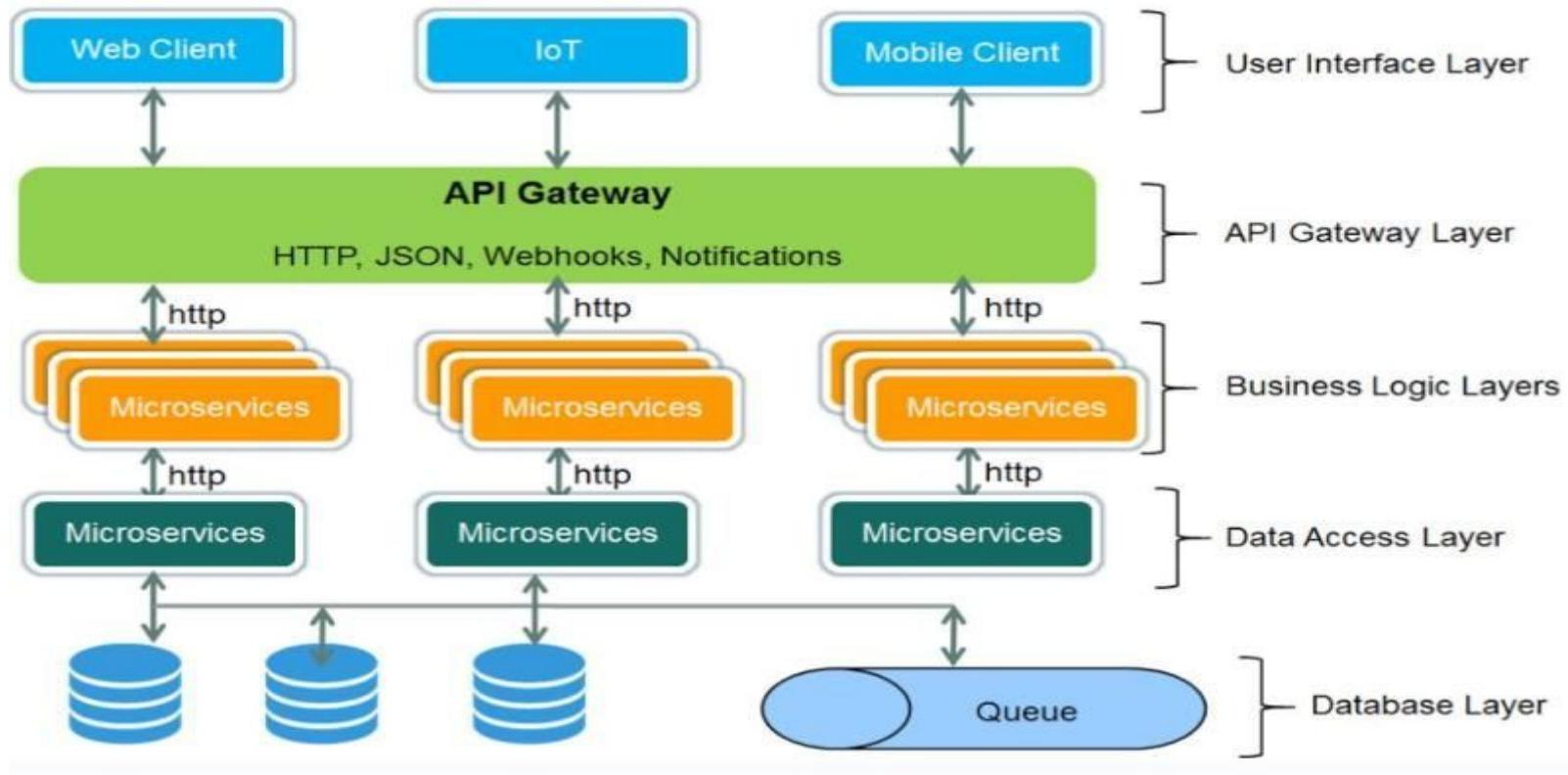
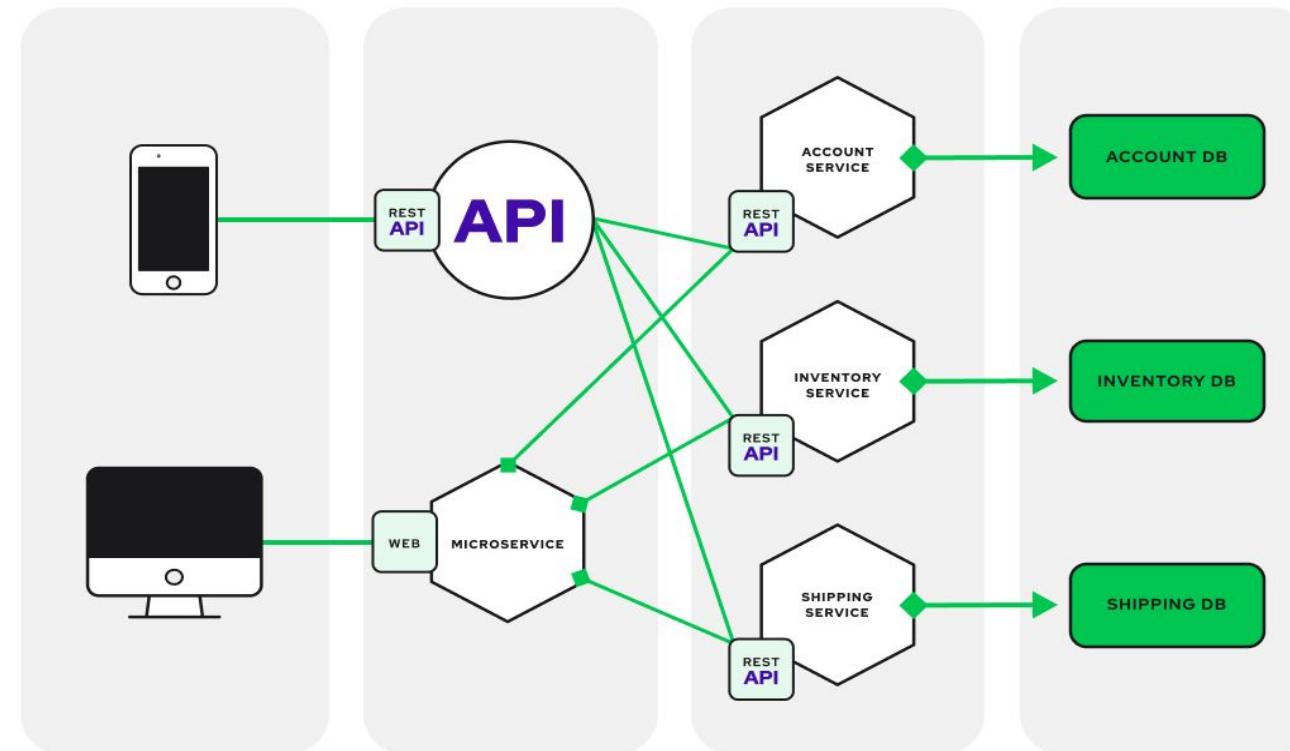


Image Credits: Arun Kottoli

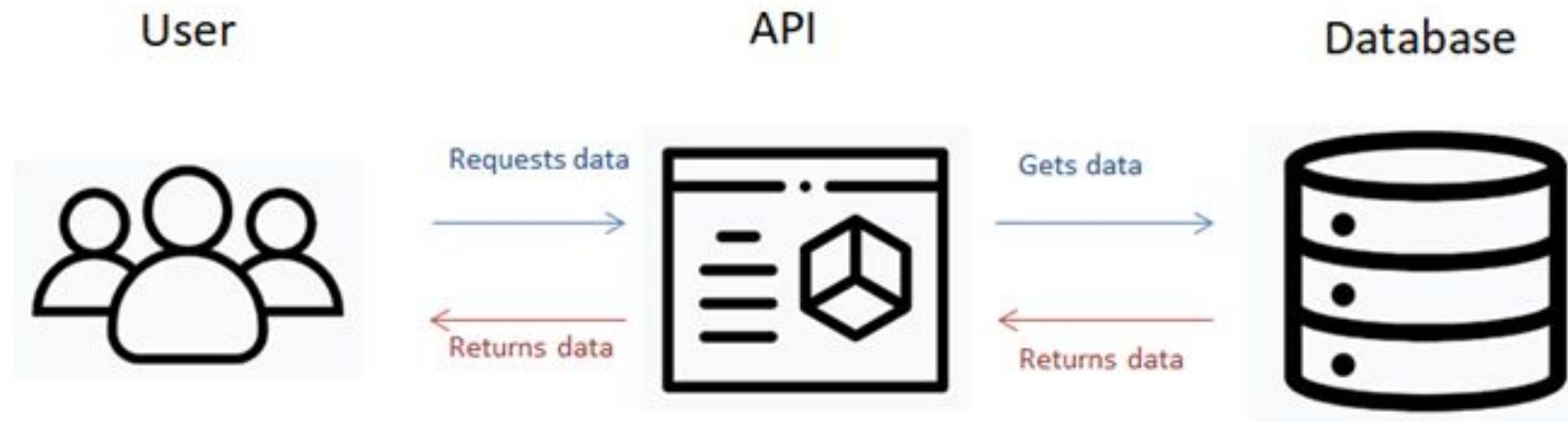
Microservices

From: <https://uestorefront.io/microservices>



APIs

From: <https://medium.com/nerd-for-tech/what-are-apis-anyway-13d88bf97565>



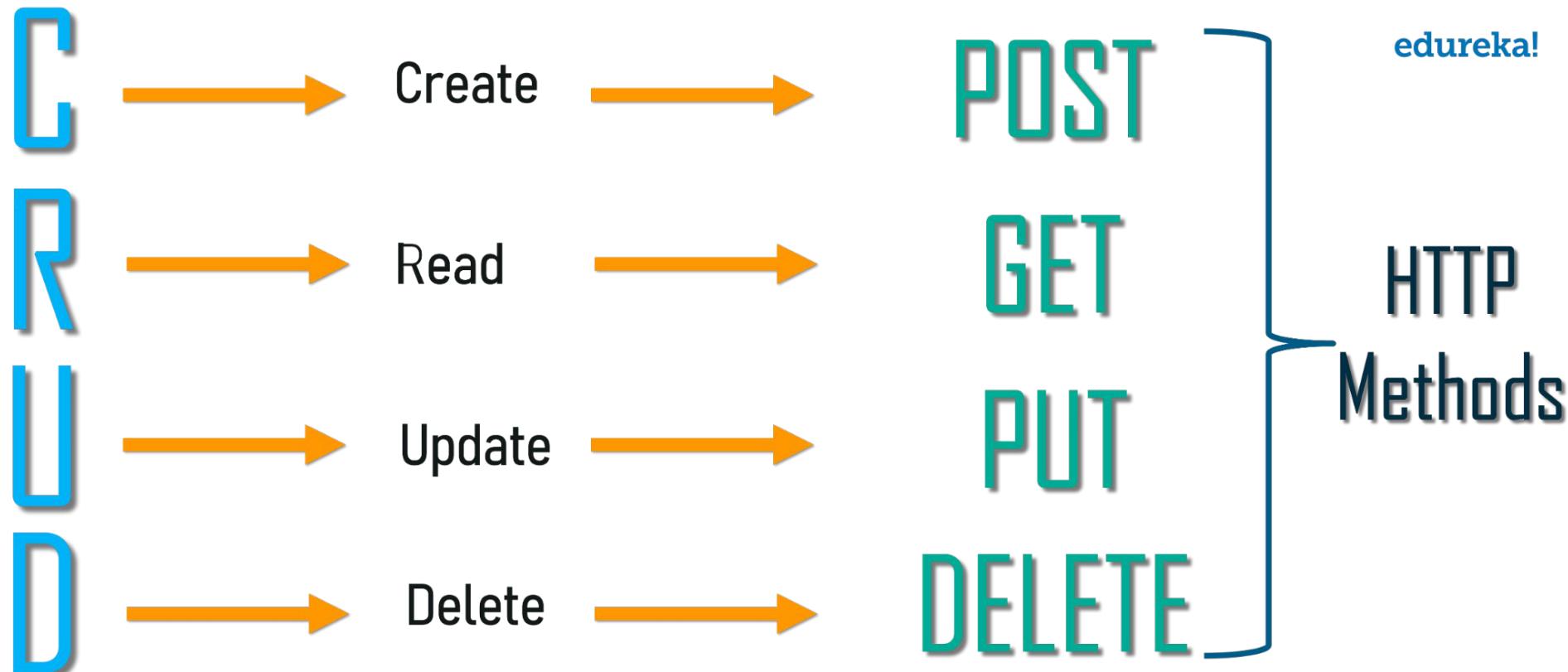
APIs



DevOps Course By M. Ali Kahoot - Dice Analytics

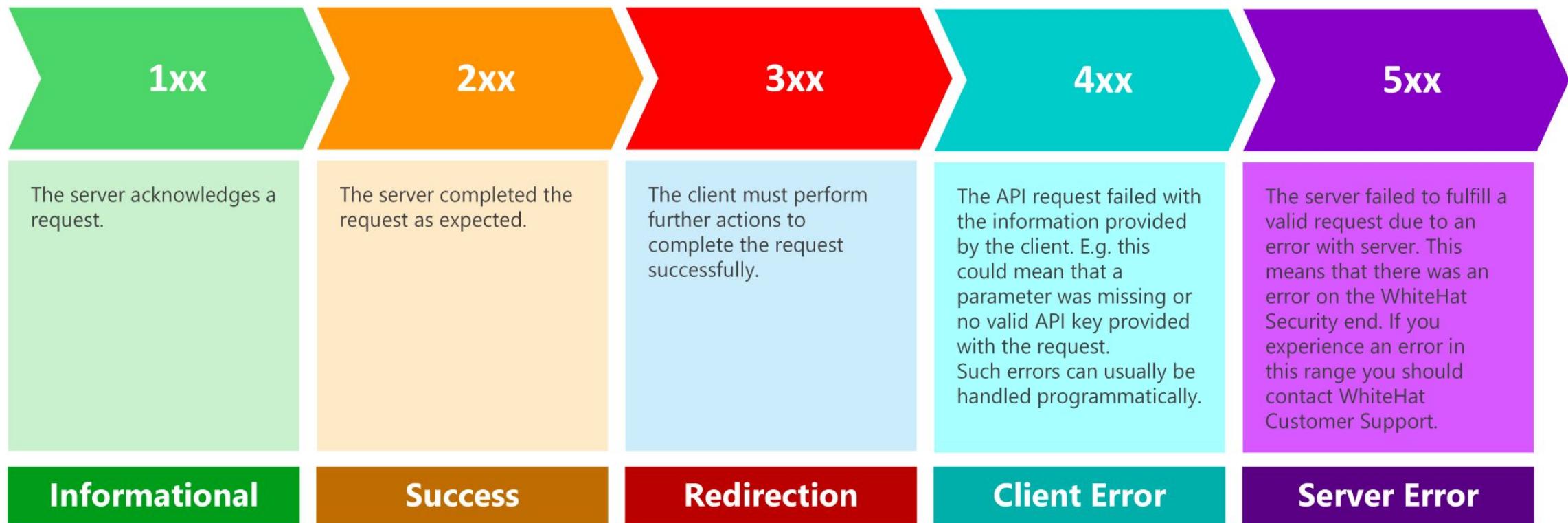
APIs

From: <https://www.edureka.co/blog/what-is-rest-api/>



APIs

From: <https://apidocs.whitehatsec.com/whs/docs/error-handling>



API Examples

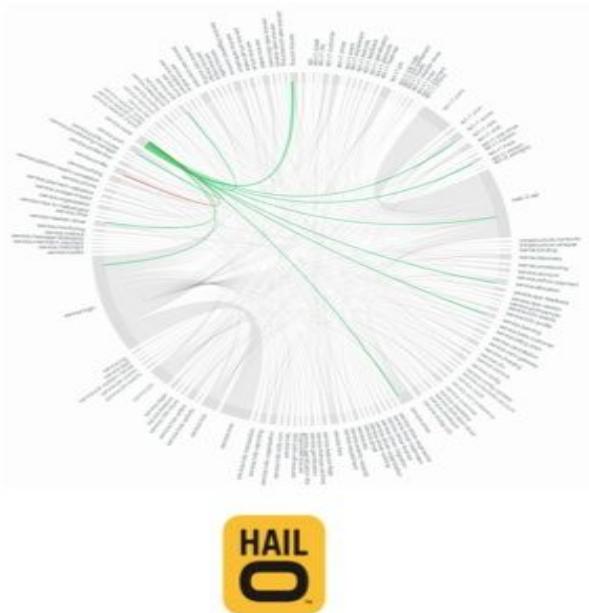
- <https://catfact.ninja/breeds>
- <https://api.coindesk.com/v1/bpi/currentprice.json>
- <https://api2.binance.com/api/v3/ticker/24hr>
- <https://datausa.io/api/data?drilldowns=Nation&measures=Population>
- https://official-joke-api.appspot.com/random_joke

API Examples

- <https://api.alquran.cloud/v1/page/1/quran-uthmani>
- <https://makeup-api.herokuapp.com/api/v1/products.json?brand=maybelline>
- <https://randomuser.me/api/>
- <https://www.foodpanda.pk/>

Examples

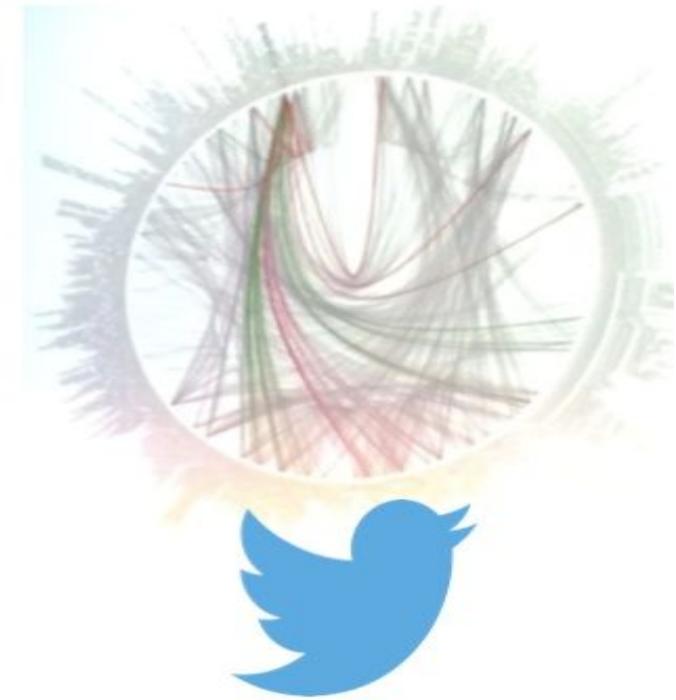
450 microservices



500+ microservices



500+ microservices



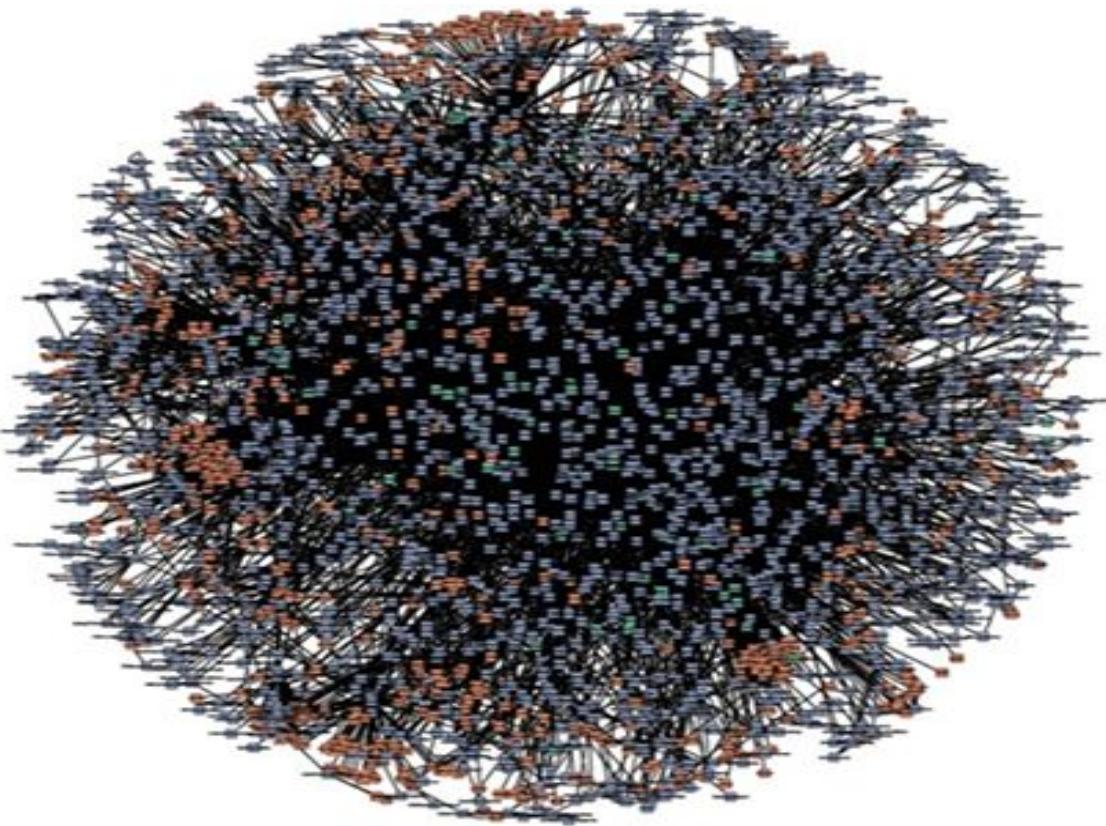
Source:

Netflix: <http://www.slideshare.net/BruceWong3/the-case-for-chaos>

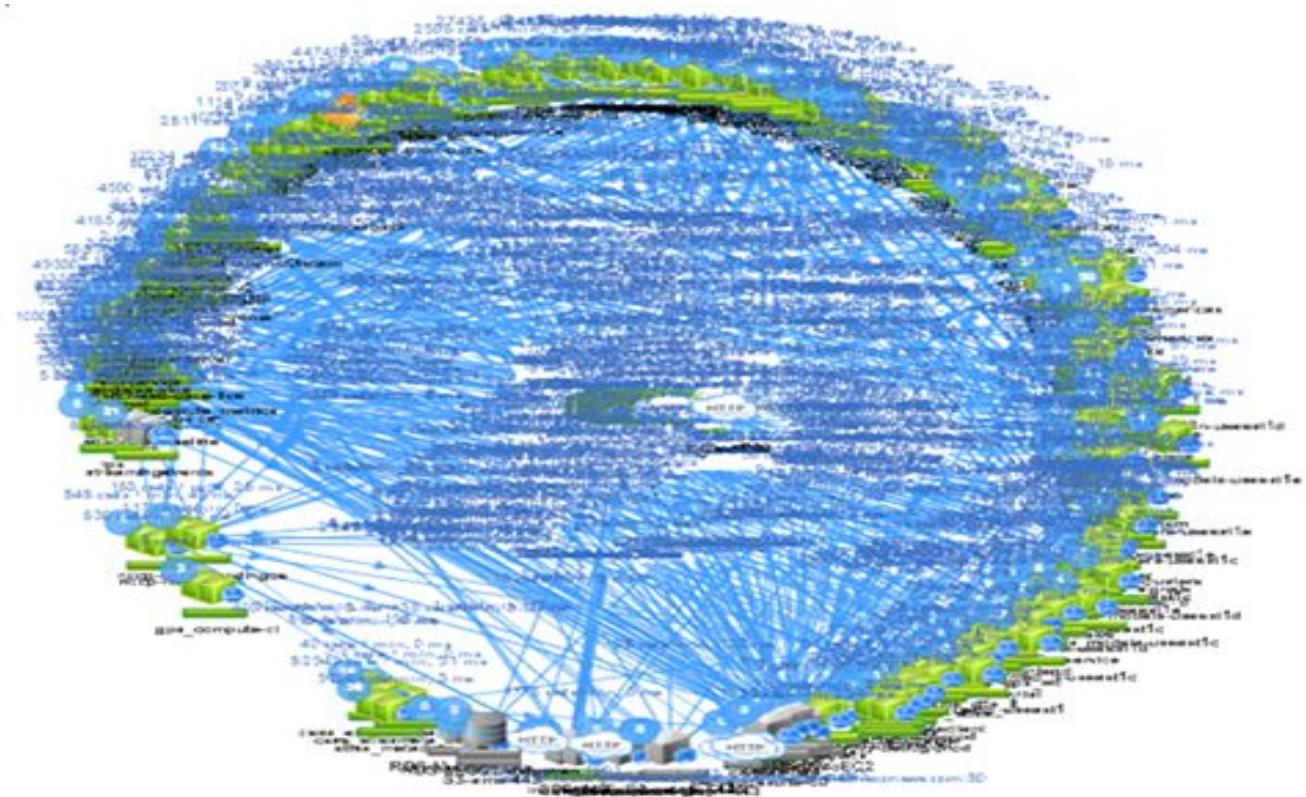
Twitter: <https://twitter.com/adrianco/status/441883572618948608>

Hail-o: <https://sudo.hailoapp.com/services/2015/03/09/journey-into-a-microservice-world-part-3/>

Examples



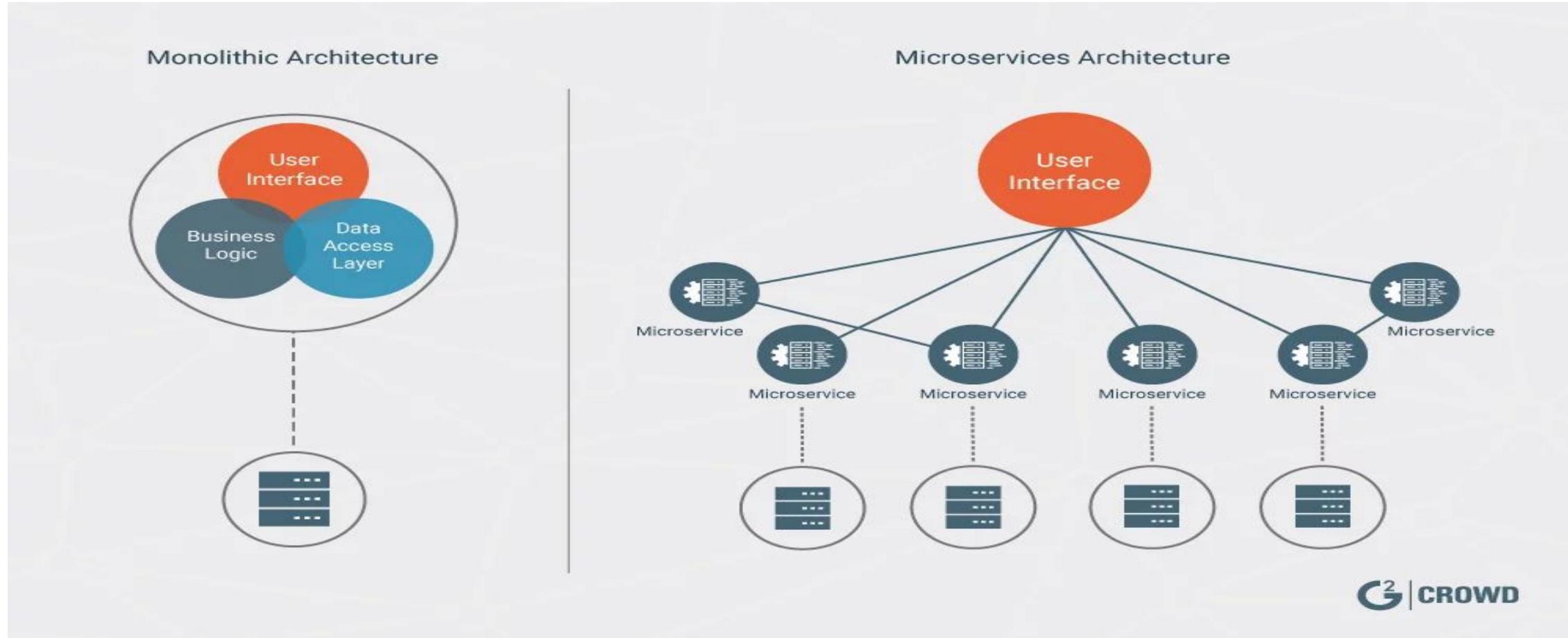
amazon.com



NETFLIX

Microservices vs Monolithic

From: <https://blog.g2crowd.com/blog/trends/digital-platforms/2018-dp/microservices/>



Pros & Cons of Microservices

Pros

- Freedom to use technology
- Responsible of single business capability
- Separate Ownership & Tracking
- Frequent Software Releases
- Parallel releases & feature requests
- No Single Point of Failure
- Code Understanding
- Each service scaled independently
- No Delay for Developers
- Can be reused

Cons

- Complex architecture
- Single functionality becomes distributed so latency
- Difficult to trace a call and which microservice is taking time
- A good amount of integration/e2e tests are required
- Data division for Microservices
- Difficult to maintain transaction safety

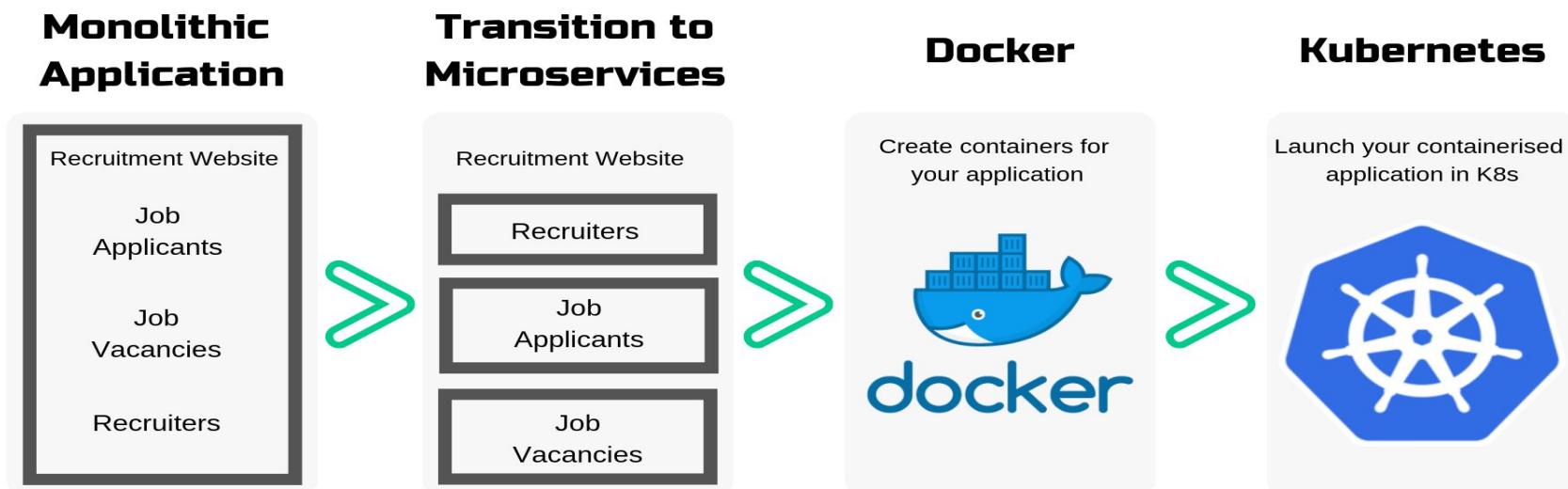
Containers & Kubernetes



DevOps Course By M. Ali Kahoot - Dice Analytics

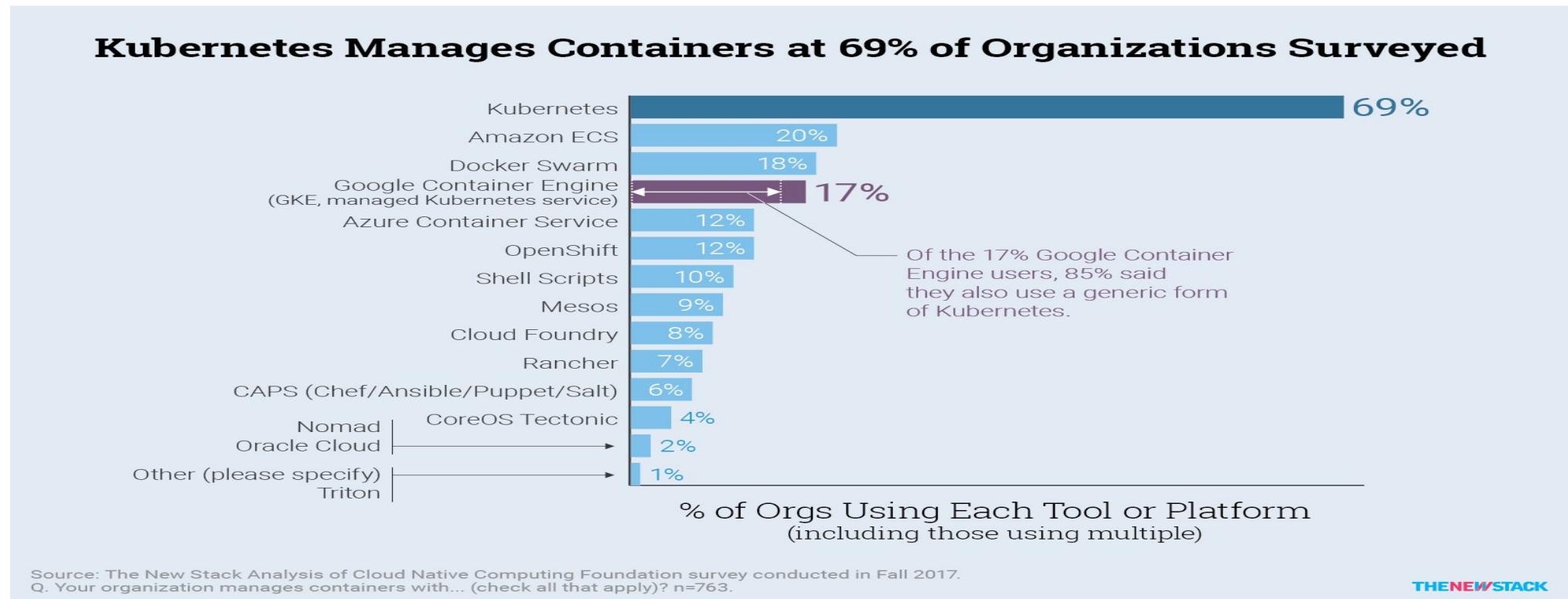
Containers & Kubernetes

From Bytemark: <https://docs.bytemark.co.uk/article/kubernetes-terminology-glossary/>



Containers & Kubernetes

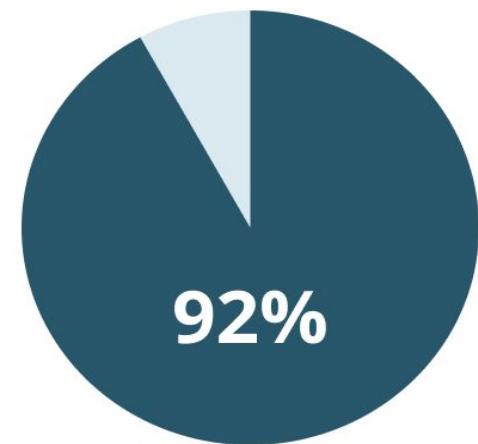
From: <https://thenewstack.io/data-says-kubernetes-deployment-patterns/>



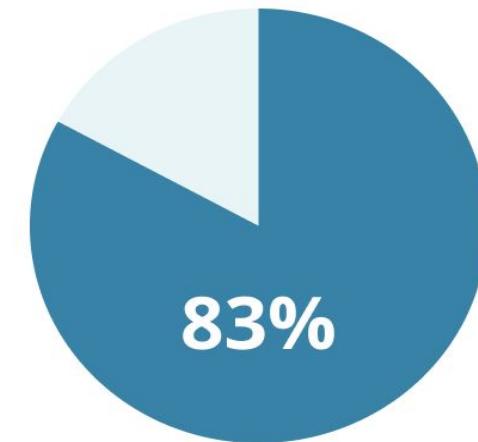
Containers & Kubernetes

From: <https://mobidev.biz/blog/when-why-how-use-kubernetes-app-development>

Kubernetes adoption in 2020



of organizations used
containers in production



of organizations used
Kubernetes for containers orcherstration

Source: The Cloud Native Survey

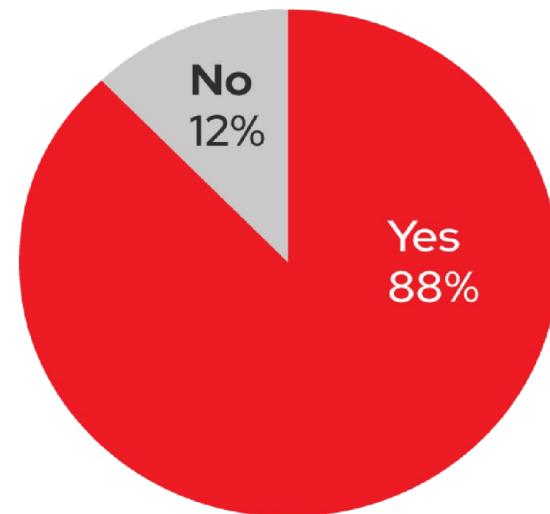


DevOps Course By M. Ali Kahoot - Dice Analytics

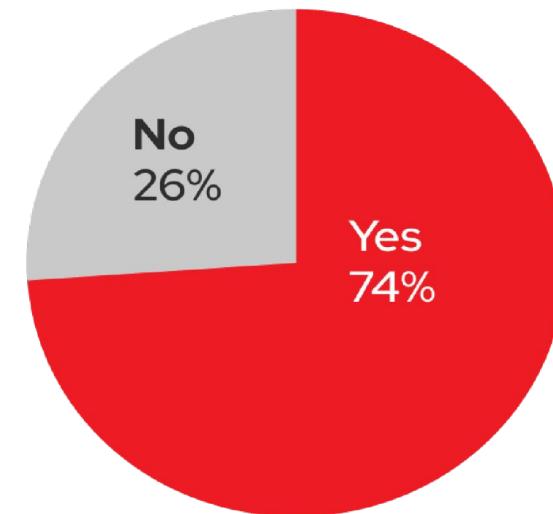
Containers & Kubernetes

From: <https://www.redhat.com/en/resources/kubernetes-adoption-security-market-trends-2021-overview>

Are you using
Kubernetes for container
orchestration?

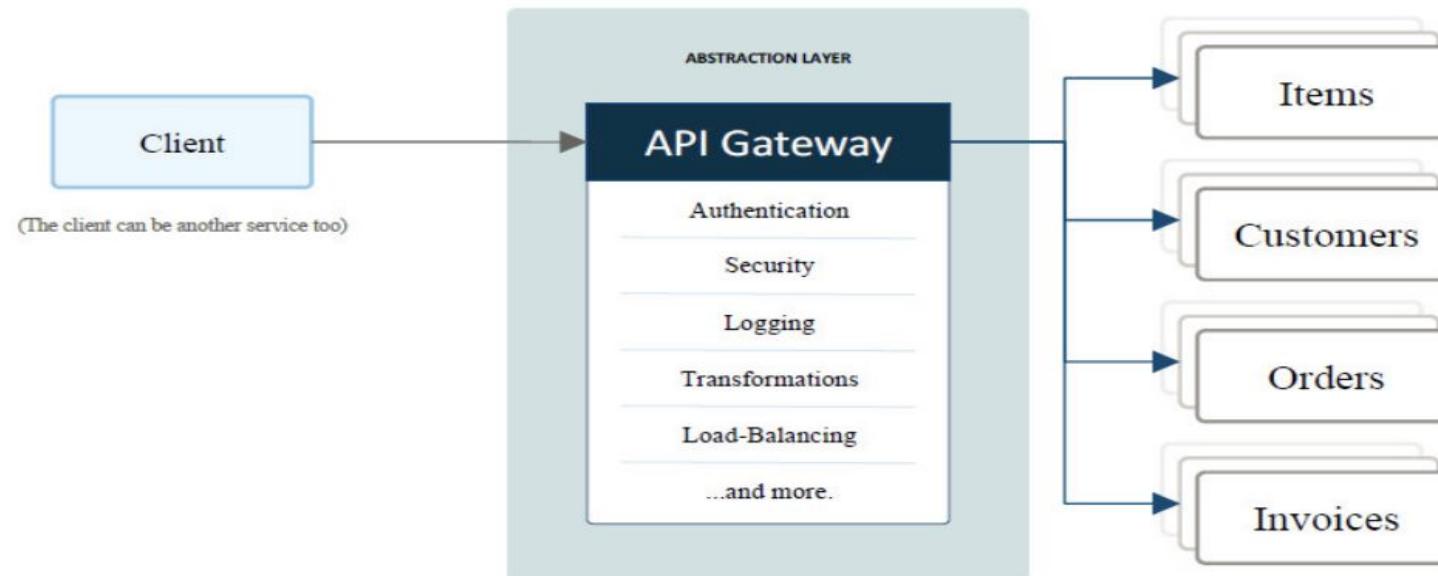


If you're using Kubernetes,
are you running workloads
in production?



API Gateways

API Gateway Pattern & Kong in a Microservices World



Microservices Related Tools

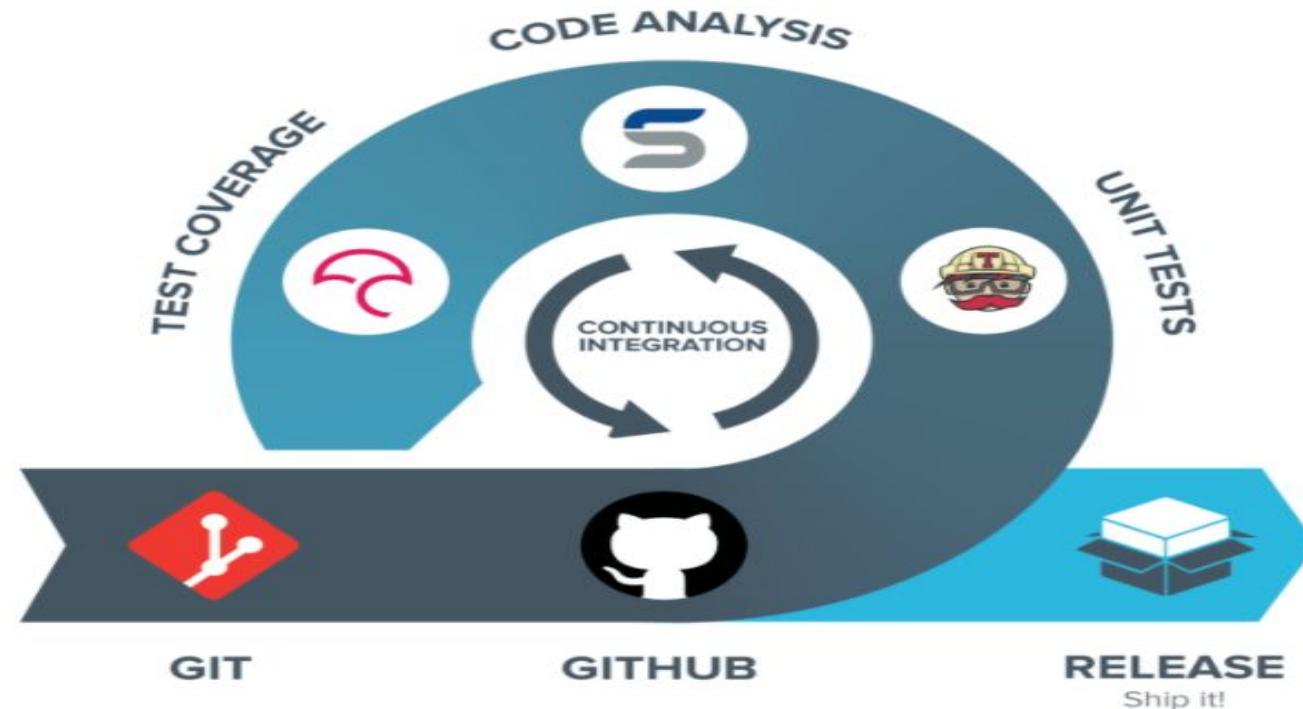
- API Gateways
 - Own, Mulesoft, Kong, 3scale, Cloud Meshery, Tyk, Azure/AWS API Gateway
- Messaging:
 - Kafka, Rabbitmq, Amazon Simple Queue Service
- Kubernetes:
 - Helm, EKS, AKS, IKS, GKE, Openshift
- Docker Containers
 - Azure CI, AWS ECS, Google CE, Docker Compose, Docker Swarm
- Service Mesh
 - Istio, Linkerd
- Serverless
 - AWS Lambda, Azure Functions, Kubeless,

Continuous Integration

- Merge code changes from different developers into a central repository
- Automated builds and tests are run
- Key goals are to fail fast and find and address bugs quicker
- Benefits Developers most, Less merge conflicts

Continuous Integration

From <https://www.silverstripe.org/blog/developers-how-we-use-continuous-integration-at-silverstripe/>



Continuous Delivery

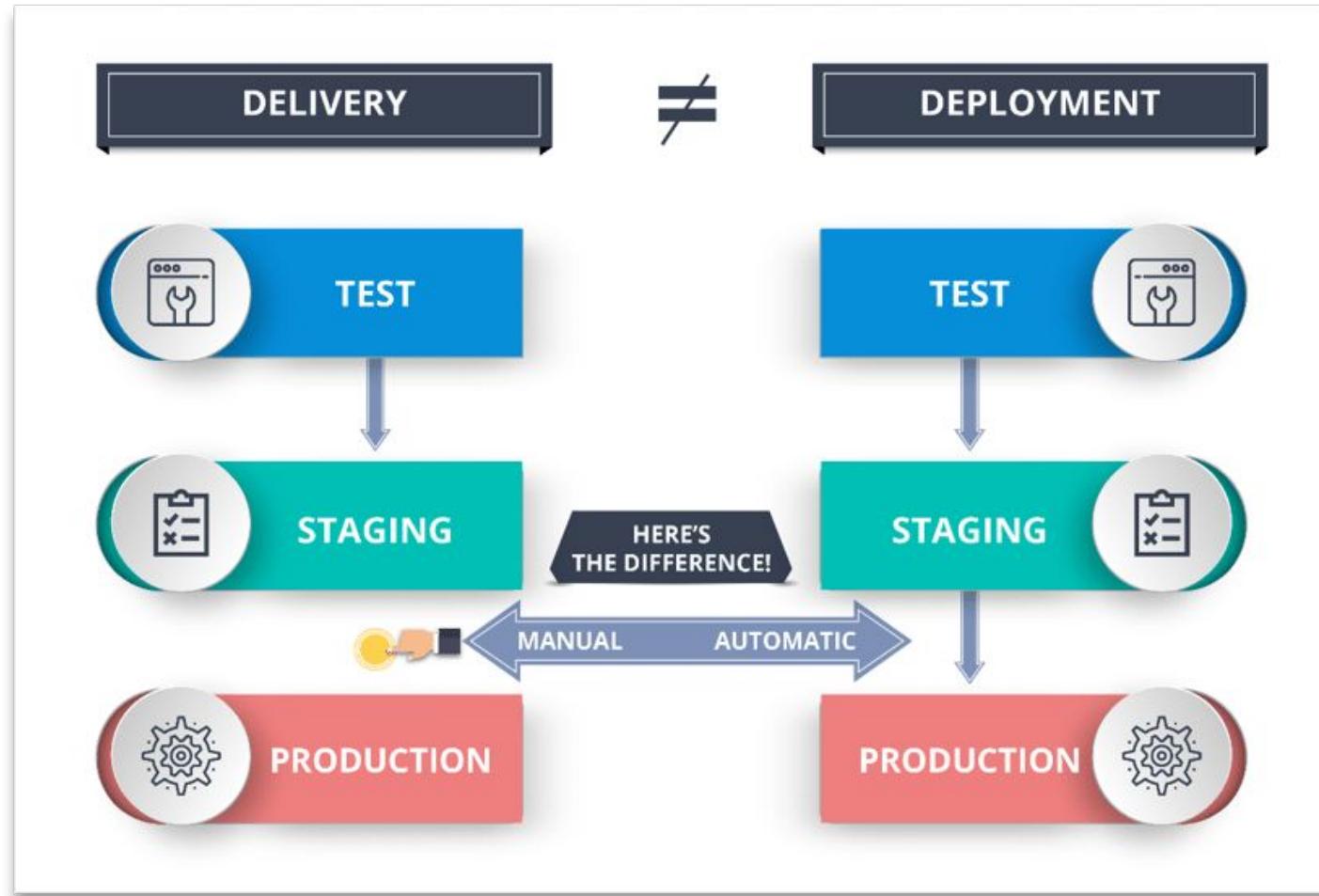
- CI stage is approved
- A small build cycle for short sprints for releasing small features
- Code changes are automatically built & tested
- Can be deployed to a test environment
- Can use branching strategy (other than master)
- Mindset to always have a deployment-ready build artifact.

Continuous Deployment

- CI/CD stages are approved
- The change approved from CI/CD are deployed to production.
- Can use branching strategy(master)
- Release features to get feedback from user

Continuous Delivery != Continuous Deployment

Continuous Integration
+
Automated software release
+
Manual Deployment To Production

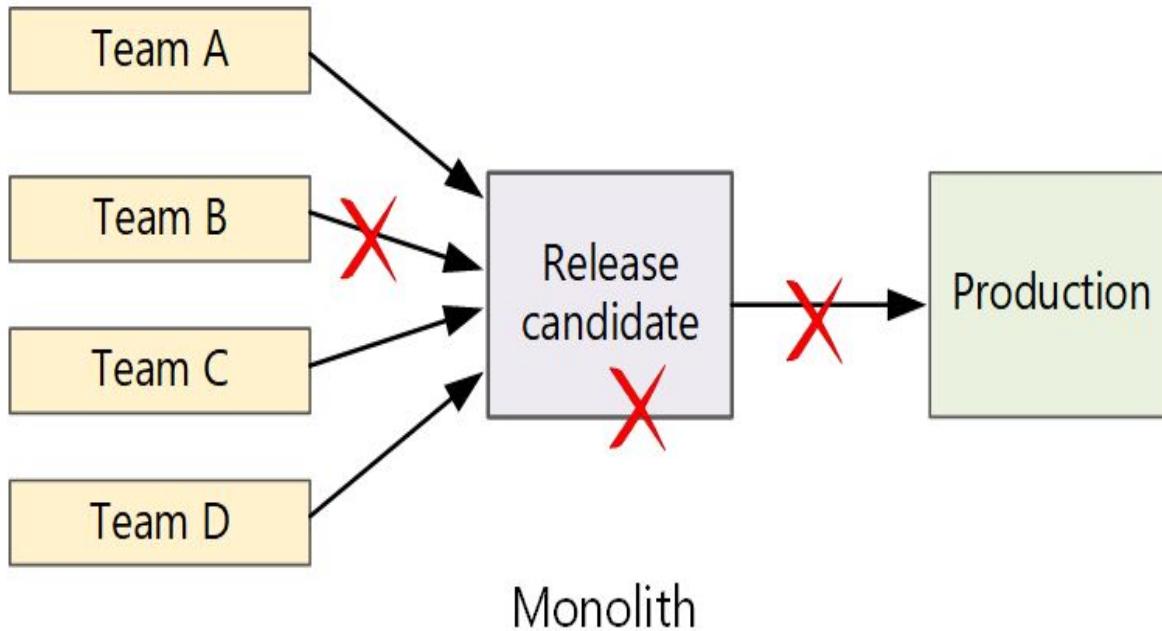


Continuous Integration
+
Continuous Delivery
+
Automated Deployment To Production

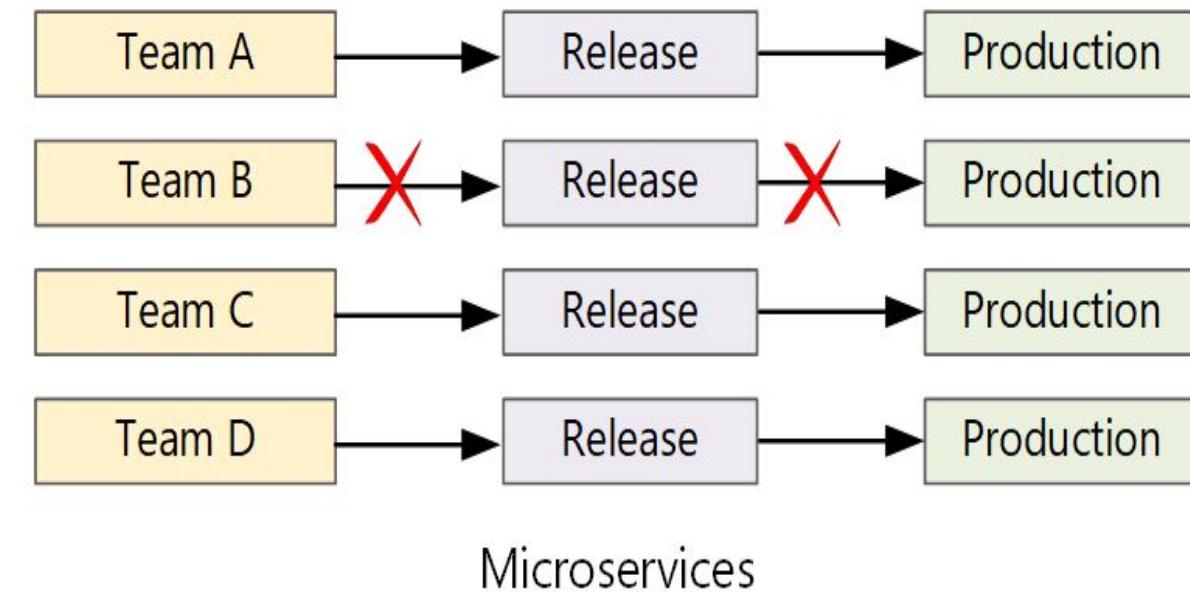
From Edureka: <https://www.edureka.co/blog/continuous-delivery-vs-continuous-deployment/>

DevOps Course By M. Ali Kahoot - Dice Analytics

CI/CD for Monolithic vs Microservice



Monolith



Microservices

From <https://docs.microsoft.com/en-us/azure/architecture/microservices/ci-cd>

CI/CD Tools

- Git / SVN / TFS
- Github / Gitlab / Bitbucket
- Jenkins
- Github Actions
- Gitlab CI
- Tekton CI
- Circle CI
- IBM DevOps
- AWS CodePipeline
- Azure DevOps
- Flux
- ArgoCD

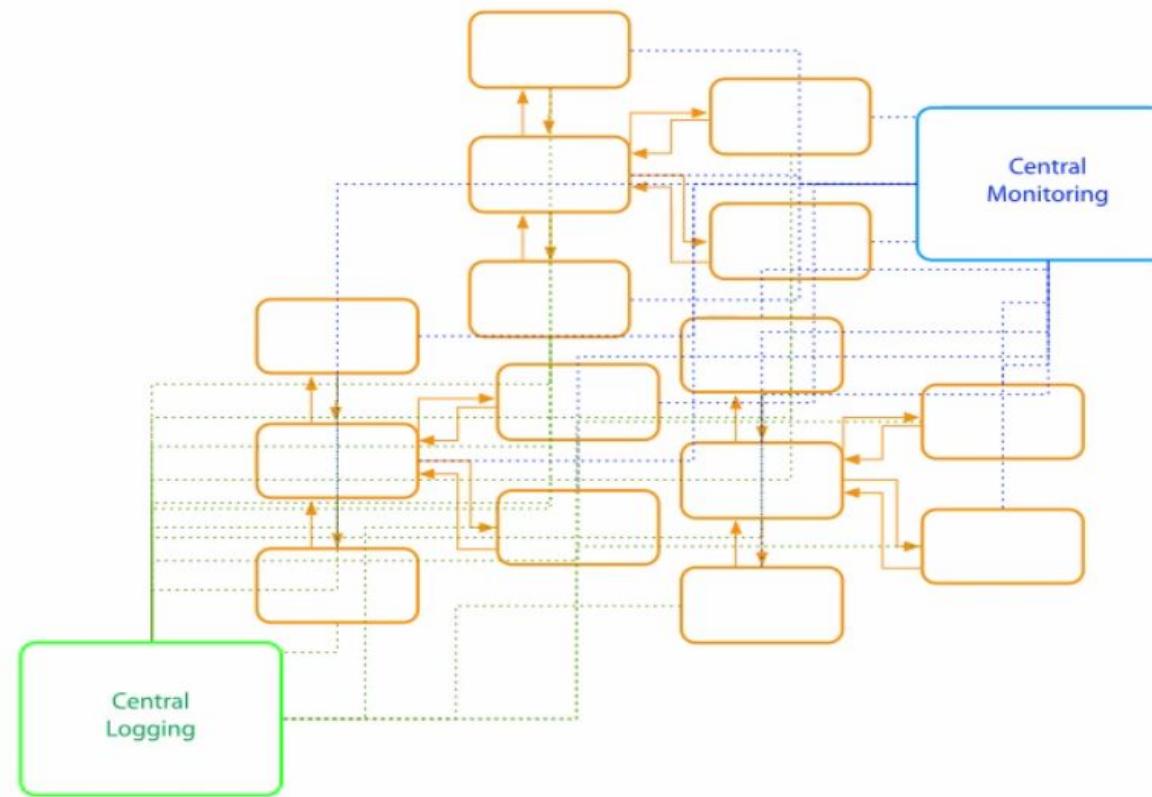
Monitoring, Logging, Tracing

- Deploying Infrastructure is not an issue, maintaining it is
- You will be responsible when
 - Infrastructure is throttled
 - Network went down
 - Machines are not working
- Deploy tools to monitor & see the logs sent by application
- Shed insights into the root causes of problems or unexpected changes.

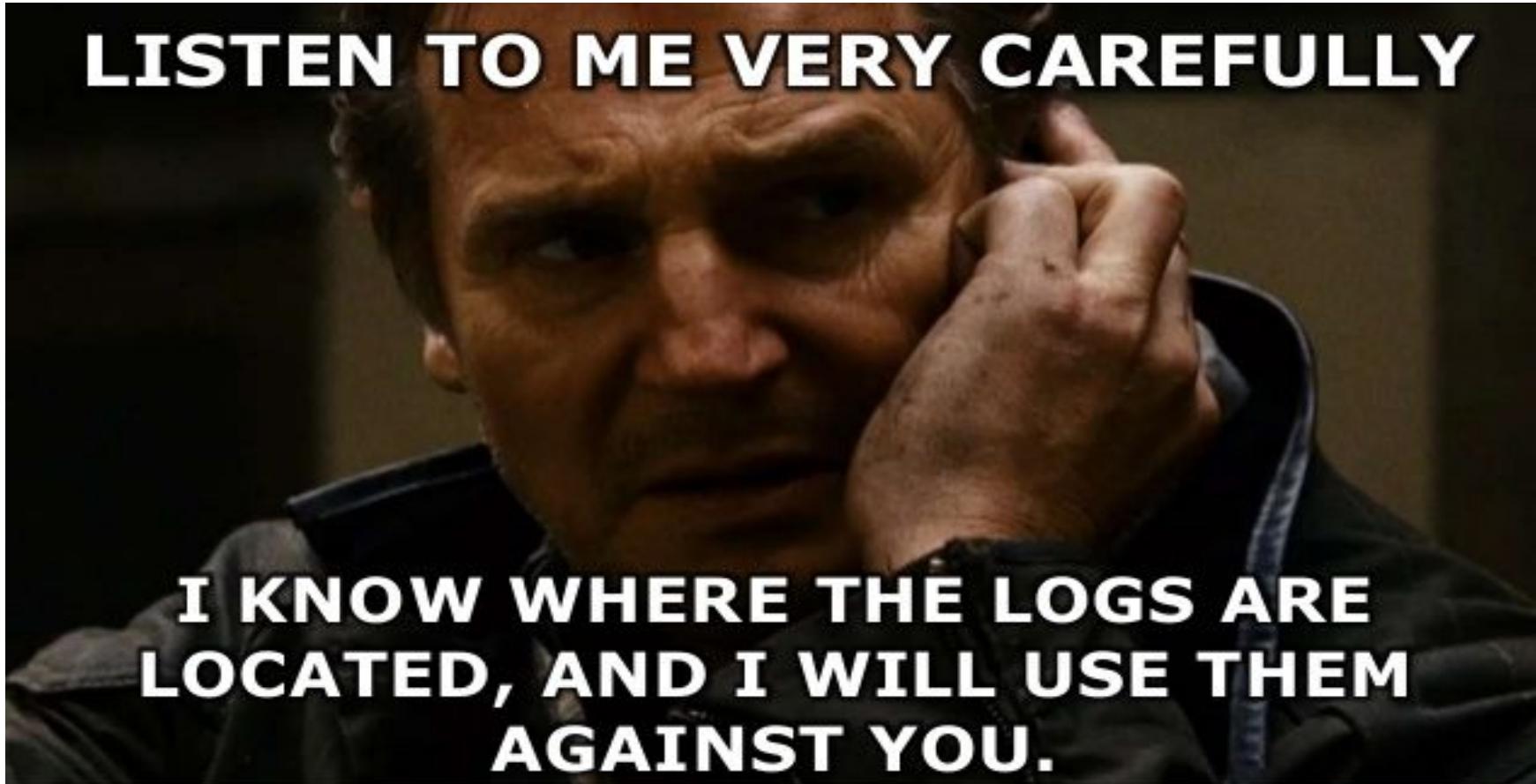
Monitoring, Logging, Tracing, Alerting

- Active monitoring to ensure 24/7 availability
- Alerts or perform real-time analysis of data for making rectifications
- Central place so that team can see logs of their application
- Teams can monitor their application stats
- Tracing means how much time is spent and where is it spent?

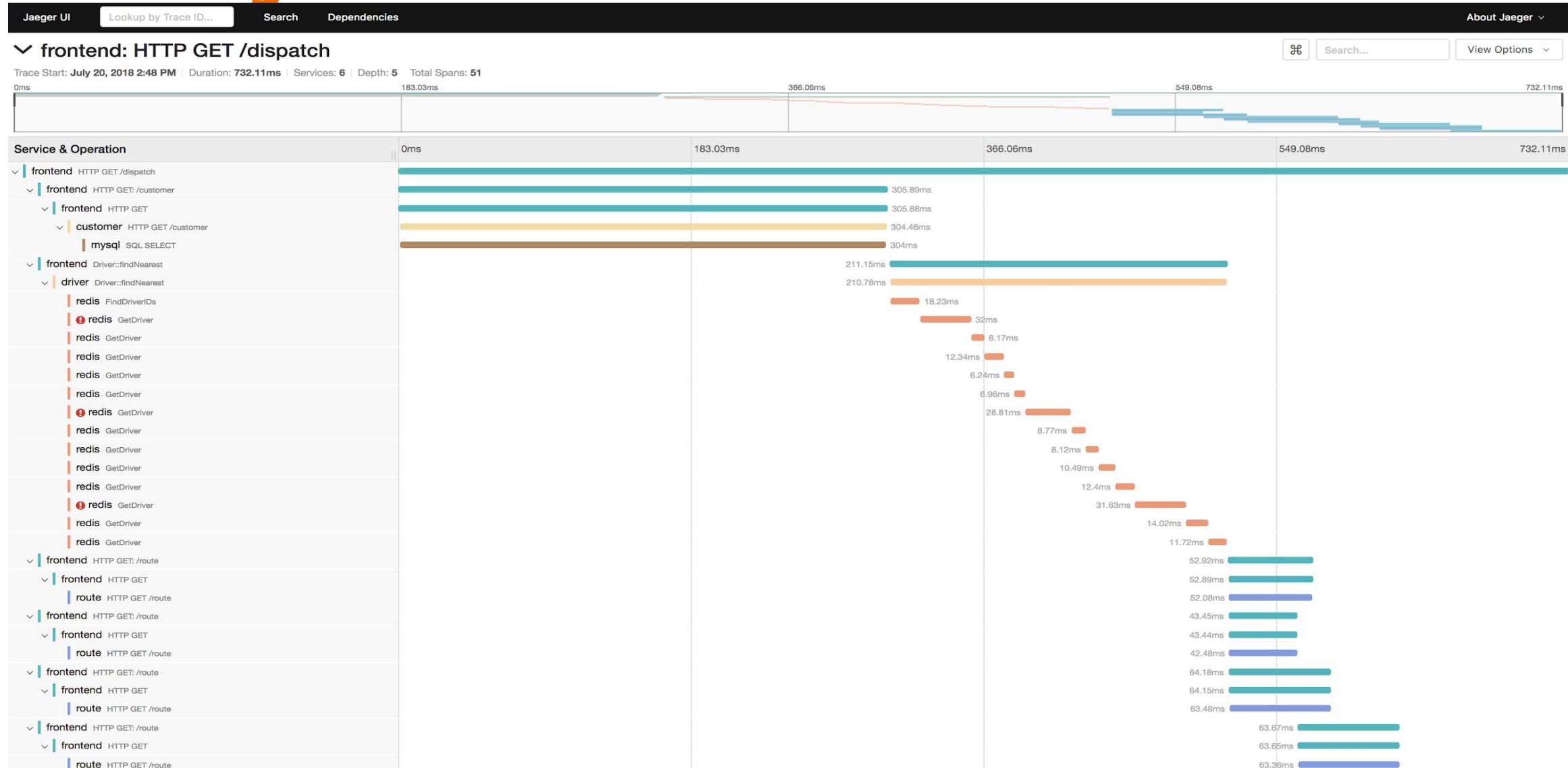
Centralized monitoring & logging



Centralized monitoring & logging



Tracing



DevOps Course By M. Ali Kahoot - Dice Analytics

Tools

- Prometheus
- Grafana
- Nagios
- Datadog
- Jaeger
- Zipkin
- ELK/EFK stack
- AlertManager
- Splunk
- PagerDuty

Communication & Feedback

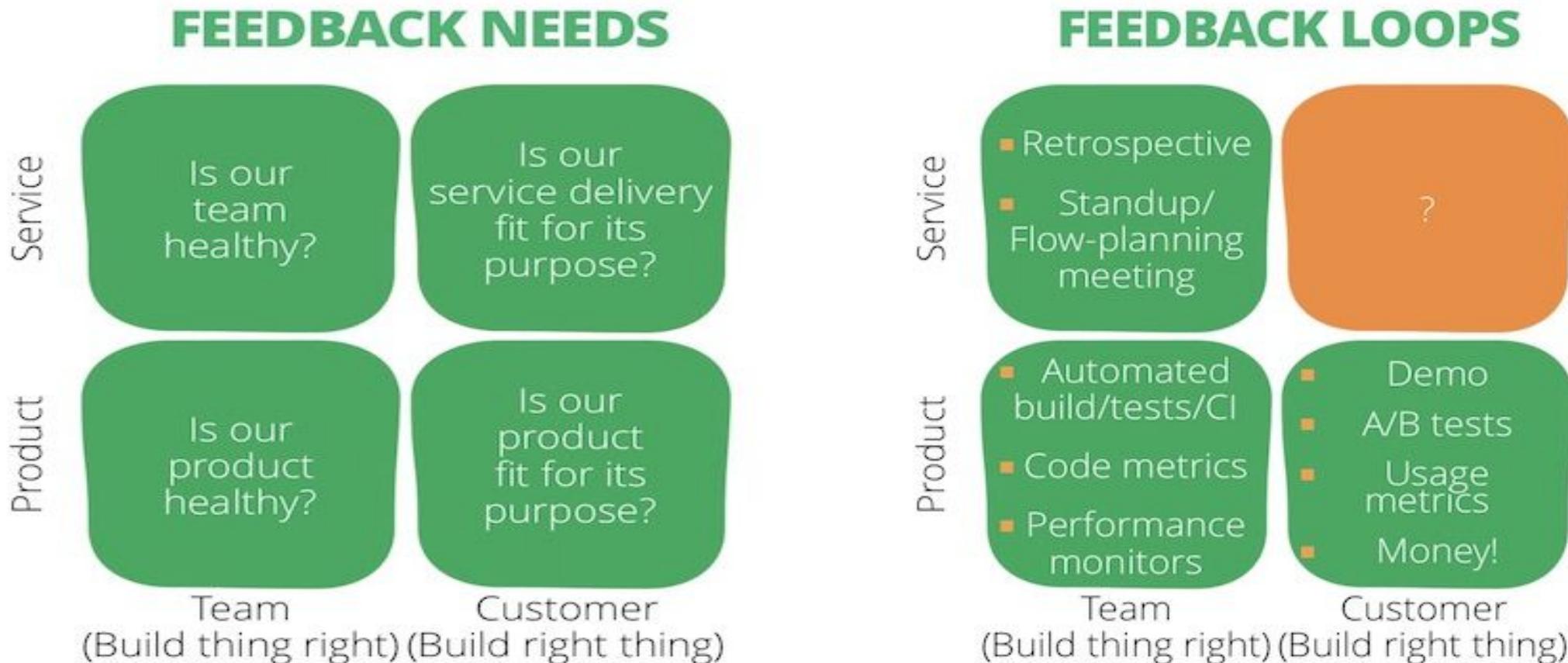
- The most important part of DevOps
- DevOps was introduced due to conflicts between Developers & Operations team blaming each other
- Physically bringing together the workflows and responsibilities of development and operations.
- Maintain visibility across organizations for various events such as deployments, bugs, server downtimes, etc.

Communication & Feedback

- Give continuous feedback to respective stakeholder
- If feature is correct and deployed, take feedback from consumer
- If issue in code, give feedback to developer
- If issue in infrastructure, communicate to teams as soon as possible

Communication & Feedback

- From: <https://www.infoq.com/articles/service-delivery-review-missing-devops-feedback-loop>



Tools

- Slack
- Teams
- Jira
- AlertManager

DevOps Requirements



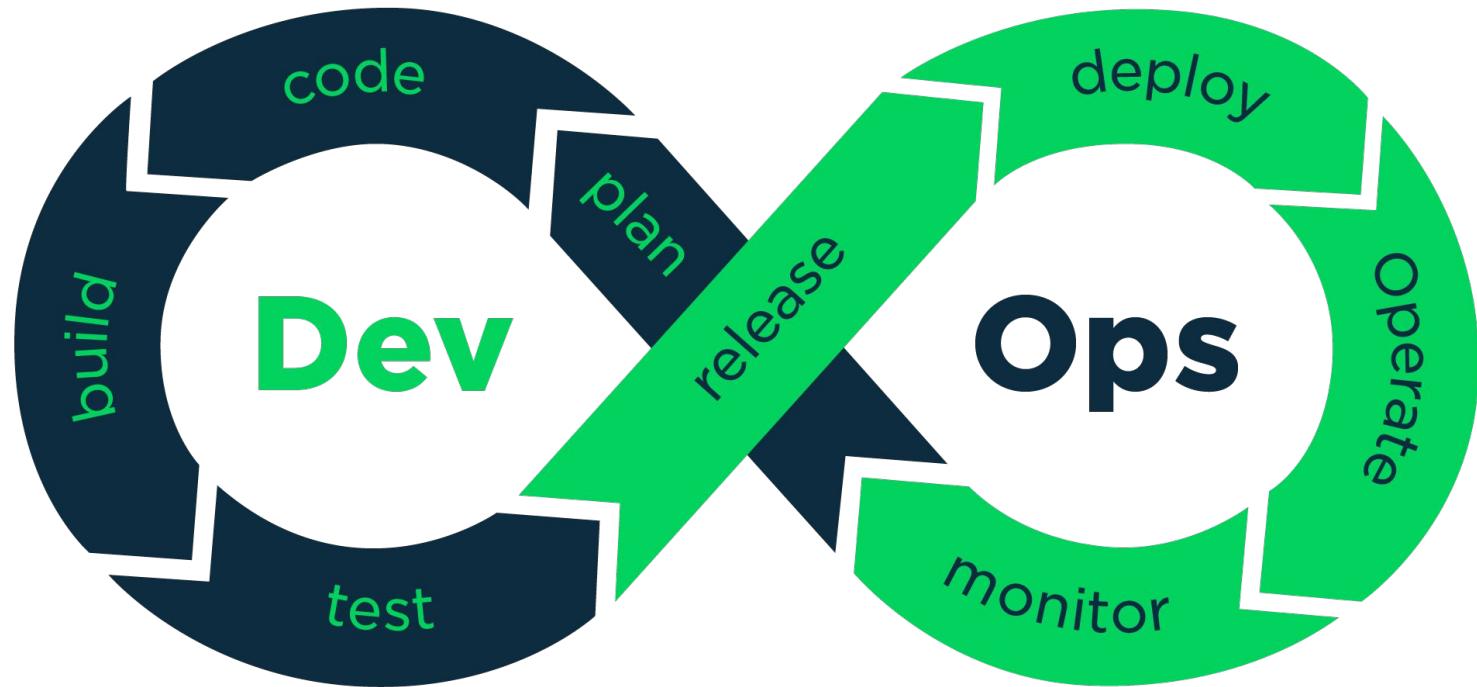
Summary

- IT operations and software development communities realized a dysfunction in the industry.
- Developers and Ops had separate (and often competing) objectives
- The result was siloed teams, long hours, botched releases, and unhappy customers.

Summary

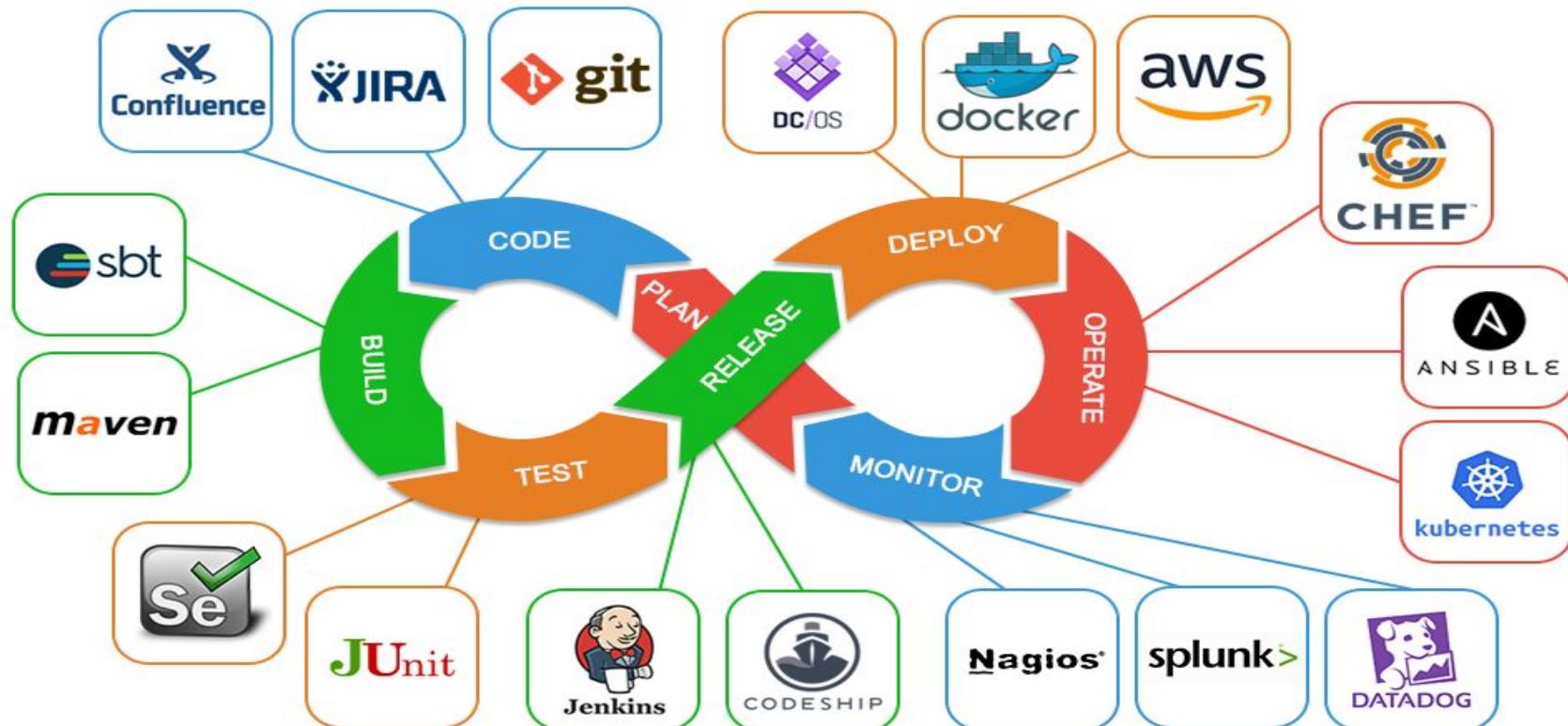
- Combination of cultural philosophies, practices, and tools
- Increase organization's ability to deliver applications/services at high velocity
- Dev and Ops teams are no longer "siloed." Sometimes even merged into a single team
- Quality assurance and security teams may also become more tightly integrated
- Teams use practices and tools to automate processes

Summary



Summary

From: <https://princepatni.com/blog/tech/what-is-devops-devops-engineer-roles-skills-courses-certification/>



DevOps Course By M. Ali Kahoot - Dice Analytics

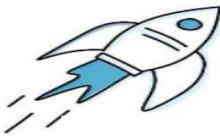
Summary

- **CI/CD:** Release once a week, higher quality of code
- **Microservices:** Loosely coupled components deployed in automated way without waiting on individual component
- **Responsibility:** Shared Responsibility, common goals, process and culture
- **Infrastructure:** Cloud, Containers, Kubernetes, Build once run anywhere.

DevOps Engineer Summary

- Be tool Independant
- Adaptable
- Development Experience
- Able to Learn & Unlearn

DevOps vs SRE

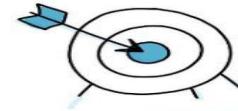


DevOps

it is a paradigm shift that bridges the gap between the siloed **development** and **operations** teams.

What is?

SRE



focuses on the operations of DevOps keeping the software **available** and **resilient** against failure.



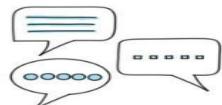
What is the Role of?

DevOps

increase software **delivery velocity**, improve service reliability, and build **shared ownership** among stakeholders

SRE

keep focus on what **matters to customers**: the platforms and services must be **available**



Metrics for Success

DevOps

- Lead time for changes
- Deployment frequency
- Time to restore service
- Change failure rate

SRE

- Service Level Objectives
 - Error Budget
 - Latency & Traffic
 - Errors & Saturation

Things to do before next class

- Setup a Ubuntu VM or environment
 - [https://github.com/msannan/DiceLab-Installations/blob/main/ubuntu-vm-setup.
md](https://github.com/msannan/DiceLab-Installations/blob/main/ubuntu-vm-setup.md)
 - [https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-u
sing-virtualbox#1-overview](https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-using-virtualbox#1-overview)
- Install Docker, you can search how to install Docker on Ubuntu, its really simple, must be able to run the command “docker info”
- Have git installed on your system