Microservices

Authored by Chris Bohnet

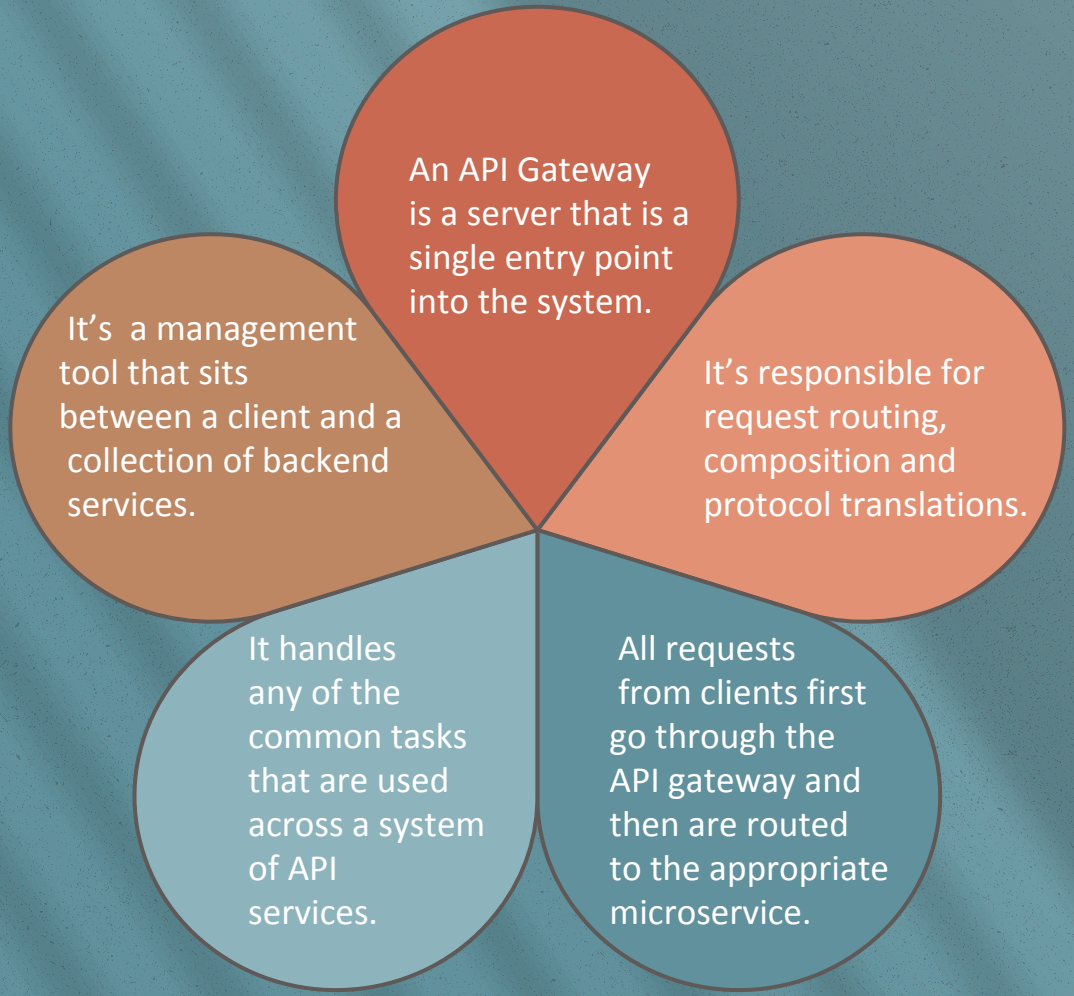
Agenda

- What are microservices?
- What is an API gateway?
- Advantages of microservices
- Disadvantages of microservices
- How microservices are deployed and managed in production
- How microservices are scaled

What are microservices?

- A suite of small applications that each performs a single function.
- Combination of services structured to be highly maintainable and scalable.
- Designed for rapid delivery of complex applications broken down in smaller services.
- Able to be developed, tested and deployed independent of other services & by multiple development groups.
- Services communicate with each other through simple, universally accessible application programming interfaces, APIs.
- Individual services that are built around business capabilities.

What is an API Gateway?

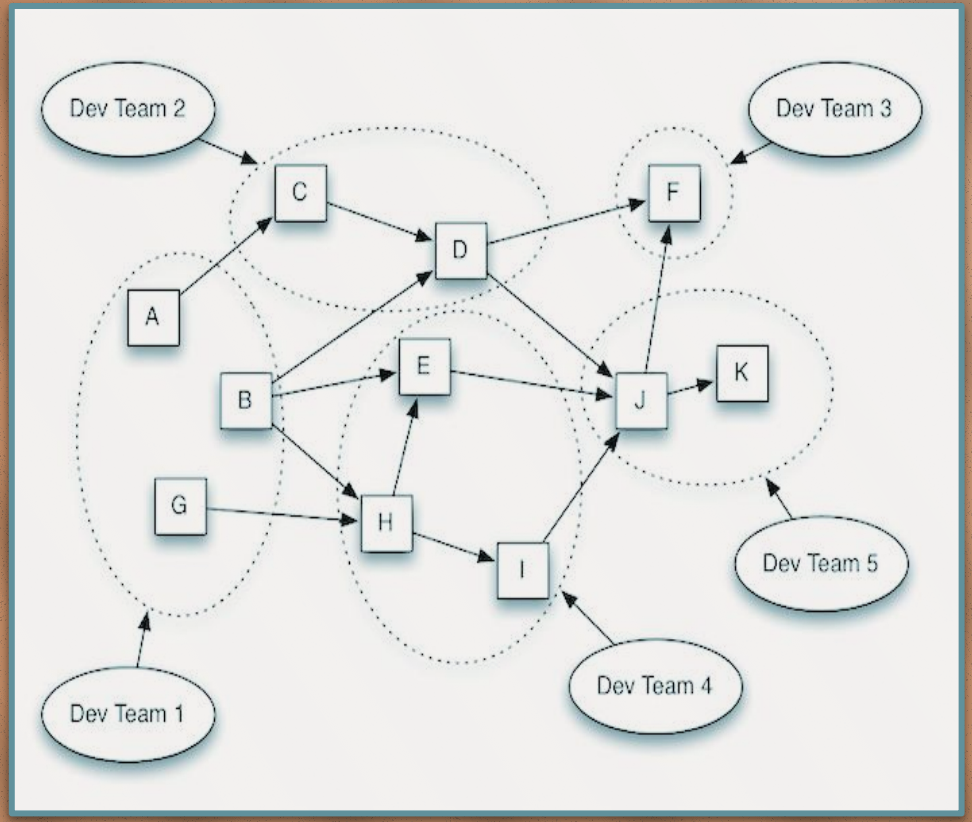


Advantages

Efficient, productive development with each microservice having a clear team owner.

Collisions avoided with each team having a clear, non-overlapping responsibility.

Applications can grow as the company & business requirements grow.



Continuous delivery possible by independently deploying services also aiding to rapid delivery.

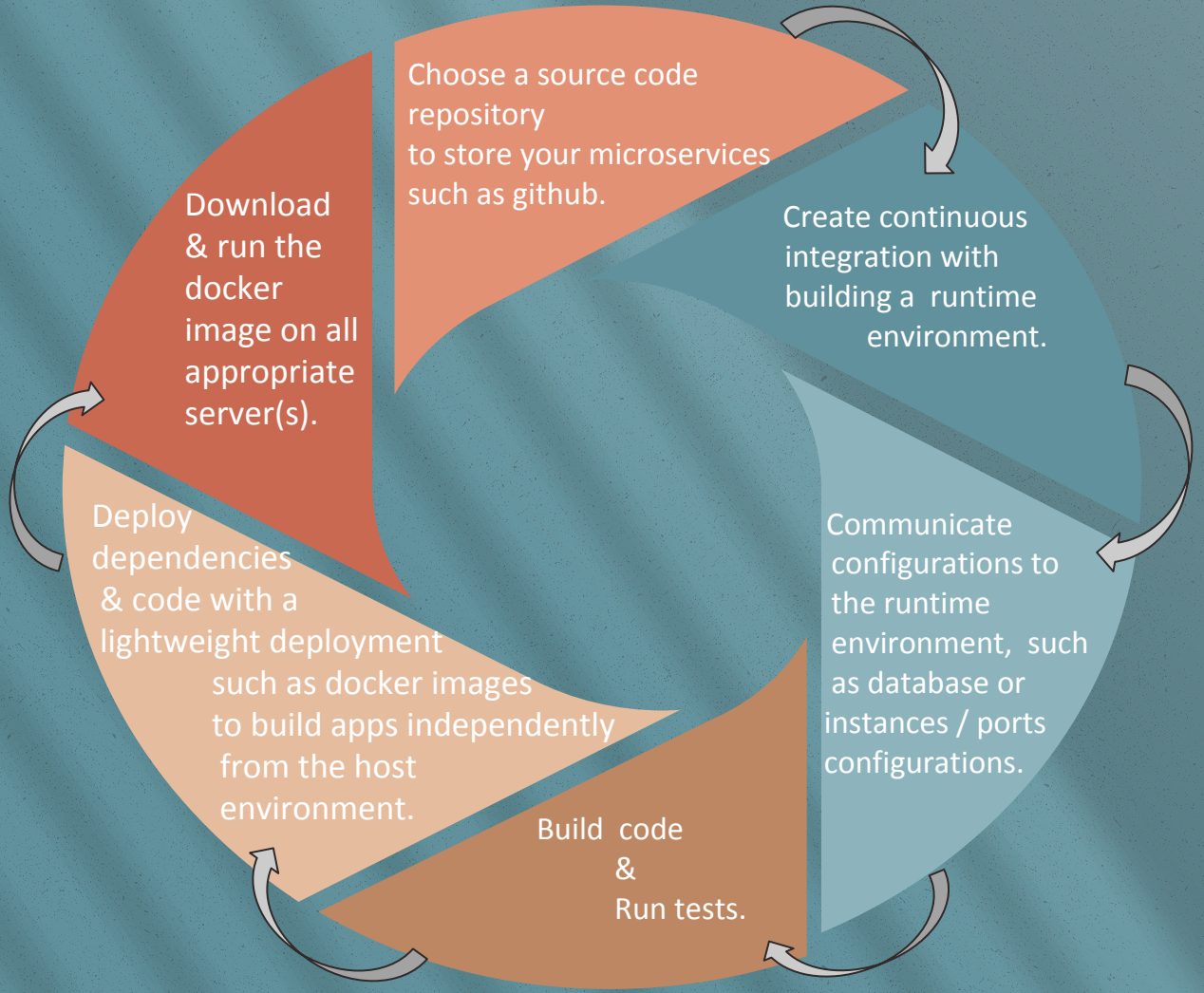
Minimal code of usually no more than 2000 lines performing a single function making it easy to develop, test, deploy, scale and maintain.

Services are loosely coupled driving ease of development, scalability & maintenance.

Disadvantages

- Challenging to maintain multiple languages and frameworks & existing tools are likely incompatible with new services.
- Each service has its own database & transaction management system making data consistency challenging.
- Each service requires individualized monitoring and testing so additional cost of automation tools and skills needed.
- The number of processing can grow quite large when messaging services and load balancing are factored in.
- Each service has its own database & transaction management system making data consistency challenging.
- Security challenges due to the volume of services and data exchanged among them being exposed on the network.

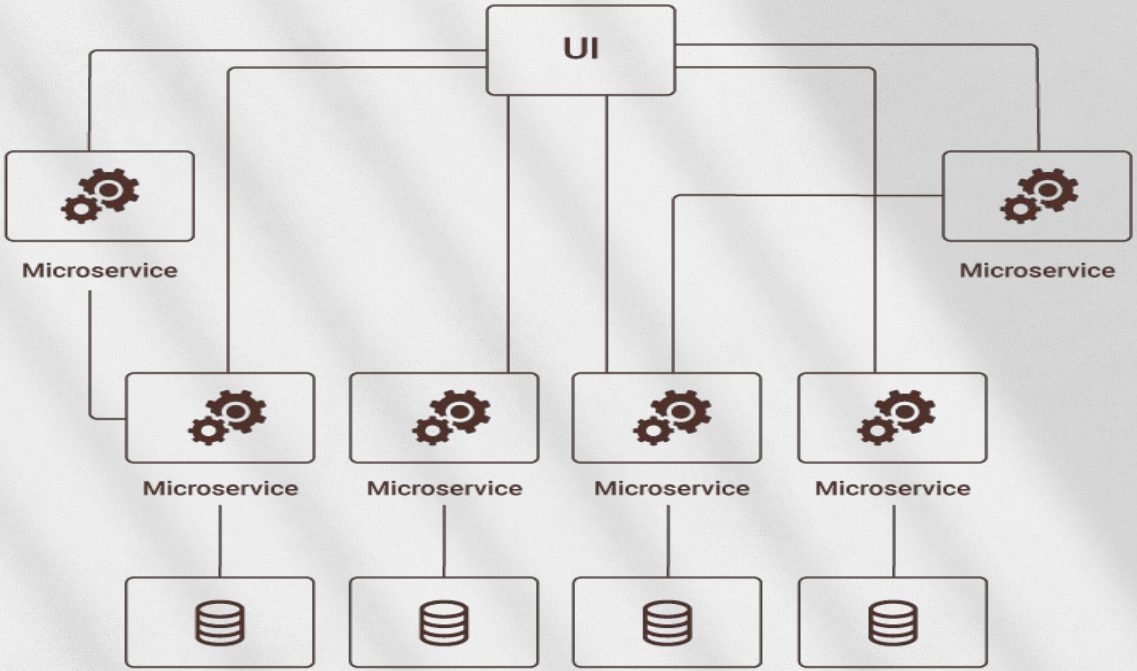
How microservices are deployed & managed in production



How are microservices scaled?

- Scalability is determined by how efficiently tasks are divided and broken down.
  - Each microservice needs to both scale individually and as part of a larger system.
  - Dependencies of the microservice should scale with it.
- Performance measures how efficiently the system is able to perform these tasks.
  - Concurrency is the process by which each individual task is broken down into smaller pieces or services.
  - Partitioning determines the efficiency by which these services run in parallel to each other.

Microservice Architecture Diagram



Resources

Newrelic, “Microservices: What they are and how to use them”, as referenced at

https://blog.newrelic.com/technology/microservices-what-they-are-why-to-use-them/

https://microservices.io/

Tiempo development, “Top Microservices Disadvantages”, as referenced at
https://www.tiempodev.com/blog/disadvantages-of-a-microservices-architecture/

Programmable Web, “Understanding the Role of APIs in microservices architecture”, as referenced at
https://www.programmableweb.com/news/understanding-role-apis-microservice-architectures/analysis/2016/05/05

Datawire, “Nine questions to ask when continuously deploying microservices”, as reference at

https://www.datawire.io/guide/deployment/nine-questions-ask-continuously-deploying-microservices/

DZone, “Scaling Microservices: The Challenges and Solutions”, as referenced at

https://dzone.com/articles/scaling-microservices-the-challenges-and-solutions