

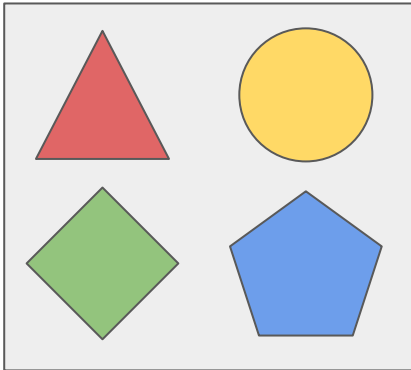
Microservices & Load Balancing

Anders Eriksson & Haris Adzemovic

Monolith (classic)

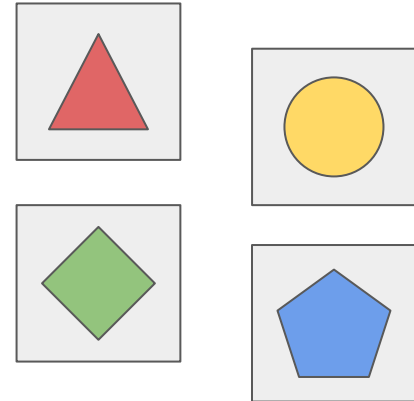
- Client facing
- Database
- Server side

← **MONOLITH**



Microservices

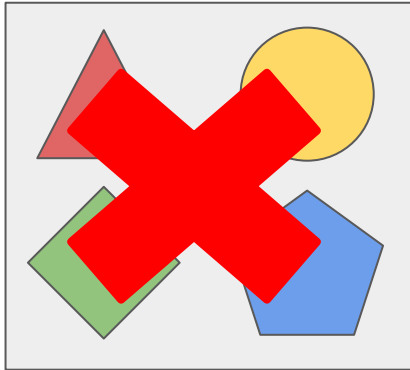
- Divide parts into independent services
- Services communicate over web



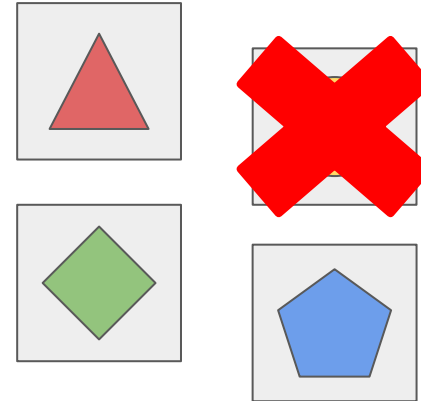
Some differences

Something goes wrong

Monolith



Microservices



Some differences

Technologies freedom

Monolith



Microservices

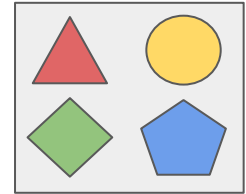
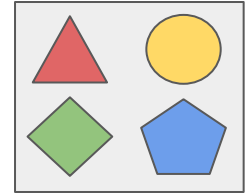
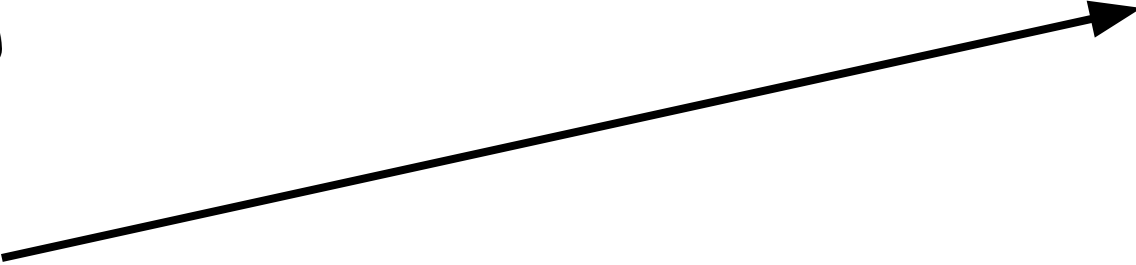
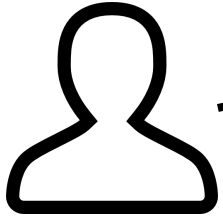
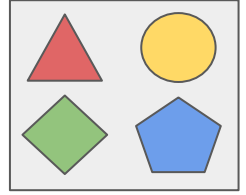
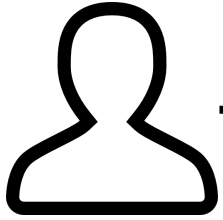


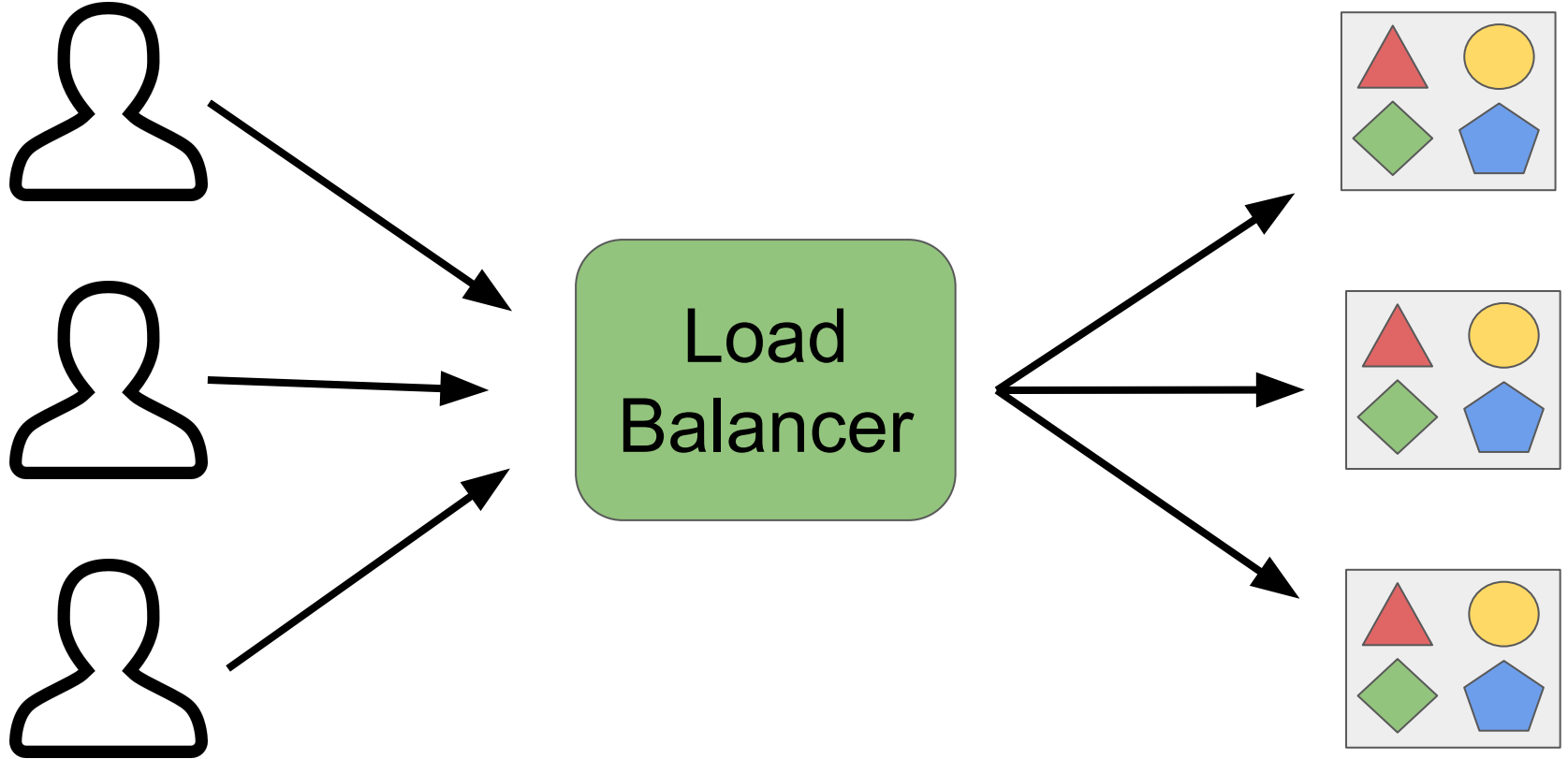
Other differences

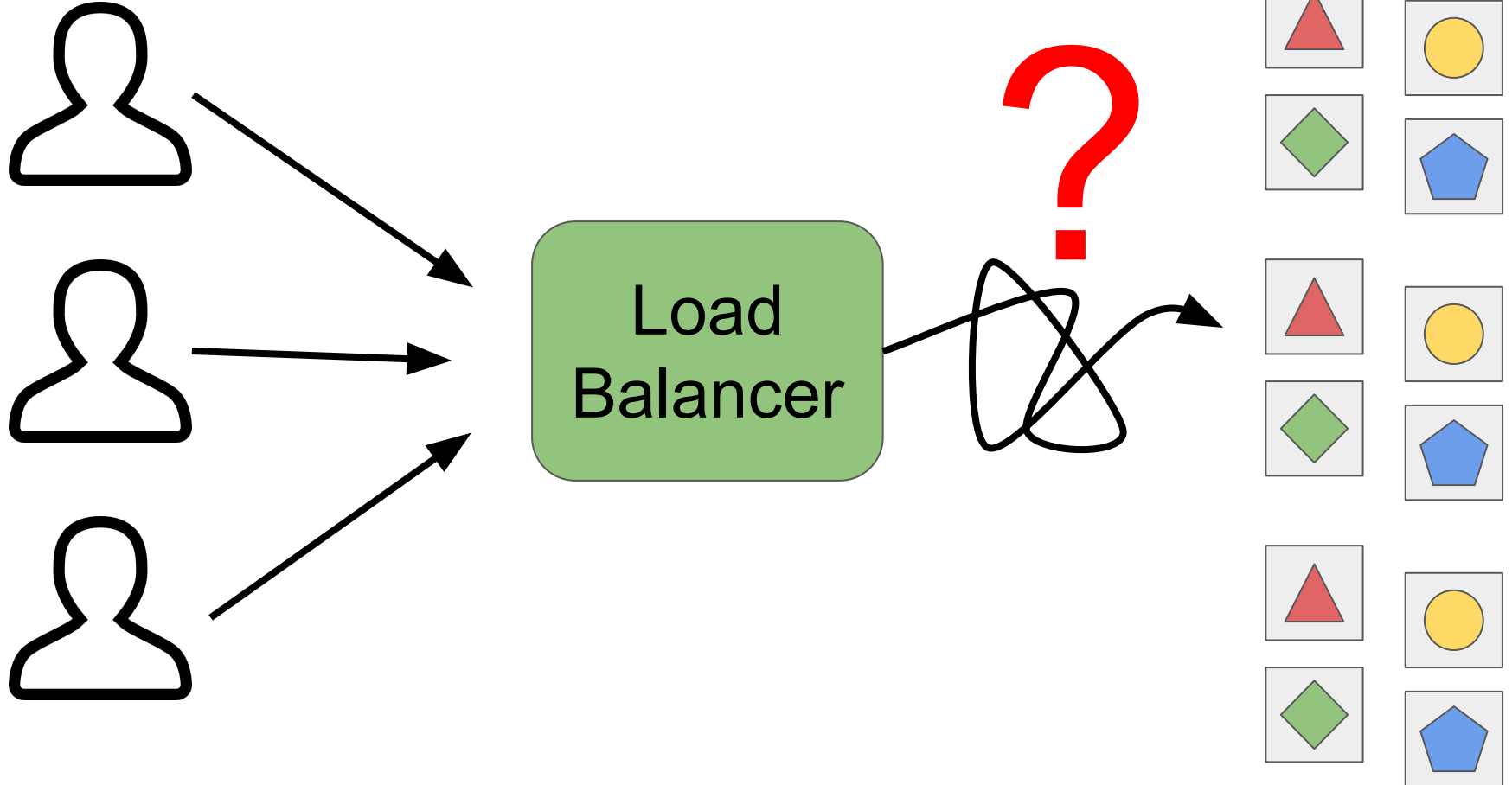
- Communication
- Scaling
- Monitoring
- Speed
- Difficulty of implementation
- ...
- **Load balancing**

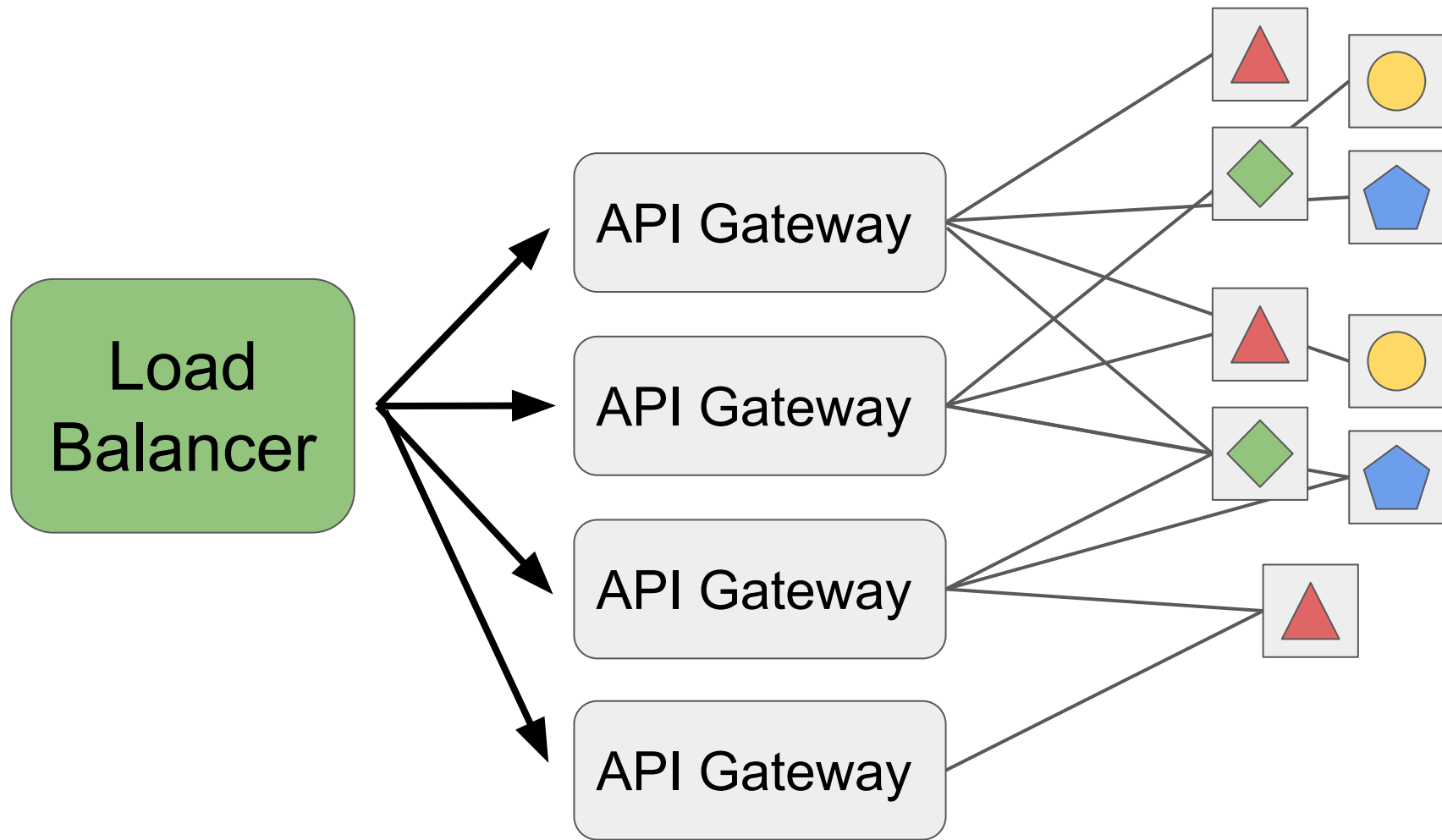
Load Balancing

- The distribution of workload across multiple resources
- Nginx
- Modern cloud computing platforms (AWS, MS Azure, ...)





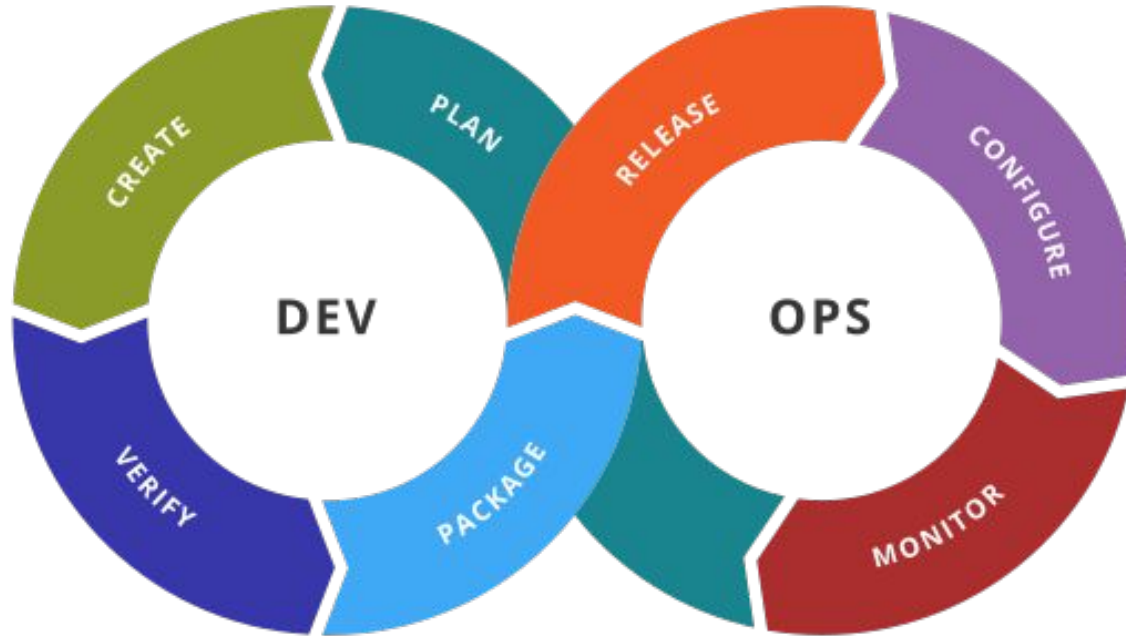




DevOps



Microservices



How to go from monolith to microservice

- You want to separate services
- Communication between these parts is hard
- “The biggest issue in changing a monolith into microservices lies in changing the communication pattern.”
 - - <https://martinfowler.com/articles/microservices.html>
- “Monolith first” or not?

MonolithFirst



Martin Fowler

3 June 2015

Don't start with a monolith

... when your goal is a microservices architecture



9 June 2015

Stefan Tilkov

