

Container Bootcamp

Microservice Size



Absolute value?

- Lines of Code?
- Depends on programming languages
- ...and the problem to solve
- No absolute value

**Microservices Should be
Small, right?**

Modules

- **Microservices = modules**
- **Developer only needs to understand module**
- **Smaller modules easier to understand**

Replaceability

- **Many projects try to replace old systems**
- **Microservices should be replaceable**
- **Smaller modules easier to replace**
- **Large modules (e.g. deployment monoliths) impossible to replace**

Microservices Should be Small!

- **Goal: team should be able to work independently**
- **One team responsible for each Microservice**
- **Microservice must be small enough for one team**
- **Team might have more than one**

**Microservices should be
Big, right?**

Infrastructure

- Demand on infrastructure
 - Build and Deploy
 - Continuous Delivery pipeline
 - Version control
-
- Bigger services → fewer services → less infrastructure

Distributed Communication

- **Distributed communication slow**
 - **...and unreliable**
 - **Refactoring across services hard**
-
- **Bigger services → fewer services**

Transactions and Consistency

- **Transactions limit to one Microservice**
 - **Technical constraints**
 - **Locking**
-
- **Consistency: e.g. value of orders and revenue must be equals**
 - **Only possible in one Microservice**

Microservice Size

