

Table of Contents

- Benefits** 1
- Things to resolve** 1
- Deployment** 1
- Cluster** 1
- Microservices Map** 2

Service Decomposition		
→	→	→
Monolithic	SOA	Microservices
No granularity o	Some granularity oo	Fine granularity ooooo

Benefits

- The services are easy to replace
- Not dependent on programming language
- Best / finest granularity
- Each service is elastic, resilient, composable, minimal, and complete
- Continuous delivery software
- Do one thing well (The services are small - fine-grained to perform a single function)
- Decentralized
- Independent
- Low / stable latency

The functionality of the service may not be so simple that the output of the service always needs to be combined with another service's functionality.

Things to resolve

- Auto service discovery
- Monitoring
- Loadbalancing of course

Deployment

- Containers (LXC, OpenVZ or Docker)
- Image is the version
- Test and deploy same artifact (image)
- Stateless servers decrease change risk

Cluster

- Geo Located
- Resource Pool
- Grouping Related Instances with Containers
- Start empty, dynamically scalable

Microservices Map

From:
<https://wiki.janforman.com/> - **wiki.janforman.com**

Permanent link:
<https://wiki.janforman.com/microservices>

Last update: **2019/06/27 11:39**

