



University of Stuttgart
Germany

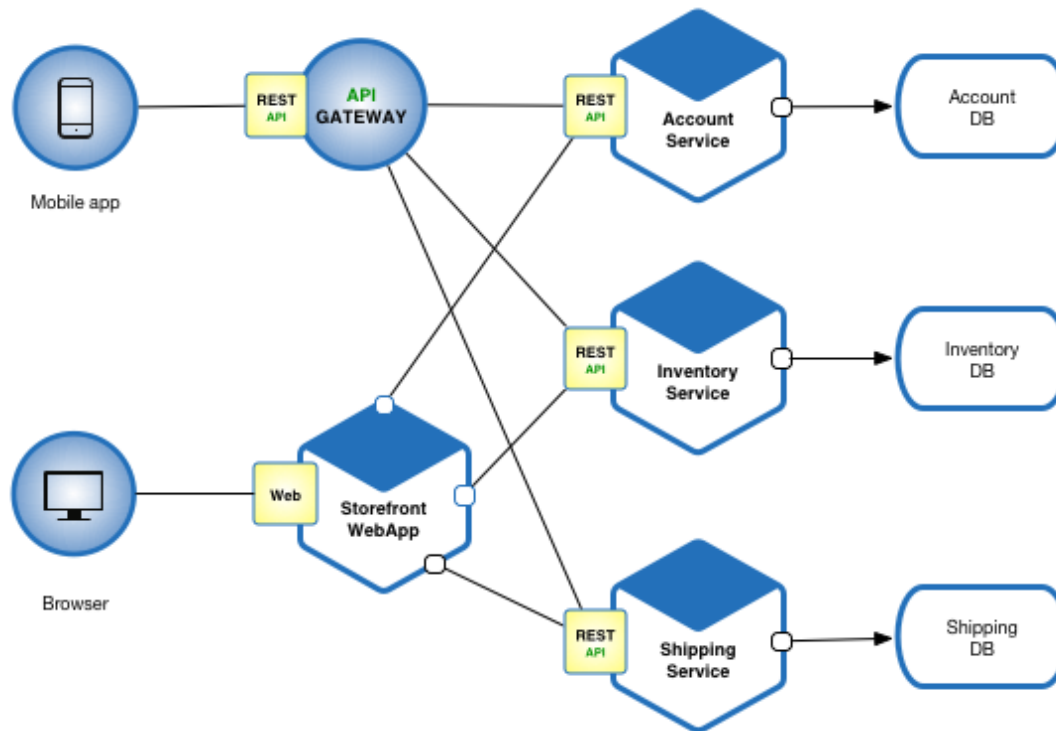
Simulation-based resilience prediction of microservice architectures

Proposal Fachstudie

6/1/2017

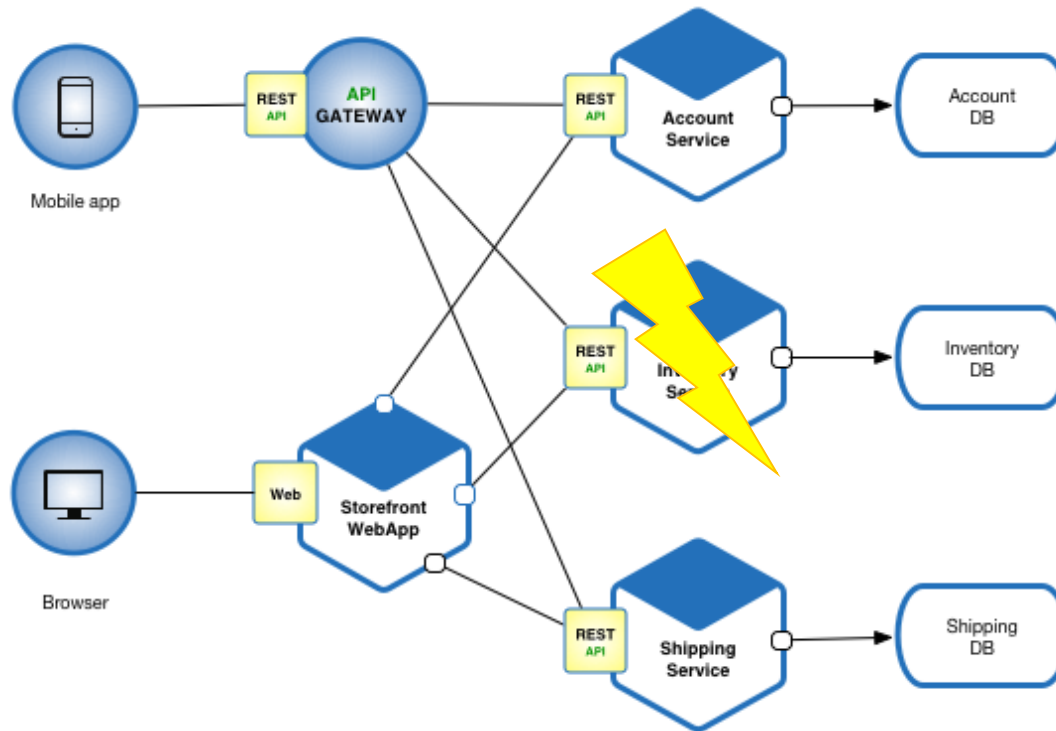
**Samuel Beck
Johannes Günthör
Christoph Zorn**

Microservice Architecture



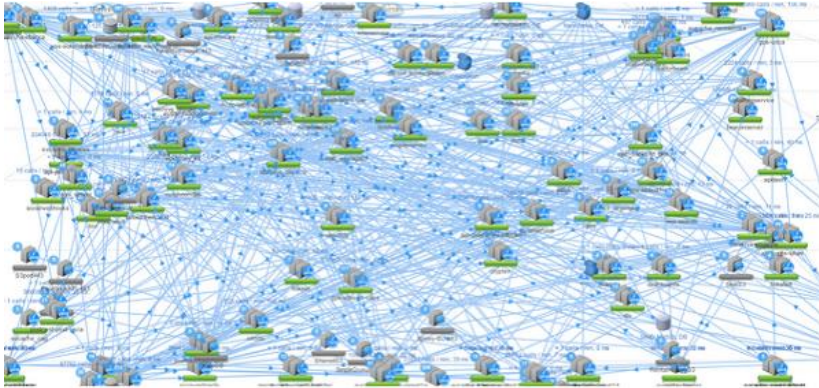
http://microservices.io/i/Microservice_Architecture.png

Failures in Microservice Architectures

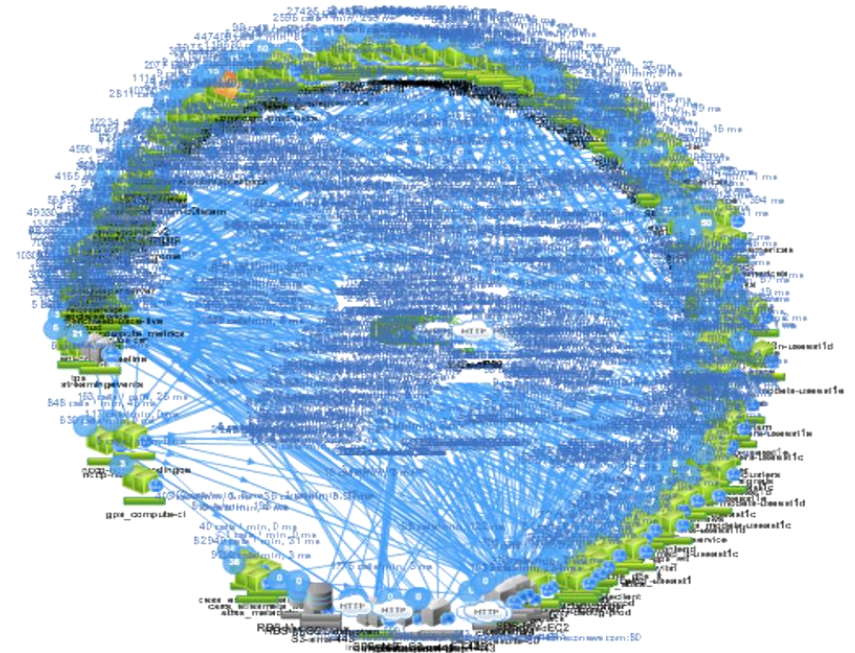


http://microservices.io/i/Microservice_Architecture.png

Complexity of Microservice Architectures



https://cdn-images-1.medium.com/max/600/1*OhONoCsb1KvesLy-n5ZsSw.png



<http://www.embarc.de/wp-content/uploads/2015/07/Abbildung-1-copy.png>

Stability/Resilience Patterns

- Common architecture patterns are used (Circuit Breaker)
 - The use of patterns is not visible to the outside
 - Effectiveness of patterns only shown to gathered metrics

The
Pragmatic
Programmers

Release It!

Design and Deploy
Production-Ready Software



Michael T. Nygard

Do my resilience mechanisms work as expected?

Conclusion



Does a microservice shutdown
propagate through the overall
system?

State of the Art

Resilience Testing in Microservices

- Simian Army (Netflix)
 - Search Chaos Monkeys (Azure)
- ➡ Only working in a real time environment

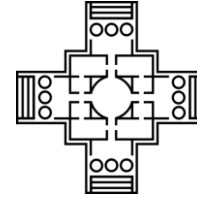


State of the Art

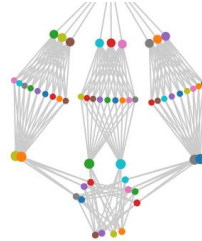
Architecture Simulation for QoS Evaluation



CloudSim



Palladio/SimuLizar



Spigo

Further alternatives are objective of research

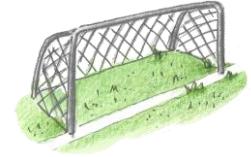
State of the Art

Limitations

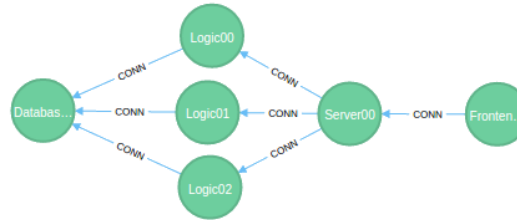
- Resilience testing in production environment is dangerous and expensive
- Most current simulators can't simulate occurrence of failures
- Or are not mature enough (Spigo)



Goals

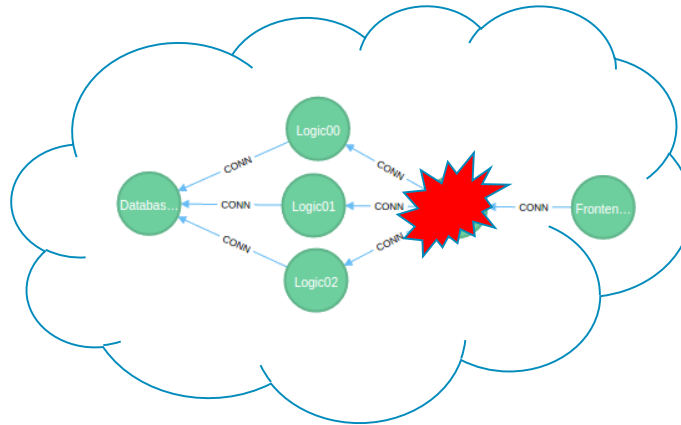


- Offline simulation of microservices



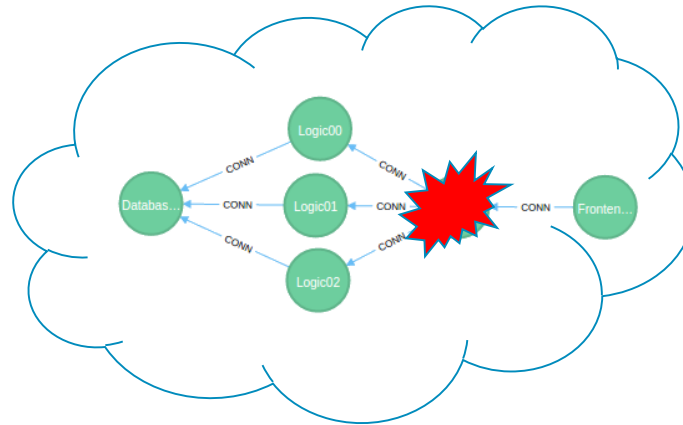
```
{"serviceName":"Frontend","dependency":"Server","count":"1"}  
{"serviceName":"Server","dependency":"Logic","count":"1"}  
{"serviceName":"Logic","dependency":"DataBase","count":"3"}  
{"serviceName":"DataBase","dependency":"","count":"1"}  
{"serviceName":"ChaosMonkey","Killing":"","count":""}
```

Goals



```
{"serviceName":"Frontend","dependency":"Server","count":"1"}  
{"serviceName":"Server","dependency":"Logic","count":"1"}  
{"serviceName":"Logic","dependency":"DataBase","count":"3"}  
{"serviceName":"DataBase","dependency":"","count":"1"}  
{"serviceName":"ChaosMonkey","Killing":"Server","count":"1"}
```

Goals



Output:

Status:

System crashed

Trace:

Server dependency is not running anymore

Solution Approaches



- Research and evaluation for existing solutions
- Development of a simulation tool:
 - Create a new tool
 - Expand an existing tool



Agenda

- Research
 - What is the state of the art and what tools could be used for our purposes?
- Specification
 - What kind of metrics should we use?
 - Input, Output, Error modes, Events, Description language
- Simulation Tool
 - What should our Simulator for microservice architectures with resilience prediction achieve?
- Evaluation
 - What are the results of our work?



Agenda

