Software Design & Architecture

Engr. Abdul-Rahman Mahmood

DPM, MCP, QMR(ISO9001:2000)

armahmood786@yahoo.com
alphapeeler.sf.net/pubkeys/pkey.htm
pk.linkedin.com/in/armahmood
www.twitter.com/alphapeeler
www.facebook.com/alphapeeler
abdulmahmood-sss alphasecure
armahmood786@hotmail.com
http://alphapeeler.sf.net/me

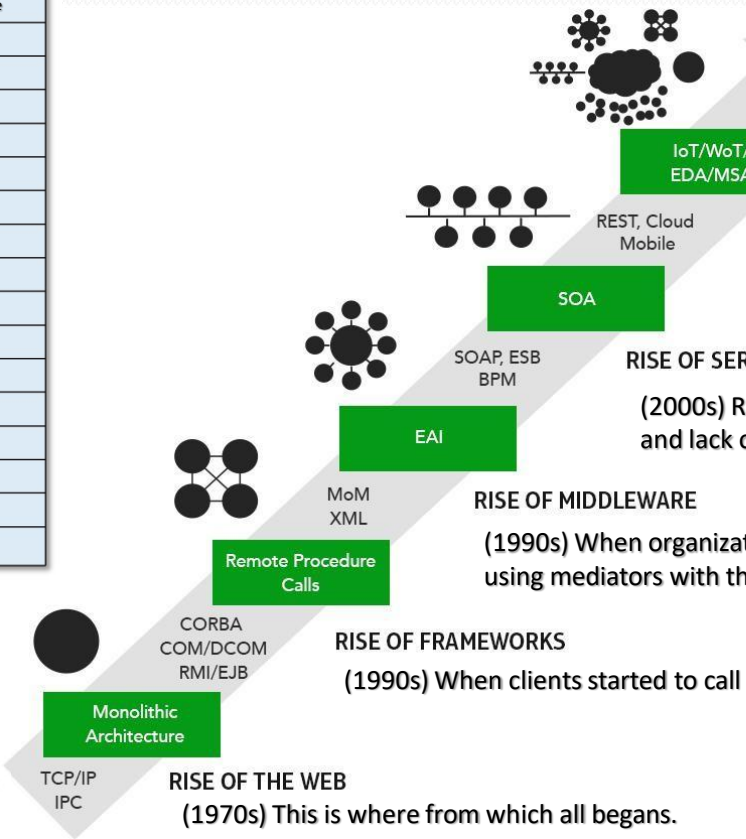
alphasecure@gmail.com
http://alphapeeler.sourceforge.net
http://alphapeeler.tumblr.com
armahmood786@jabber.org
alphapeeler@aim.com
mahmood_cubix 48660186
alphapeeler@icloud.com
http://alphapeeler.sf.net/acms/

Microservices

overview

The architecture journey over the years ...

CORBA	Common Object Request Broker Architecture
EJBs	Enterprise JavaBeans
RMI	Remote Method Invocation
COM	Component Object Model
DCOM	Distributed Component Object Model
EAI	Enterprise Application Integration
MoM	Management Object Model
XML	eXtensible Markup Language
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
ESB	Enterprise Service Bus
BPM	Business Process Model
IoT	Internet of Things
WoT	Web of Things
EDA	Event Driven Architecture
MSA	Microservices Architecture
REST	Representational State Transfer



... and, in this presentation, we will gonna discuss this. **MSA (Microservice Architecture)**

RISE OF THE MACHINES

(2010s) Rise of consumer services that demand highly scalable and resilient distirbuted system.

RISE OF SERVICES

(2000s) Rise of distributed system with a lot of coupling, and lack of indenpence.

RISE OF MIDDLEWARE

(1990s) When organization started to integrate monolith enterprise applications using mediators with the help of standardized messaging formats and processes.

RISE OF FRAMEWORKS

(1990s) When clients started to call methods/functions on server.

RISE OF THE WEB

(1970s) This is where from which all begans.

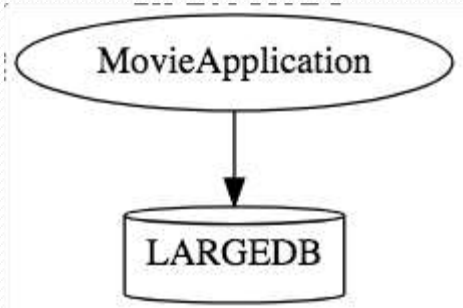
It all started from here ...

Introduction

Microservices is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API.

Microservices allow large systems to be built up from a number of collaborating components. It does at the process level what Spring has always done at the component level: loosely coupled processes instead of loosely coupled components.

In comparison with Monolith application



Monolith applications are typically huge - more 100,000 line of code. In some instances even more than million lines of code. Monoliths are characterized by -

- Large Application Size
- Long Release Cycles
- Large Teams

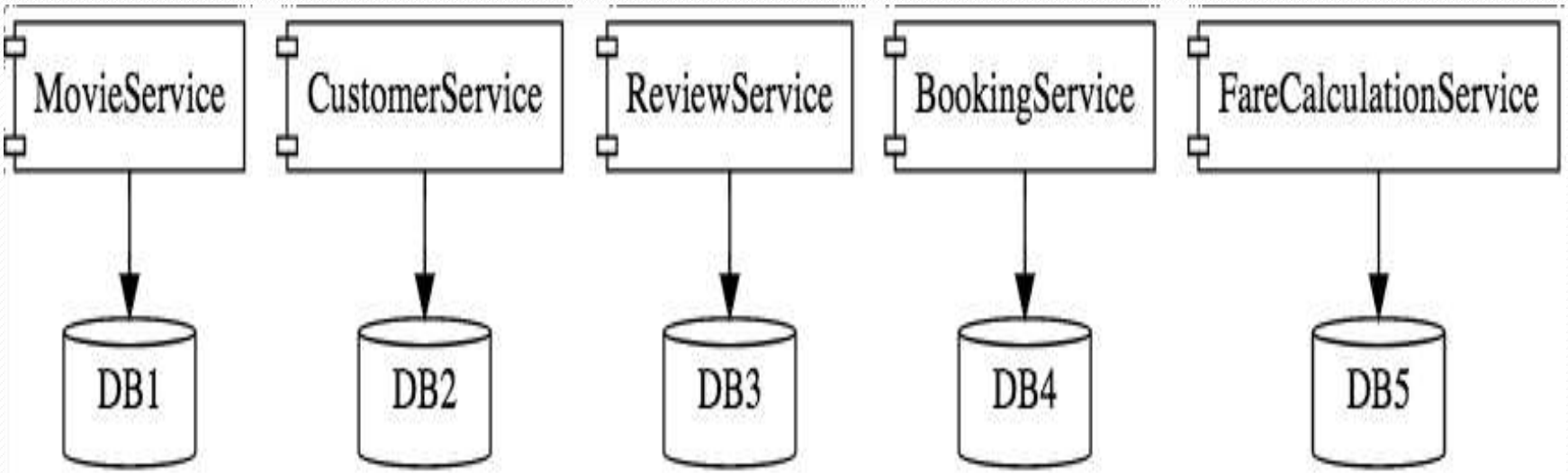
Typical challenges in Monolith application

- Scalability challenges
- New technology adoption
- Difficult to perform automation tests
- Difficult to adapt to modern development practices
- Adapting to device explosion

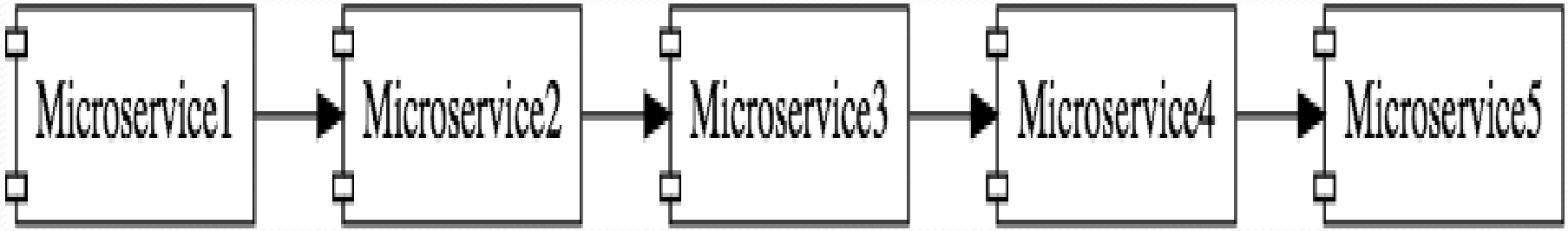
Microservice architectures evolved as a solution to the scalability and innovation challenges with monolithic architectures.

What Does Microservice Architecture Look Like?

This is how the same application would look like when developed using microservices architecture.



involve a number of small, well-designed components interacting with messages.



Advantages of Microservices

- New technology and process adaptation becomes easier. You can try new technologies with the newer microservices that we create.
- Faster release cycles.
- Scaling with the cloud.

Challenges With Microservice Architectures

- Quick setup needed: You cannot spend a month setting up each microservice. You should be able to create microservices quickly.
- Automation: Because there are a number of smaller components instead of a monolith, you need to automate everything - Builds, Deployment, Monitoring, etc.
- Visibility: You now have a number of smaller components to deploy and maintain, maybe 100 or maybe 1000 components. You should be able to monitor and identify problems automatically. You need great visibility around all the components.
- Configuration Management: You need to maintain configurations for hundreds of components across environments. You would need a Configuration Management solution

frameworks for micro services

- 1. Spring Boot with Spring Cloud
- 2. Eclipse Vert.X Microservices framework
- 3. Oracle Helidon Microservices framework
- 4. GoMirco (Golang Microservices framework)