

CS 590 Software Architecture

Final exam solution

[10 minutes]

For which of the following architecture requirements would you use **CQRS**?

Select all architectural requirements so that if this architectural requirements applies, you would apply the CQRS pattern.

- A. When you need future proof systems
- B. When you build a distributed system
- C. When read performance is critical and need to be extremely fast for your microservice
- D. When you design your microservice using Domain Driven Design
- E. When queries and commands have different scaling requirements for your microservice
- F. When your screens start to look very different then your tables in your microservice
- G. When you have an event driven architecture
- H. When you need load balancing for your microservice
- I. When you use a NoSQL database for your microservice
- J. When strict consistency is required for your microservice
- K. When security is important for your microservice

[10 minutes]

Explain clearly the difference(s) between a **Service Oriented Architecture (SOA)** and a **Microservice architecture**.

A SOA uses an ESB to connect the services (orchestration), while in a microservice architecture you do not use an ESB but choreography. SOA have coarse grained services while in Microservices you have small services

[10 minutes]

- a. Explain how **orchestration** works in a distributed system

You have one central orchestrator that talks to all the other systems

b. Give the advantages and disadvantages of **orchestration**

Advantage: Easy to monitor the process

Disadvantage: the orchestrator will become very complex. Does not work well in a large/complex system

[10 minutes]

a. Explain how **choreography** works in a distributed system

You do not have one central orchestrator that talks to all the other systems. The individual systems know how to talk to the other systems

b. Give the advantages and disadvantages of **choreography**

Disadvantage: Difficult to monitor the process

Advantage: No complex orchestrator. Works well in a large/complex system

[10 minutes]

a. Explain clearly **Conways law**

Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations

b. Explain clearly what Conways law means for a microservice architecture.

If you want to use a microservice architecture in your organization, you have to change the structure of the organization so that the structure of the organization is the same as the structure of the microservice architecture. Every Scrum/devops team controls one or more microservices.

[15 minutes]

a. Explain clearly why local logging does not work in a microservice architecture

Containers come and go dynamically

You want to see the overview of what happens in the microservice architecture, and you only get that if you collect all logging in one place.

- b. Explain clearly how logging should be implemented in a microservice architecture.

Collect all logging in one place. You can do that using the ELK stack. You can also publish logging to kafka and collect it centrally.

[2 minutes]

In one short sentence explain the problem that **Zipkin** solves

Distributed tracing

[2 minutes]

In one short sentence explain the problem that **Ribbon** solves

Client side load balancing

[2 minutes]

In one short sentence explain the problem that **Zuul** solves

API gateway

[2 minutes]

In one short sentence explain the problem that **Eureka** solves

Registry

[2 minutes]

In one short sentence explain the problem that **Hystrix** solves

Resilience

[10 minutes]

Explain clearly how a **circuit breaker** works in a microservice.

We set a timeout on every remote call

If the call takes longer than the timeout, the circuit breaker trips, so that the next time we call this service we fail fast without waiting for the timeout to occur. If we fail fast, we call an alternative method. The circuit breaker keeps checking if the remote service will come online

Explain the 3 techniques that a circuit breaker uses.

Fail fast: Service A gets an immediate response

Fail gracefully: Service A can fall back to an alternative

Recover seamlessly: Circuit breaker will periodically check if Service B is back online

[15 minutes]

Kafka uses **event sourcing** for its messages.

- a. Explain clearly what this means.

Event sourcing means we store the events instead of the state.

- b. Explain clearly why kafka uses event sourcing. What are the advantages.

Because messages are never changed or deleted it does not matter how many publishers or subscribers we have on one topic.

[10 minutes]

For securing a microservice architecture we looked at **Oauth2** and **JWT**.

- a. Explain the difference between OAuth2 and JWT.

OAuth2 is a standard for authorization. It does not define a standard for tokens

JWT is a standard for tokens

- b. What does JWT add to OAuth2/ In other words, why do we need JWT if we have OAuth2?

JWT defines a standard for the token. The token contains the role of the user and JWT will sign and encrypt the token so that the receiver of the token can securely retrieve the role of the user without having to call the authentication server.

[10 minutes]

Describe how we can relate the **API Gateway** to one or more principles of SCI. Your answer should be about 2 to 3 paragraphs. The number of points you get for this questions depends how well you explain the relationship between the **API Gateway** to one or more principles of SCI.