

W3WI_110.2 - Distributed Systems

Lecturer: Prof. Dr. Michael Eichberg
Contact: michael.eichberg@dhbw.de, Raum 149B
Version: 2.0 (WWI23SCB)

Core Topics

- Terminology, concepts, architectures, requirements profiles and architecture models for distributed systems
- Design and implementation approaches
- Comparison of different middleware concepts
- Synchronous and asynchronous communication, remote method calls
- Asynchronous communication and messaging systems
- Security aspects in distributed systems

Previous Knowledge

■ Who is familiar with Java?

Most

■ Who is familiar with Python? Basically none

■ Who is familiar with JavaScript/Typescript? Not really discussed as part of the Web Engineering lecture.

■ Who knows what a RESTful API is? Some knowledge.

■ Who has ever logged in to a server via SSH? No real experience.

■ Who has ever done any administration of a Linux server?

■ Who is familiar with Unix/Linux/Mac OS command line tools?

■ Who has done any development outside of university projects?

Examination - Portfolio

Background

- the module has 55 lecture hours
- the lecture *Distributed Systems* has 22 lecture hours
⇒ Distributed Systems will contribute up to **50** points to the final grade for the module. (Please, don't do the math.)

2 Parts

01 Presentations - max. 15 points

02 Programming Exercise - max. 35 points

Attention!

Students who present together should not be in the same team for the programming exercise.

Presentations - General Conditions^[1]

- Each person should present for 10 Minutes sharp!
- The presentations should deal in particular with the core content of the lecture and *be of a conceptual nature*.

This means that, after briefly presenting the overall purpose of the framework/technology/protocol, the architecture/the details must be presented. That is, how errors are dealt with, which services are offered, which guarantees/security aspects are implemented, how scalability is achieved, etc.

No promotional presentations!

- Presentations should be in English.
- The speakers should not rotate several times during the presentation. I.e. the first speaker presents first, then the second, and so on. This is necessary for the grading.

Attention!

The presentations are to be uploaded to Moodle the evening before the agreed date.

[1] Further Recommendations (in Ger.)

Presentations - Available Topics

26.6.

Virtualization and Virtualization Platforms [7P]:

The goal of this presentation is to give an overview of the different virtualization technologies and to explain the commonalities and differences between them. It is also possible to demonstrate some of the differences using concrete products.

Some keywords: Containers (e. g., Docker, Linux), Firecracker, Hypervisors (KVM, Xen, Hyper-V), Docker Swarm, Kubernetes, Proxmox, Open Stack

The Students giving this presentation are not allowed to work in one team.

Web-App Security [3P]:

Web application security: CORS (Cross-Origin Resource Sharing), SOP (Same-Origin Policy), CORP / COOP / COEP (Cross-Origin Resource/Opener/Embedder Policies), CSP (Content Security Policy)

Introduction and concrete examples how they are used/specified and help prevent attacks.

Monitoring & Debugging Distributed Systems [1P]:

Open Telemetry and Log Aggregation

Network Protocols [3P]:

- HTTP/3 and QUIC [2P]
- BitTorrent [1P]: Usage and Protocol

11.7.

Consensus Algorithms [4p]:

- Paxos [2P]
- RAFT [2P]

Eventual Consistency [2P]:

Gossip Protokoll

Leader Election [1P]:

Bully Algorithm and/or Ring Algorithm

gRPC [2P]:

Google RPC/Protobuf; a comparison with alternative is also possible.

Fault Tolerance [2P]:

(Practical) Byzantine Fault Tolerance


Quorum Systems [1]:

in particular majority voting (i. e., quorum -distributed computing)

Distributed File Systems [2P]:

for example GFS, HDFS and Ceph

Portfolio Task

 See Moodle for task description.