

Distributed Systems (DSy)

Admin

Thomas Bocek

24.02.2022

About Me

- OST, Lecturer
 - Distributed systems and blockchains
 - Previously (VSS/DSA/DS1/DS2,WED1,3)
- Co-founder **Axelra AG**
 - May 2019
 - Tech Delivery, MVP in 100 days - Digitalization / Blockchain
 - E.g., **THORWallet** (DeFi app for cross chain assets)
 - CTO, operative
- Co-founder **modum.io AG**
 - Since 2016
 - Digitalizing Supply Chain - Digitalization / Blockchain
 - Board member, strategic
- General interests: distributed systems, blockchain, startups



DSy Administration

- Module Description
- Website: <https://dsl.i.ost.ch/>
- Exercises: Teams
 - Team code: 2tbddgw (works only for OST students)
- Lectures: pre-recorded, videos available on <https://dsl.i.ost.ch/lect/fs22/> and Youtube
 - Videos and PDFs will be uploaded before the lecture starts on Friday 08:10
 - Youtube will only have the lectures videos, admin related videos & PDFs only via <https://dsl.i.ost.ch/lect/fs22/>
 - Same as in previous semester FS21

Distributed Systems

Kurzzeichen:

M_DSy

Durchführungszeitraum:

FS/22-FS/23

ECTS-Punkte:

4

Lernziele:

Verteilte Systeme sind allgegenwärtig in beruflichem und privatem Alltag; sie führen Konzepte und Technologien aus unterschiedlichen Bereichen der Informatik zusammen. Die Themengebiete umfassen:

- Protocols for Distributed Systems
- Fully Distributed Systems and NAT issues
- BitTorrent, Tor, Rsync
- RPC and Queues
- Blockchain Introduction
- Smart Contracts and Cryptocurrencies

Das Lernziel in diesem Modul ist, dass Studenten ein verteiltes System selbständig entwickeln können.

Verantwortliche Person:

Bocek Thomas

Standort (angeboten):

OST Rapperswil

Empfohlene Module:

Betriebssysteme 1 (M_Bsys1, HS/19-HS/20), Software-Engineering 1 (M_SE1, HS/17-HS/20)

Zusätzlich vorausgesetzte Kenntnisse:

Englisch

(Dieses Modul wird teilweise in Englisch gehalten.)

Dsy Administration

Kurzzeichen:

DSy

Lernziele:

Plan und Lerninhalt:

Die Themengebiete können variieren und werden in der ersten Vorlesung bekanntgegeben.

- If there is interest in further topics (distributed systems and blockchain related), please talk to me.

DSy 2022: Plan

Nr	Date	Lecture*	Lecturer
01	25.02.2022	Admin, Introduction	Bocek
02	04.03.2022	Protocols, Basics	Bocek
03	11.03.2022	Containers and VMs	Bocek
04	18.03.2022	Distributed Applications: Load Balancing	Bocek
05	25.03.2022	Authorization (Basic, Digest, Session, JWT and OAuth)	Bocek
06	01.04.2022	Application Protocols and RPC	Bocek
07	08.04.2022	Deployment (Docker Swarm, Kubernetes)	Bocek
-	15.04.2022	- No Lecture (Spring Vacation)	-
08	22.04.2022	Blockchain, Bitcoin, Ethereum Introduction	Bocek
09	29.04.2022	Smart Contracts	Bocek
10	06.05.2022	Guest lecture	tba
11	13.05.2022	Challenge Task Presentations I	you
12	20.05.2022	Challenge Task Presentations II	you
-	27.05.2022	- No Lecture	-
13	03.06.2022	Guest lecture	tba
14	10.06.2022	Challenge Task Award Winner Announcement, Q&A	Bocek

DSy Exercises

- Challenge Task
 - Starts 04.03.2022 (lecture 2)
 - **No exercises today!**
- Goal: practical application of distributed systems concepts and tools
- Online: via MS Teams
 - Friday exercise u11: 10:10 – 11:50
 - Friday exercise u12: 13:10 – 14:50

Plan und Lerninhalt:

Die Themengebiete können variieren und werden in der ersten Vorlesung bekanntgegeben.

In den Übungen entwickeln Studentengruppen eine einfache verteilte Applikation (Challenge Task I), welche diverse Requirements erfüllen muss. Das Erfüllen dieser Requirements ist Voraussetzung für das Bestehen der Übungen und für die Zulassung zur Prüfung. Ein Beispiel eines Challenge Tasks könnte ein vollständig verteiltes Tic-tac-toe sein.

BlCh Challenge Task

- Leistungsbewertung

Während der Prüfungssession:

Schriftliche Prüfung, 60 Minuten

Zulassungsbedingungen zur Prüfung:

Nach Ankündigung, Abgabe erfolgreich gelöste Übungsaufgaben.

- Simple Frontend (e.g., HTML, Vue, React, Svelte)
- Loadbalancer(e.g., traefik, nginx, HAproxy, Caddy)
- Two instances of a service (your choice), and during the challenge task presentation, one instance will be shut off.
- One storage backend

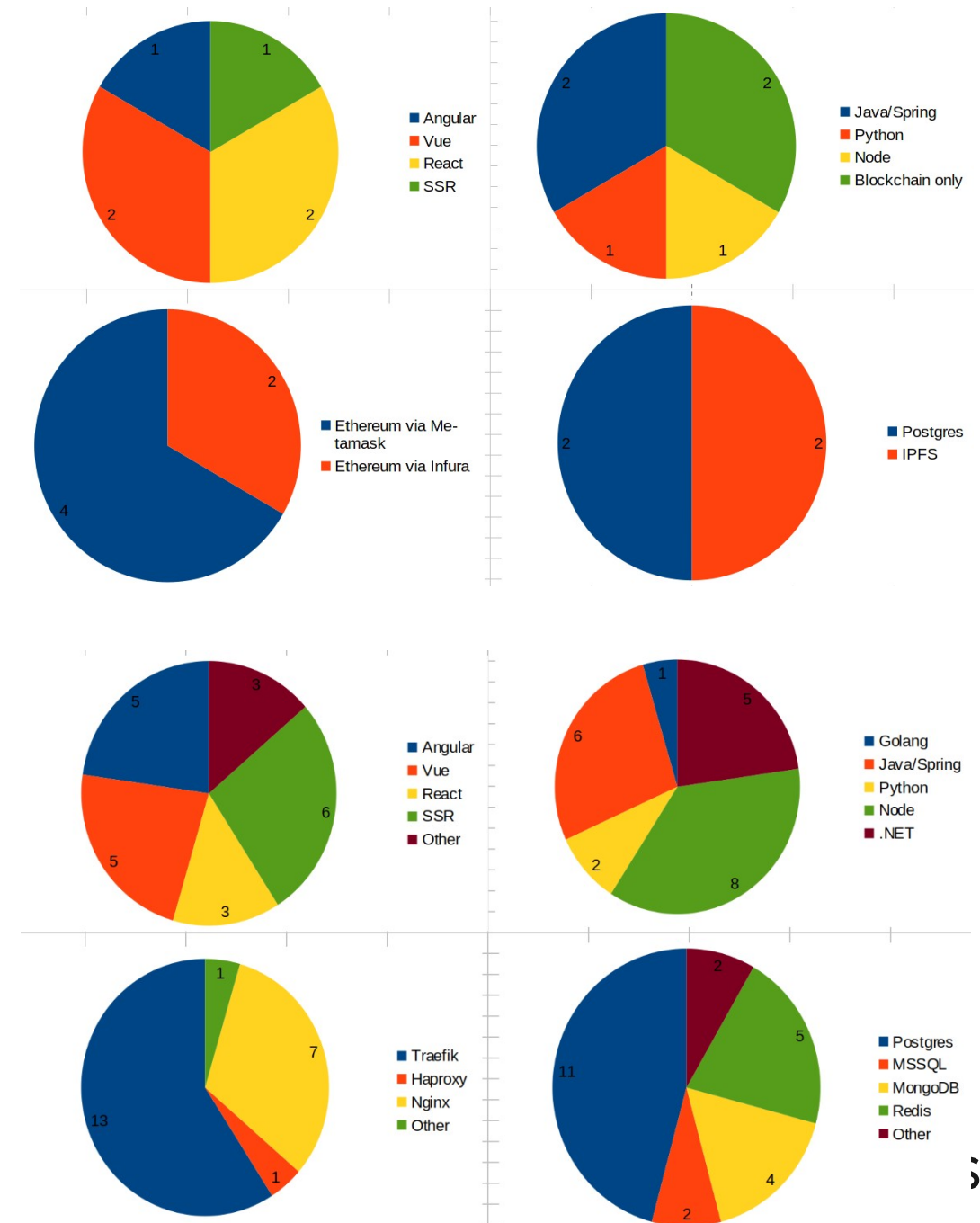
- Challenge Task FS 2022
 - Design and implementation of a simple distributed system (of your choice), with at least one service with a websocket, and where a service instance can fail
- Requirements
 - Load balancing with scalable service
 - Failover of a service instance
 - A websocket needs to send data from the service to the frontend
 - Dockerized
 - Persistent storage (storage does not need to be scalable, but you can build it scalable if you want)
 - Use latest stable releases of chosen libraries and frameworks

DSy 2022: Plan

Nr	Date	Lecture*	Suggested Milestone
01	25.02.2022	Admin, Introduction	Group forming, idea for app
02	04.03.2022	Protocols, Basics	Setup dev environment
03	11.03.2022	Containers and VMs	Information flow
04	18.03.2022	Distributed Applications: Load Balancing	
05	25.03.2022	Authorization (Basic, Digest, Session, JWT and OAuth)	
06	01.04.2022	Application Protocols and RPC	
07	08.04.2022	Deployment (Docker Swarm, Kubernetes)	Hand-in 1 initial version (14.04.2022, 23:59 CET)
-	15.04.2022	- No Lecture (Spring Vacation)	
08	22.04.2022	Blockchain, Bitcoin, Ethereum Introduction	
09	29.04.2022	Smart Contracts	Connecting everything
10	06.05.2022	Guest lecture	Testing, testing, testing
11	13.05.2022	Challenge Task Presentations I	Hand-in 2 final version (12.05.2022, 23:59 CET)
12	20.05.2022	Challenge Task Presentations II	
-	27.05.2022	- No Lecture	
13	03.06.2022	Guest lecture	
14	10.06.2022	Challenge Task Award Winner Announcement, Q&A	

DSy Challenge Task

- Hand-in 1: 14.04.2022, 23:59 (CET) - initial version of your project
- Final hand-in: 12.05.2022, 23:59 (CET)
 - Well documented code and infrastructure
 - Presentation (slides) and demo of the application, also showing the architecture and design decisions
- Libraries and tools
 - Allowed to use any language, framework, and platforms. However, the supervisors are familiar with those: Java, Golang, JavaScript, Linux (see FS21, HS21)
- If you **fail** the CT, you **fail** this course



DSA Challenge Task 2018

- Design and implementation of a fully-distributed, P2P messaging application with an integrated, blockchain-based notary service
- Requirements
 - Full decentralization and P2P
 - Send messages between peers
 - Quickly and easily verify messages
- Each winner: 0.01311 BTC (50CHF)
 - They used the IOTA blockchain as a signaling layer for WebRTC.
 - Everything decentralized



DSA Challenge Task 2019

- Design and implementation of a fully-distributed, P2P gambling game
- Requirements:
 - Fully decentralization and P2P communication
 - Waiting room with chat
 - Fully decentralized gambling
- Each winner: 0.0075 BTC (~50CHF)
 - Simple yet efficient design
 - Blockchain-based four-in-a-row
 - Technology: Ethereum with web3.js and WebRTC



DS1/VSS Challenge Task FS 2020

- Free choice project
- Requirements
 - Use at least one text protocol, at least one binary protocol
 - Load balancing
 - Scalable service with authentication
- Each winner: 5.27mBTC (0.00527BTC)
 - They gathered weather data from multiple locations, and they used many components: ActiveMQ, Eureka discovery services, Kibana, Docker compose, running on self-hosted gitlab.



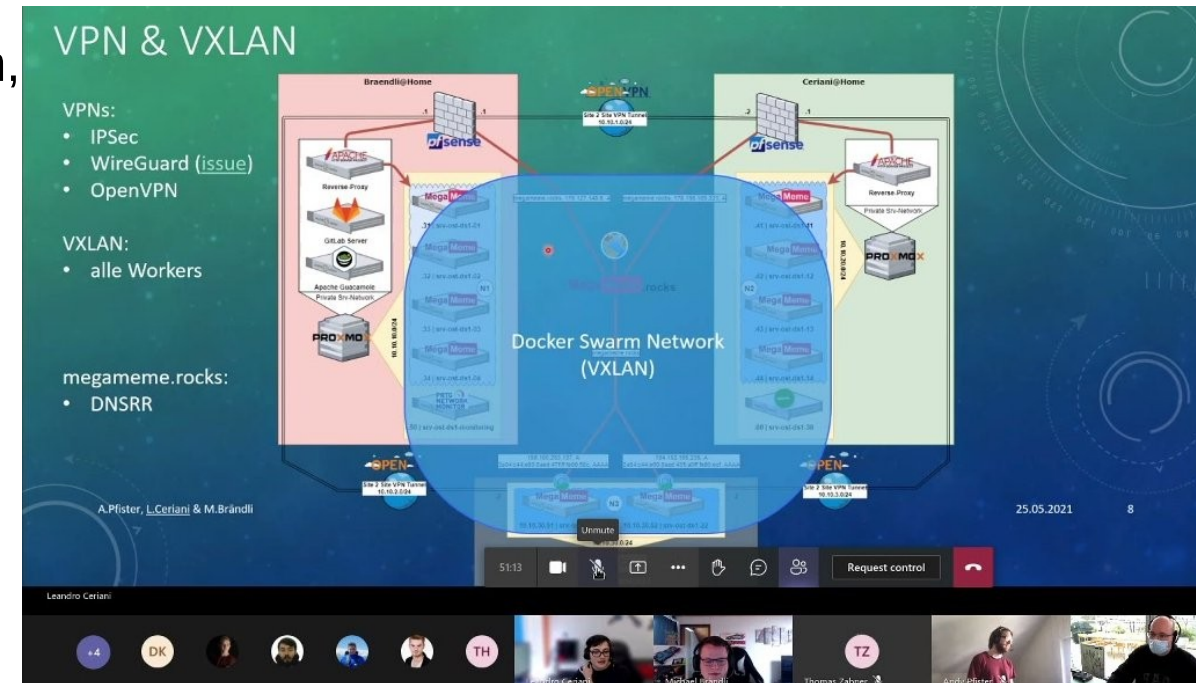
ADSB Challenge Task HS 2020

- Design and implementation of a decentralized financial application
- Requirements
 - Full decentralization and P2P mechanisms for the financial product (e.g., WebRTC, TomP2P, libp2p) or use scalable mechanisms ready to be deployed on the cloud
 - Graphical (GUI/Web-based) frontend
 - Public blockchain based
- Each winner: 2.625 mBTC (0.002625 BTC)
 - Donation platform and used the blockchain to keep track of donations



DS1 Challenge Task FS 2021

- This semester's challenge task (CT) is the design and implementation of a simple distributed system, with the following components:
 - Simple Frontend (e.g., HTML, Vue, React, Svelte)
 - Loadbalancer(e.g., traefik, nginx, HAproxy, Caddy)
 - Two instances of a service (your choice)
 - One storage backend (Postgres, simple KV memory storage)
 - JWT Authentication
- Each winner: 1.35 mBTC (0.00135 BTC)
 - Voting meme platform with Svelte and Ruby on Rails with docker swarm, GlusterFS, and CockroachDB



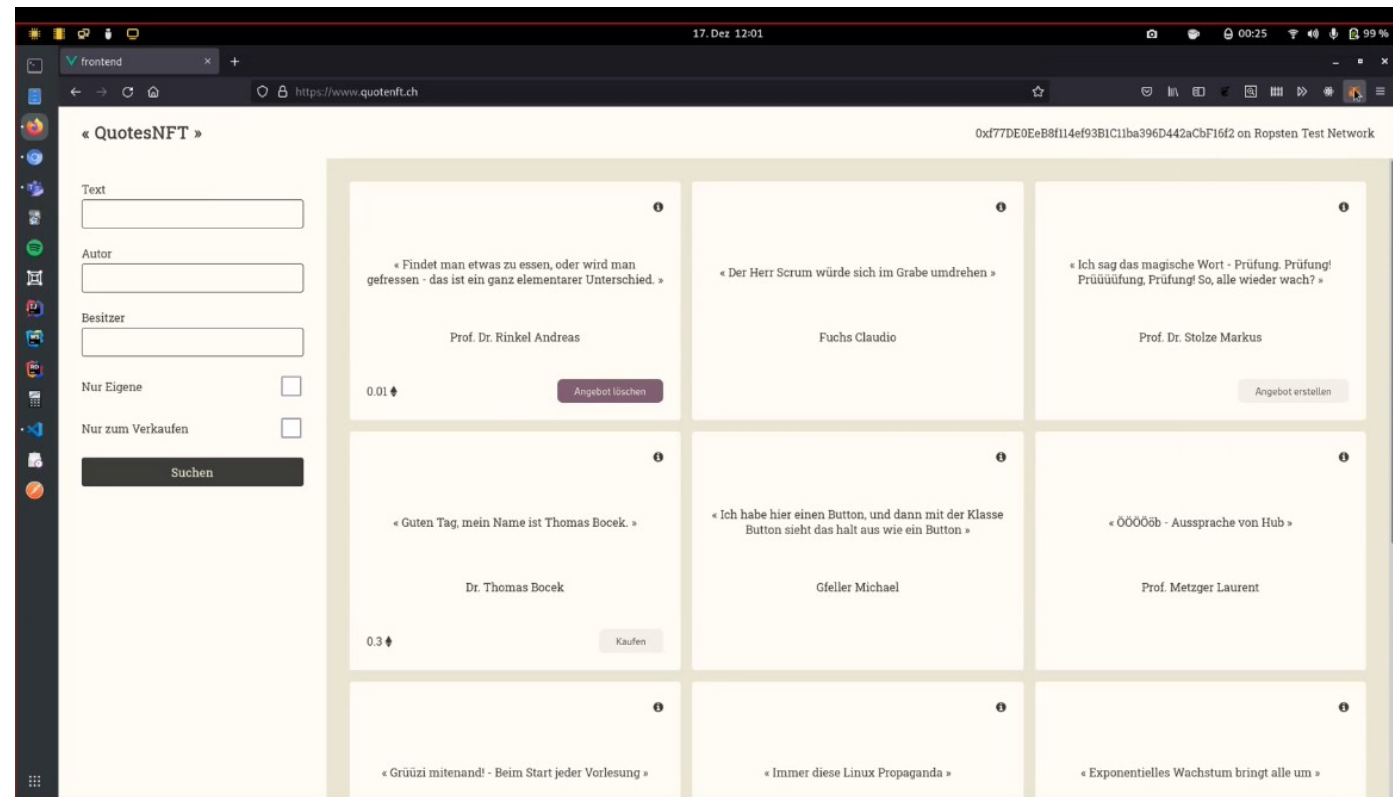
BlCh Challenge Task HS 2021

- This semester's challenge task (CT) is the design and implementation of an application including NFTs. Ideally, the application is a decentralized application (Dapp).

Requirements:

- A working prototype for your use-case
- Use latest stable releases of chosen libraries and frameworks
- An NFT stored on a public blockchain (can be testnet)
- Transfer of an NFT triggered directly or indirectly via the frontend/backend (can be testnet)
- Each winner: 1 mBTC (0.001 BTC)

- QuoteNFT is a platform to trade NFT quotes. The trading happens on a smart contract and is fully decentralized. Their project uses Vue.js for the frontend, node and express with TypeScript for the backend, and MetaMask to connect to Ethereum. Their QuoteNFT platform uses Traefik as a reverse proxy and docker-compose to orchestrate the services

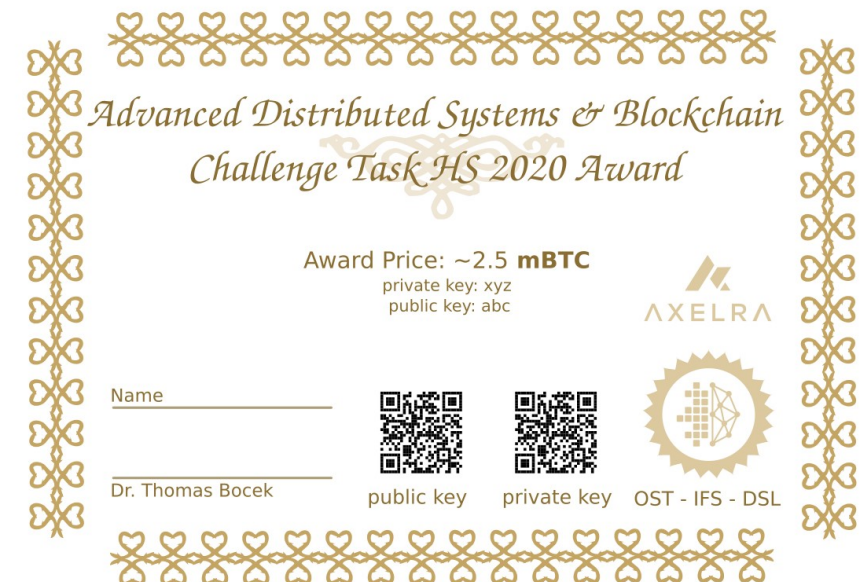


DSy Recommendation

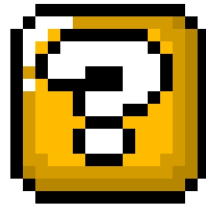
- Follow the lecture closely
 - Lecture slides contain relevant
 - Reading further literature is recommended
 - Link are provided in the slides
- Post-processing the lecture
 - If topics or statements from the lecture are not clear, research and resolve them
- Actively participate in the challenge task
 - Trying to understand it theoretically is not enough – hands-on is key!
 - If its not working, **read logs**, try to figure it out by yourself. If its still not working, ask
 - Do not miss the opportunity of discussing details with your supervisor
- 10 lectures
 - (25h = 20h + 5h; ca. 2.5h / week)
 - ~2h frontal teaching
 - ~1/2 h post-processing
- 12 exercises / challenge task
 - (72h = 60h + 12h ca. 6h / week)
 - ~5h challenge task
 - ~1h post-processing
- Exam
 - ~23h exam preparation
- 4 ECTS = 120h
- Prüfung:
 - Closed book 60min, no coding, Hilfsmittel: none

Summary

- Lectures Videos online before Friday Morning – 08:10
- Challenge task, group members: send to thomas.bocek@ost.ch until next week
 - Work on challenge task during exercise hours and at home
- 6 requirements to be met
 - Load balancing with scalable service
 - Failover of a service instance
 - A websocket needs to send data from the service to the frontend
 - Dockerized
 - Persistent storage (storage does not need to be scalable, but you can build it scalable if you want)
 - Use latest stable releases of chosen libraries and frameworks
- All 6 requirements must be fulfilled to pass this course
- Deliverables
 - 14.04.2022, 23:59 (CET) first hand-in of your initial version
 - 12.05.2022, 23:59 (CET) - well documented infrastructure, source code, and presentation
- Presentations during on the 13.05.2022 and 20.05.2022 via MS Teams
- OST DSy Challenge Task Winner FS 2022
 - 10.06.2022



Questions?



Dr. Thomas Bocek
thomas.bocek@ost.ch