Practical Lab

Cloud Systems Engineering

(cloud-lab)

Chair of Decentralized Systems Engineering
https://dse.in.tum.de/

Welcome to the cloud-lab!

Goals of the lab

- Learn how to build, deploy and manage cloud systems
- Gain knowledge of properties and challenges of distributed systems

Most importantly: have fun!

Your instructors

Emmanouil Giortamis

Maurice Bailleu

Dimitra Giantsidi

Jiyang Chen

Simon Ellmann

Masanori Misono

What we do

Chair of Decentralized Systems Engineering

Research topics:

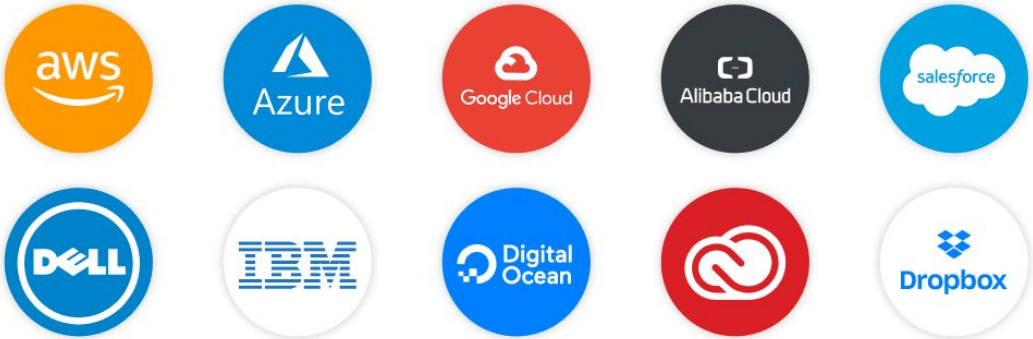
- OS virtualization (VMs, containers, …) and emulation (QEMU)
- High performance I/O (DPDK, SPDK, RDMA)
- FPGAs, smart NICs, smart SSDs
- Persistent memory
- …

Looking for a bachelor’s or master’s thesis, guided research or IDP?

→ https://dse.in.tum.de/theses/

Cloud computing

- Cloud computing is powering the Internet
  - Large-scale computing resources
  - On-demand and cost effective
  - Geo-distributed data centers



Motivation: Cloud systems engineering

- Cloud systems
  - Modern cloud systems handle millions of users, TBs of data
  - Cloud software systems employ large geo-distributed data centers
- How can we build cloud systems that:
  - Scale seamlessly ?
  - Are highly available ?
  - Are fault tolerant ?
  - Are easily configurable?
  - Are easily maintained?
- Cloud systems engineering aims to achieve all the above
  - In a cost-effective manner

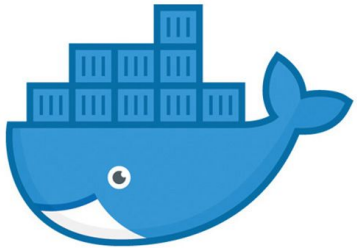
Our focus: Learning goals

- Part I: Cloud systems workflow
  - Container: How to build applications using containers?
  - Cluster orchestrators: How to deploy jobs?
- Part II: Distributed systems system architecture
  - Sharding / re-configuration of servers
  - Fault tolerance / replication
  - Consistent hashing
  - Consistency
  - ~~Transactions / data management~~
  - ~~Distributed locking / synchronization~~
  - Concurrency and high-performance architectures
  - Fault detection
  - Configuration management

Learn by building an end-to-end system!

Technologies

APACHE ZooKeeper™



RocksDB

Format

- A set of four programming tasks:
  - Each related to a different aspect of distributed systems
  - Built on top of each other, like a stack
- For each task, we will provide
  - Necessary background via a lecture
  - Q&As: after lecture and online via Slack
- Team effort: 1-2 students per team
  - The teams need to be finalized by the end of the first task
- The submitted tasks will be evaluated by
  - Automated grading system
  - Instructors

Layered architecture

~~#4: Distributed TXs: w/ and w/o replication~~

#3: Replicated distributed KVS

#2: Distributed KVS

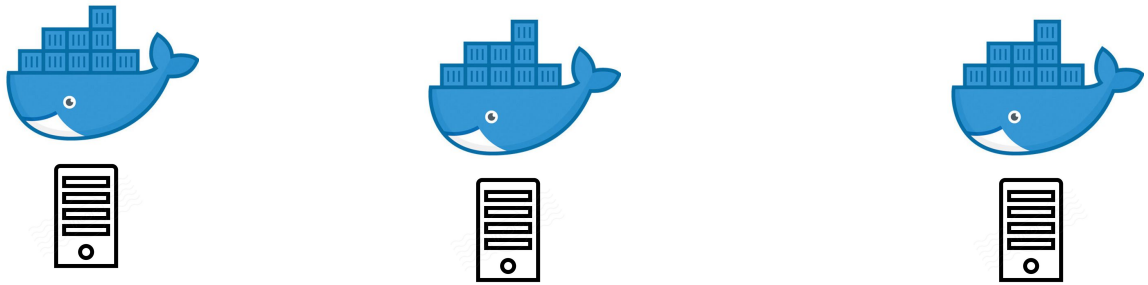
#1: Single-node KVS

#1: Single-node KVS

…

#1: Single-node KVS

#0: Container and job deployment



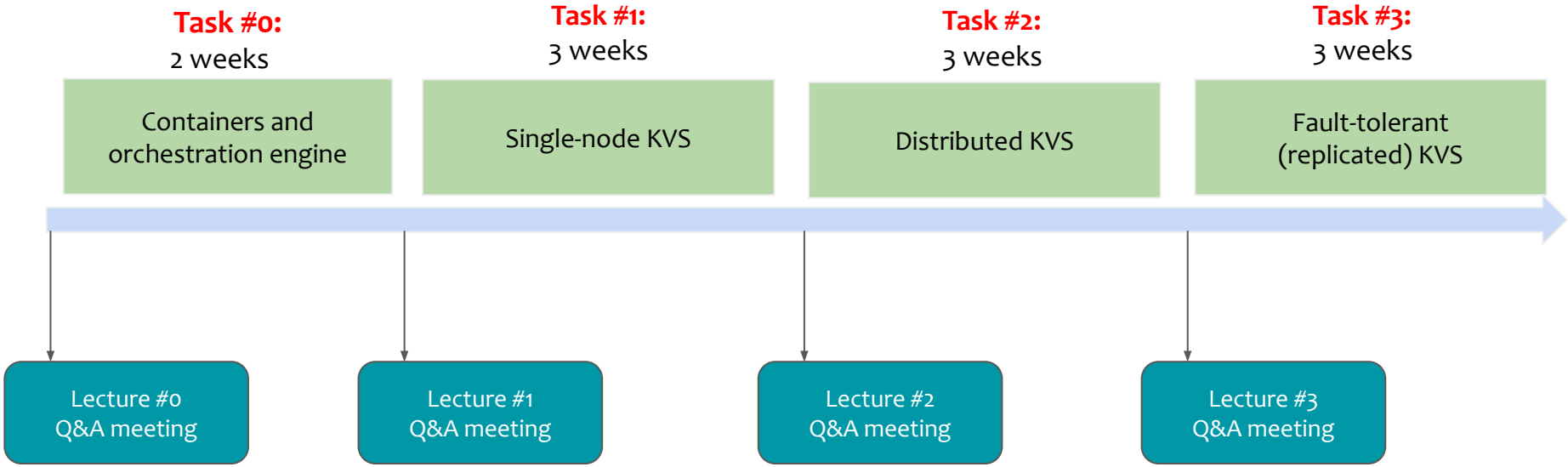
APACHE ZooKeeper™

Grading

Lecture	Category	Details	Grade
	Kick-off meeting	Introduction	-
#0	Container deployment	Deploying an application with containers	10%
#1	Single-node KVS setup	Build and deploy a single node KVS as a container	30%
#2	Distributed KVS	Shard the keys across multiple nodes: fault detection and server reconfiguration	30%
#3	Replicated distributed KVS	Replicate the KVS instances across these nodes for fault tolerance	30%
~~#4~~	~~Distributed Transactions~~	~~Support distributed transactions across keys and nodes: w/ and w/o replication~~	~~40%~~

Timeline

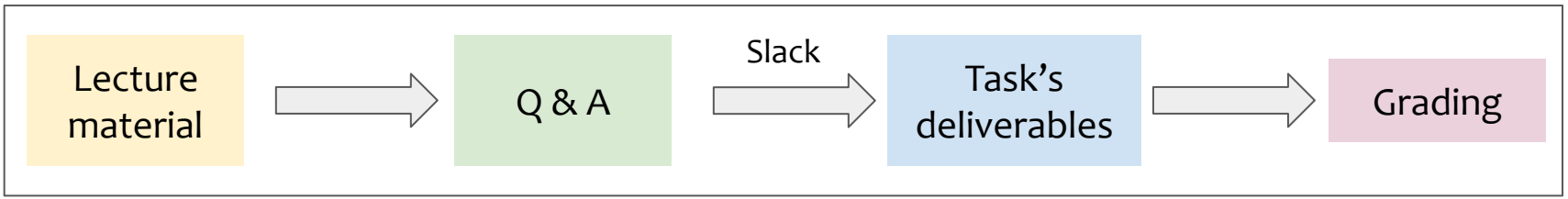
First week (kick-off meeting)



Lab organization

- Recorded lectures
  - Four recorded lectures
    - Lecture videos will be released on Slack on Mondays before the Q&A
- Q&A session
  - Four online Q&As
    - The Thursday **after** the video release, at 10:00 - 11:00 hr
    - Video for online Q&A: See the slack channel description
- Communication via Slack: https://ls1-courses-tum.slack.com/
  - Register with your TUM email address and join #ss-22-cloud-lab

Life of a task



Important dates

Q&A lectures	Task duration	Lecture video release (Mondays, 10:00h)	Q&A (Thursdays, 10:00-11:00h)
kick-off	NA	NA	28th April
#0	2 weeks	2nd May	5th May
#1	3 weeks	16th May	19th May
#2	3 weeks	6th June	9th June
#3	3 weeks	27th June	30th June

Grading

- Submission system: GitHub classroom
  - Template repository for each task with detailed instructions and test cases
- Automated tests
  - Tests run on our self-hosted CI runners
  - Grades determined by test scores
- Grading scheme:

Score	Grade
[0 .. 20]	5.0
[21 .. 30]	4.7
[31 .. 40]	4.3
[41 .. 50]	4.0
[51 .. 60]	3.7

Score	Grade
[0 .. 20]	3.3
[21 .. 30]	3.0
[31 .. 40]	2.7
[41 .. 50]	2.3
[51 .. 60]	2.0

Score	Grade
[0 .. 20]	1.7
[21 .. 30]	1.3
[31 .. 40]	1.0

- Exam[illegible]one by us – you do not have to register yourself!

Contact

- Emmanouil (Manos) Giortamis
  - emmanouil.giortamis@in.tum.de

- All cloud-lab info: https://github.com/TUM-DSE/cloud-lab

Workspace: http://ls1-courses-tum.slack.com/

Website: https://dse.in.tum.de/

Channel: #ss-22-cloud-lab

Join us with TUM email address (@tum.de)