

Cloud Computing Technologies

DAT515 - Fall 2024

Course Information

Prof. Hein Meling



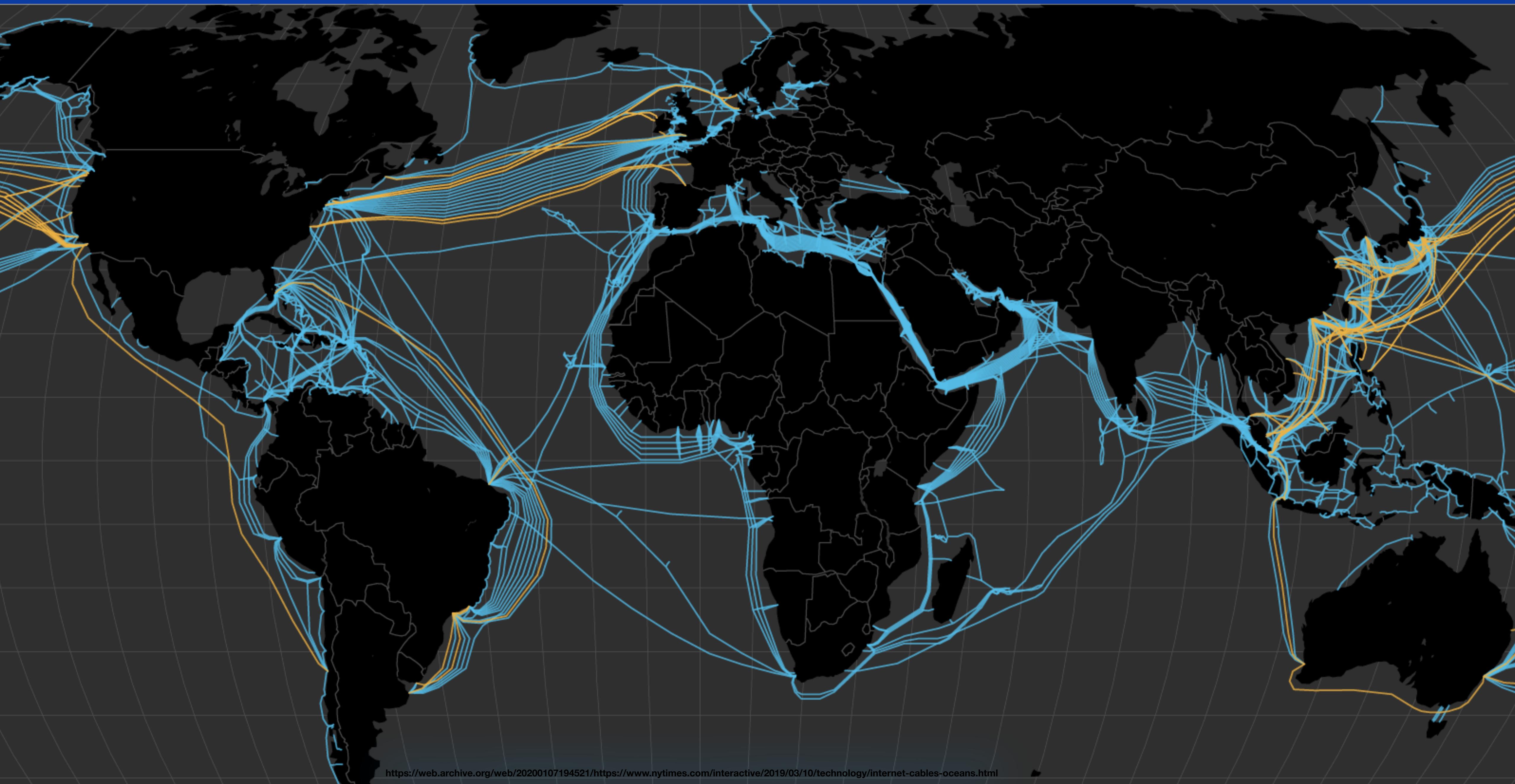
WHERE
THE HECK
IS MY DATA?

IT IS
UP THERE IN
IN THE CLOUD.



‘People think their data is
in the cloud.’

It's not!



About us

Course Instructor: Prof. Hein Meling

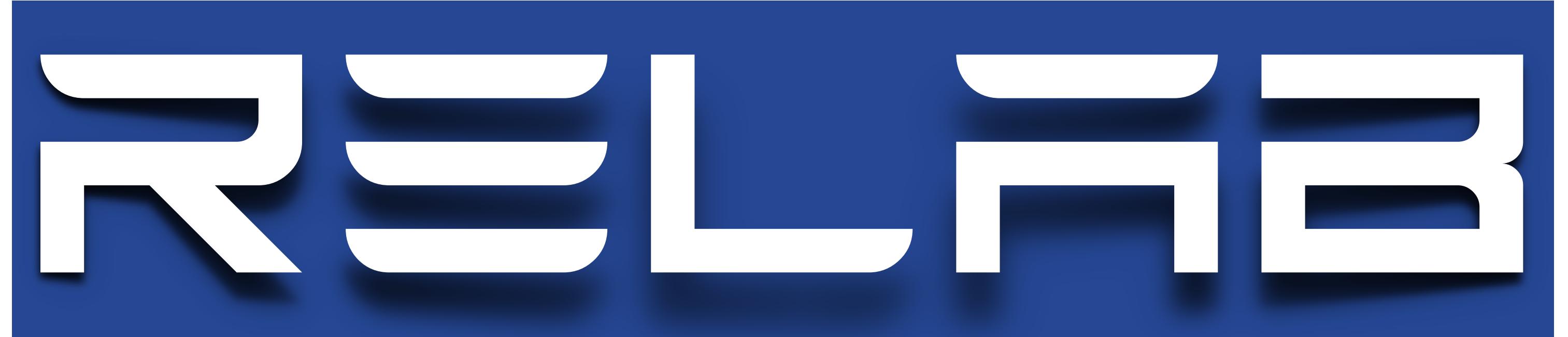
- hein.meling@uis.no
- Office: KE D427 (no office hours)
 - Talk to me between lectures
 - Come by the office – I may be busy ...
 - Send email to book an appointment
- Research interests: Distributed Systems
 - Fault tolerance, availability, security, reliable storage, blockchain systems, etc...



Teaching Assistants

- Jayachander Surbiryala
 - Office KE D4xx
 - jayachander.surbiryala@uis.no
- Maroof Mushtaq
 - m.mushtaq@stud.uis.no
- ...





{Resilient | Reliable} Systems Lab



2018-2023

BBChain (UiS & UiT): An Efficient Trustworthy Distributed Document Verification System

2019-2022

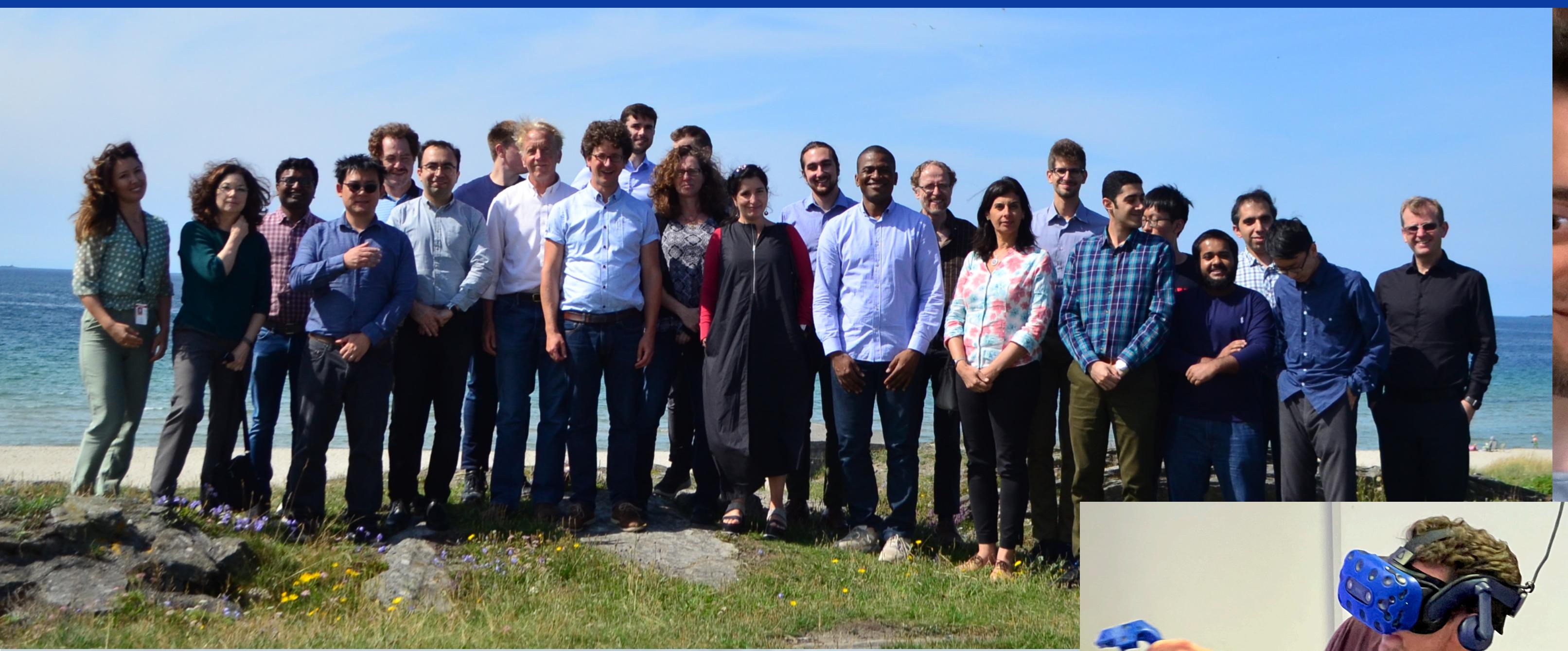
Credence (USA, Canada, Brazil, and Norway research institutions): Collaboration Network for Excellent Education and Research in Dependable and Secure Distributed Systems

2021-2022

Leander Sabbatical: Technische Universität Braunschweig

2022-2023

Hein Sabbatical: University of California, Berkeley



Who are you?

- What's your study program?
 - Computer Science
 - Data Science
 - Others?
- Who is working as a developer?
 - What do you do?
- Experience using cloud platforms:
 - AWS, Azure, Google Compute, Others?
- Experience using cloud technologies:
 - Docker
 - APIs for online services
 - GitHub API, Discord API,...
 - Kubernetes?

Learning Outcomes?

Knowledge

- Understand elements of Cloud infrastructures.
- Characterize and compare typical service models like IaaS, PaaS, and SaaS.
- Compute abstractions, including virtual machines, containers, and serverless computing.
- Storage abstractions with varying consistency requirements.
- Resource management, including storage and container management.
- Understand and compare the most common commercial Cloud offerings.
- Ethical, environmental, and legal implications of Cloud technologies, e.g., United Nation's Sustainable Development Goals, GDPR, and MLAT/CLOUD Act.

Skills

Be able to

- design a Cloud-based solution based on a given specification.
- deploy a Cloud-based solution optimized to available resources.
- analyze the security risks of a specific Cloud-based deployment.
- implement applications utilizing cloud APIs on the application layer,
e.g., GitHub, Discord, and the storage/compute layer.
- implement secret management for Cloud applications using,
e.g., Passkey, credentials, passwords, or tokens.

Logistics

Recommended prerequisites

- Git and GitHub proficiency expected
- Operating Systems
- Databases
- Computer Networking
- Web Programming
- Advanced Programming

GitHub

github.com/dat515-2024/info

github.com/dat515-2024/assignments

- Course information
- Syllabus
- Lecture Plan
- Lab project information
- Reading material (links)

QuickFeed

- Submit assignments to GitHub
- Tested by QuickFeed
- View results of lab
- Lab approval

<https://uis.itest.run/>

*Sign up using your
GitHub account*

Enroll in DAT515

*Consider creating a separate
GitHub account for course work*

Discord (Helpbot)

- Announcements
 - (Some on Canvas)
- Get help during labs
- Queue management
- Lab approval

<https://discord.gg/vnp7ceZt4y>

Register with Helpbot using your GitHub username: /register

Weekly Schedule

w34	Monday 19.8	Tuesday 20.8	Wednesday 21.8	Thursday 22.8
12		12:15 DAT515 1-2024 HØST, Cloud Computing Technologies Lab KE E-456 Hein Meling		
13				
14	14:15 DAT515 1-2024 HØST, Cloud Computing Technologies Forelesning KE E-456 Hein Meling	Undervisningsfri Ullandhaug		14:15 DAT515 1-2024 HØST, Cloud Computing Technologies Forelesning KE E-456 Hein Meling
15				
16			16:15 DAT515 1-2024 HØST, Cloud Computing Technologies Lab KE E-456 Hein Meling	
17				

Week	Month	D	Monday	D	Thursday
34	August	19	Introduction to Cloud Computing	22	Virtualization and Containers
35	August	26	Docker	29	Docker
36	September	2	Kubernetes: Container Orchestration	5	Kubernetes: Container Orchestration
37	September	9	(No lecture)	12	DevOps: Best Practices
38	September	16	Guest Lectures	19	Cloud APIs and Microservices Concepts
39	September	23	Serverless Computing	26	(No lecture)

“I love deadlines. I like the whooshing sound they make as they fly by.”

— Douglas Adams, *The Salmon of Doubt*

Lab	Topic	Grading	Approval	Submission	Deadline
1	<u>Getting Started with Docker: Deploying a Basic Web Application</u>	Pass/fail	TA	Individually	August 27
2	<u>Getting Started with Kubernetes: Building and Deploying</u>	Pass/fail	TA	Individually	September 10
3	<u>Getting Started with AWS</u>	Pass/fail	TA	Individually	September 24
4	<u>Building Apps with Service Weaver</u>	Pass/fail	TA	Individually	September 24
5	<u>Course Project</u>	Pass/fail	TA	Group	November 7
6	<u>Project Presentation</u>	Pass/fail	TA	Group	November 14

Course Project

(Lab 5 and Lab 6)

<https://github.com/dat515-2024/assignments/tree/main/lab5>

- Start planning the project this week (read the project description)
- Prepare design doc
 - Overview
 - Architecture
 - Technologies
 - Deployment

<https://github.com/dat515-2024/assignments/tree/main/lab5>

- Will be created by QuickFeed
- When you create a group
- More details about groups and the commit policy soon
- Will be named like this:
 - <https://github.com/dat515-2024/group-name>

<https://github.com/dat515-2024/assignments/tree/main/lab5>

- Introduction
- Design
- Implementation
- Deployment
- Conclusion
- References
- Appendices

<https://github.com/dat515-2024/assignments/tree/main/lab5>

- Be **no longer than 6 pages**, excluding references and appendices.
- Use font size 10, **double column format**, line spacing 1, and standard margins.
- Be properly formatted, well-organized, clear, and concise.
- Be written in a professional and academic style, with proper references, grammar, and spelling.
- **Screenshots of code is not allowed**; instead, code should be formatted or linked to the repository.
- Screenshots are only allowed to showcase your application's user interface, etc.

<https://github.com/dat515-2024/assignments/tree/main/lab6>

- Instructions to be added later

Course policies

Group for Project

[dat515-2024/info/policy.md](#)

- Find group members first
 - Groups of two (max three)
 - Use the #group-maker channel on Discord
- Agree on a name
- Sign up for a group on QuickFeed

Partner Problems

[dat515-2024/info/policy.md](#)

- Please **tell us** and **your partner** early
 - decide to drop out of the course
 - want to dissolve the group and work alone
- To avoid problems, we suggest to organize regular
 - joint work sessions



Deadlines and Slip days

[dat515-2024/info/policy.md](#)

- If you cannot make a deadline, you can use up to **7 slip days** throughout the semester, including weekends and holidays.
- It is the **date on your lab's submission**, as viewed **on GitHub**, that counts towards the slip days.
- Slip days **cannot be used to extend** the project and presentation deadlines.
 - Project and presentation deadlines are hard.

Collaboration

[dat515-2024/info/policy.md](#)

- Talk to each other or the teaching staff about any of the assignments. Assistance is limited to general discussion of the problem. Each student/group must write out their own solutions.
- Consulting another student's/group's solution is prohibited. Submitted solutions must not be copied from any source.
- You **shall not** supply your work to other students in future instances of this course.



Generative Models

[dat515-2024/info/policy.md](#)

- May use generative models such as ChatGPT or GitHub Copilot to generate code.
- Must be able to explain the code as if you had written it yourself.
- The code must obviously **solve the assignment** and pass sufficient number of tests on QuickFeed.



Contributions

[dat515-2024/info/policy.md](#)

- Group members must **contribute equally** to code
- *Implies that group members should commit a similar amount of code on GitHub.*
- Each group member must **individually present** and explain their work to a TA for approval.
 - Group members expected to be able to explain the code written by other group members.



Two Working Styles

[dat515-2024/info/policy.md](#)

- Pair programming
 - Alternate between coding vs watching
 - Make sure to switch user that writes and commits code to GitHub
- Code Review
 - Each group member creates a branch
 - Create Pull Request
 - Partner reviews your code before it is merged into the main branch

When to Commit

[dat515-2024/info/policy.md](#)

- Commits should represent a **logical unit of work**.
 - A bug fix, a new feature, a refactoring, etc.
 - It is up to you to decide what constitutes a unit of work, but it should not be too large.
- *Unacceptable to commit an entire lab assignment in one commit.*

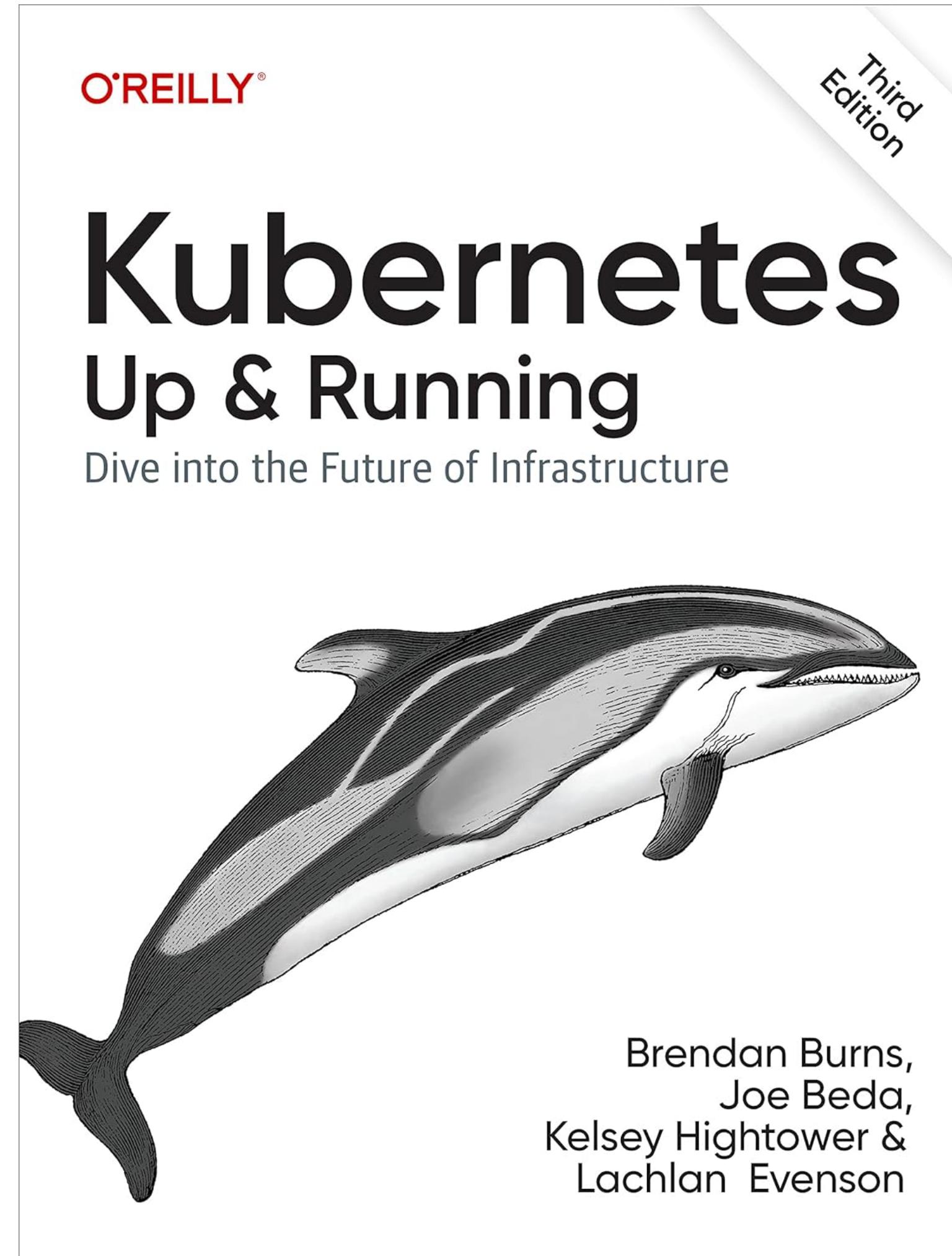
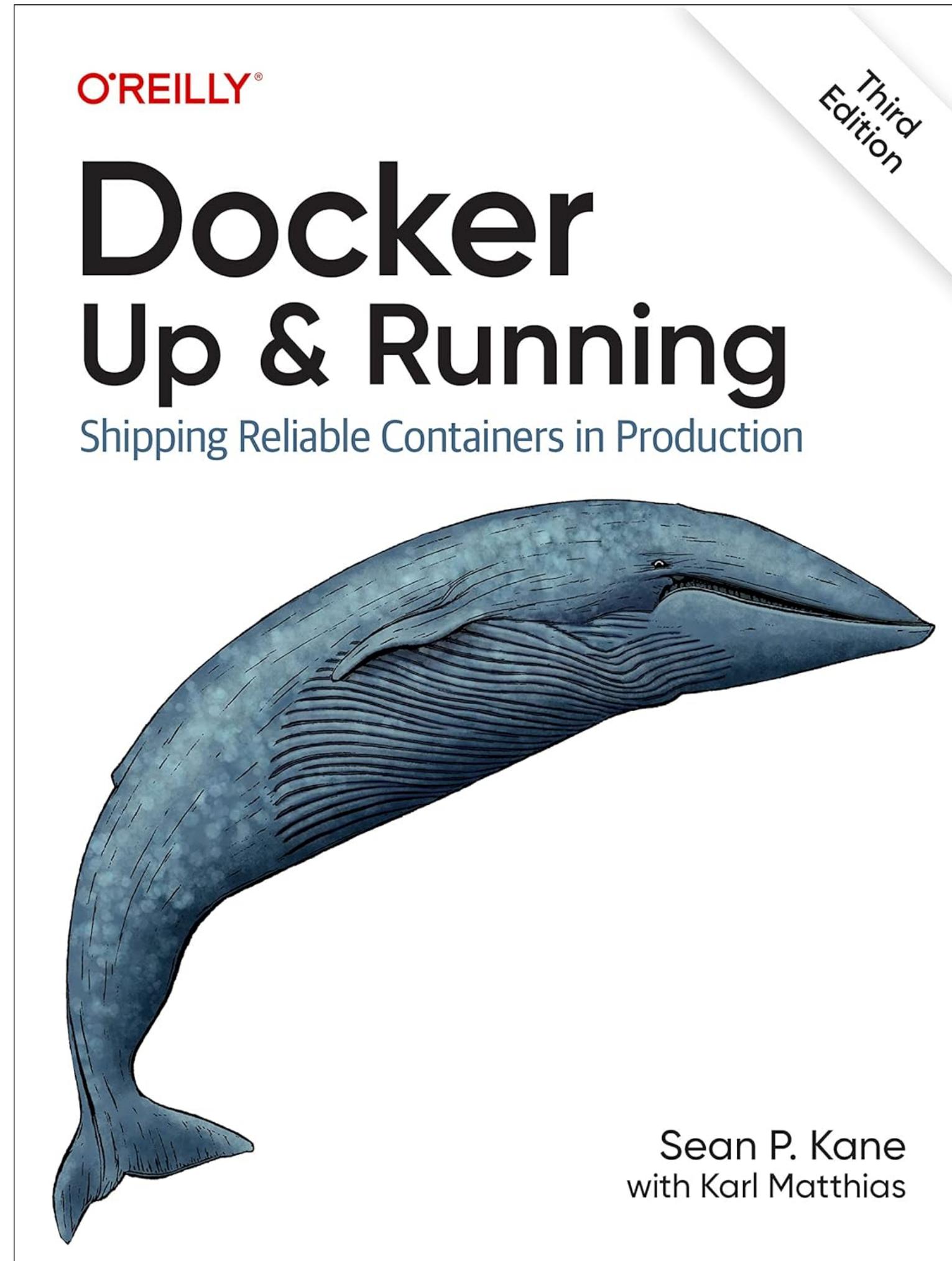
Commit Messages

[dat515-2024/info/policy.md](#)

- Describe the changes you have made.
- Must follow the Conventional Commits format.

```
93fa221 Hein Meling Mon Jan 8 13:53:59 2024 +0100 fix(lab1): fixed doc format issue in errors task
90509b3 Hein Meling Mon Jan 8 13:51:13 2024 +0100 feat(lab1): implemented fibonacci sequence task
43a353c Hein Meling Mon Jan 8 13:49:57 2024 +0100 feat(lab1): implemented multiwriter task
9d5d0bf Hein Meling Mon Jan 8 13:49:39 2024 +0100 feat(lab1): implemented errors task
2693212 Hein Meling Mon Jan 8 13:44:50 2024 +0100 feat(lab1): implemented the cipher task
f6d8a83 Hein Meling Mon Jan 8 13:43:12 2024 +0100 feat(lab1): implemented stringer task
1f9a64a Hein Meling Mon Jan 8 13:40:55 2024 +0100 mc-go(lab1): answered go questions
88da89b Hein Meling Mon Jan 8 13:38:25 2024 +0100 token(lab1): ran generate_token on go1
23e522c Hein Meling Mon Jan 8 13:20:26 2024 +0100 mc-missing(lab1): answered missing semester questions
24cf9ac Hein Meling Mon Jan 8 13:19:01 2024 +0100 mc-shell(lab1): answered shell questions
31c4fa8 Hein Meling Mon Jan 8 13:17:54 2024 +0100 mc-git(lab1): fixed git answers
ce0027e Hein Meling Mon Jan 8 11:19:53 2024 +0100 mc-git(lab1): answered git questions
8133f91 Hein Meling Mon Jan 8 09:52:23 2024 +0100 lab1: first commit
```

Study Resources



- Not mandatory reading, but they are nice if you like reading books.
- Online resources are sufficient to cover the course's topics.
- Links are included in slides and assignments.

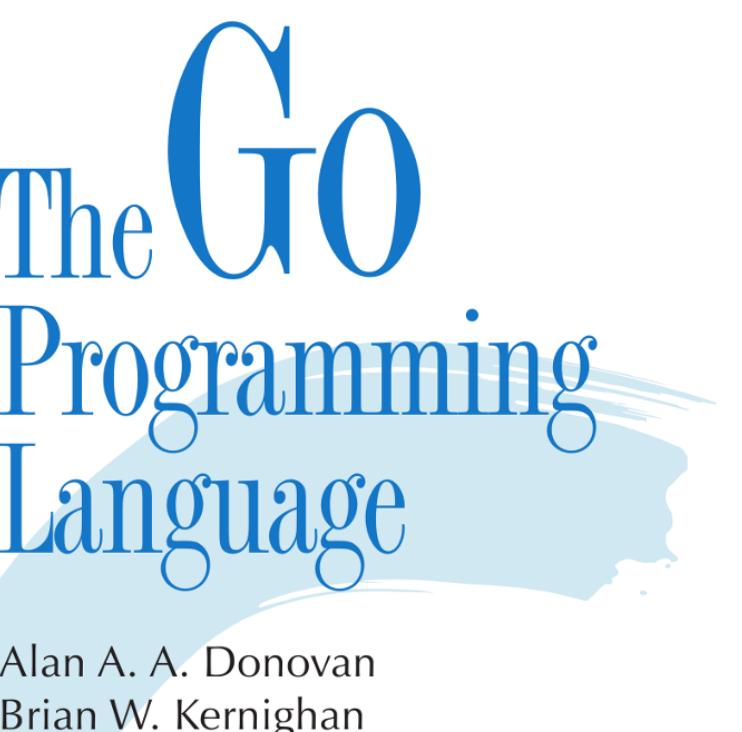
Important Online Resources

- <https://docs.docker.com/>
- <https://kubernetes.io/docs/home/>
- AWS Cloud Foundation Course (requires login; see lab3)
- <https://serviceweaver.dev/>

- Some overview and research papers
- Lab assignments and project
- Lecture slides

Links

- Official web page: go.dev
- [A Tour of Go](#)
- [Effective Go](#)
- Other [Learning resources](#)
- [Standard library](#) and [Style guide](#)
- Video: [Go Tutorial](#) from freeCodeCamp.org (7 hours)
 - Beware: some information may be outdated!!



Alan A. A. Donovan
Brian W. Kernighan



We want your feedback
on the course

Questions?