

Dependable Distributed Systems

Ashkan Ansarifard

Start date: February 11, 2023
Last modified: February 16, 2023

Contents

| | | |
|-----------|---|-----------|
| 1 | Modelling Distributed Systems | 4 |
| 2 | Time in Distributed Systems | 5 |
| 3 | Logical Clock | 6 |
| 4 | Distributed Mutual Exclusion | 7 |
| 5 | Failure Detection Abstraction | 8 |
| 5.1 | Perfect Failure Detectors | 8 |
| 5.2 | Eventually Perfect Failure Detectors | 8 |
| 5.3 | Leader Election | 8 |
| 5.4 | Eventual Leader Election | 8 |
| 6 | Broadcast Communications | 9 |
| 7 | Consensus | 10 |
| 8 | Ordered Communications | 11 |
| 9 | Registers | 12 |
| 10 | Software Replication | 13 |
| 11 | Overview on Capacity Planning | 14 |
| 12 | Modeling the Workload of a System | 15 |
| 13 | Building a Performance Model 1 | 16 |
| 14 | Building a Performance Model 2 | 17 |
| 15 | Dependability Evaluation | 18 |
| 16 | Intro to Experimental Design | 19 |
| 17 | CAP Theorem | 20 |
| 18 | Consistency Criteria for Distributed Shared Memories | 21 |
| 19 | Publish-Subscribe Communication Paradigm | 22 |
| 20 | Overlay Networks | 23 |
| 21 | DLT and Blockchain | 24 |
| 22 | Exercises | 25 |

23 Exams

26

1 Modelling Distributed Systems

A distributed system is a set of entities, computes or machines communicating, coordinating and sharing resources to reach a common goal, appearing as a single computing system.

2 Time in Distributed Systems

3 Logical Clock

4 Distributed Mutual Exclusion

5 Failure Detection Abstraction

A **failure detector abstraction** is a software module used to detect faulty processes, it encapsulate timing assumptions of a either partially synchronous or fully synchronous system. It has two properties:

- **Accuracy:** that represents the ability to avoid mistakes
- **Completeness:** that represents the ability to detect all failures

5.1 Perfect Failure Detectors

Here is a figure that illustrates the specifications of perfect failure detectors:

5.2 Eventually Perfect Failure Detectors

5.3 Leader Election

5.4 Eventual Leader Election

6 Broadcast Communications

7 Consensus

8 Ordered Communications

9 Registers

10 Software Replication

11 Overview on Capacity Planning

12 Modeling the Workload of a System

13 Building a Performance Model 1

14 Building a Performance Model 2

15 Dependability Evaluation

16 Intro to Experimental Design

17 CAP Theorem

18 Consistency Criteria for Distributed Shared Memories

19 Publish-Subscribe Communication Paradigm

20 Overlay Networks

21 DLT and Blockchain

22 Exercises

Notice: The exercises are from 2022-2023 academic year.

23 Exams