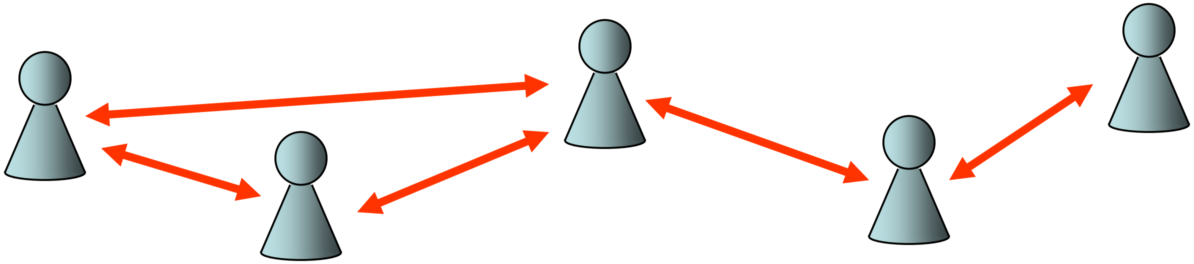
Advanced Distributed Algorithms and Data Structures



Christian Scheideler

Institut für Informatik

Universität Paderborn

Advanced Distributed Algorithms and Data Structures

Lecture: Mon 11:00-14:00, F0.530
Tutorial: Mon 09:00-11:00, F0.530 (starts 2nd week)

Website:
see http://cs.uni-paderborn.de/ti/lehre/veranstaltungen/ss-2019

Focus Areas and Grading:

- Focus areas “Algorithm Design” and “Networks and Communication”
- Prerequisites for oral exam: presentation of solution to homework problem and software project
- Grading: oral exam (recommended: by end of September)

Prerequisites:

- basic knowledge in algorithms and data structures
- recommended: distributed algorithms and data structures course

Advanced Distributed Algorithms and Data Structures

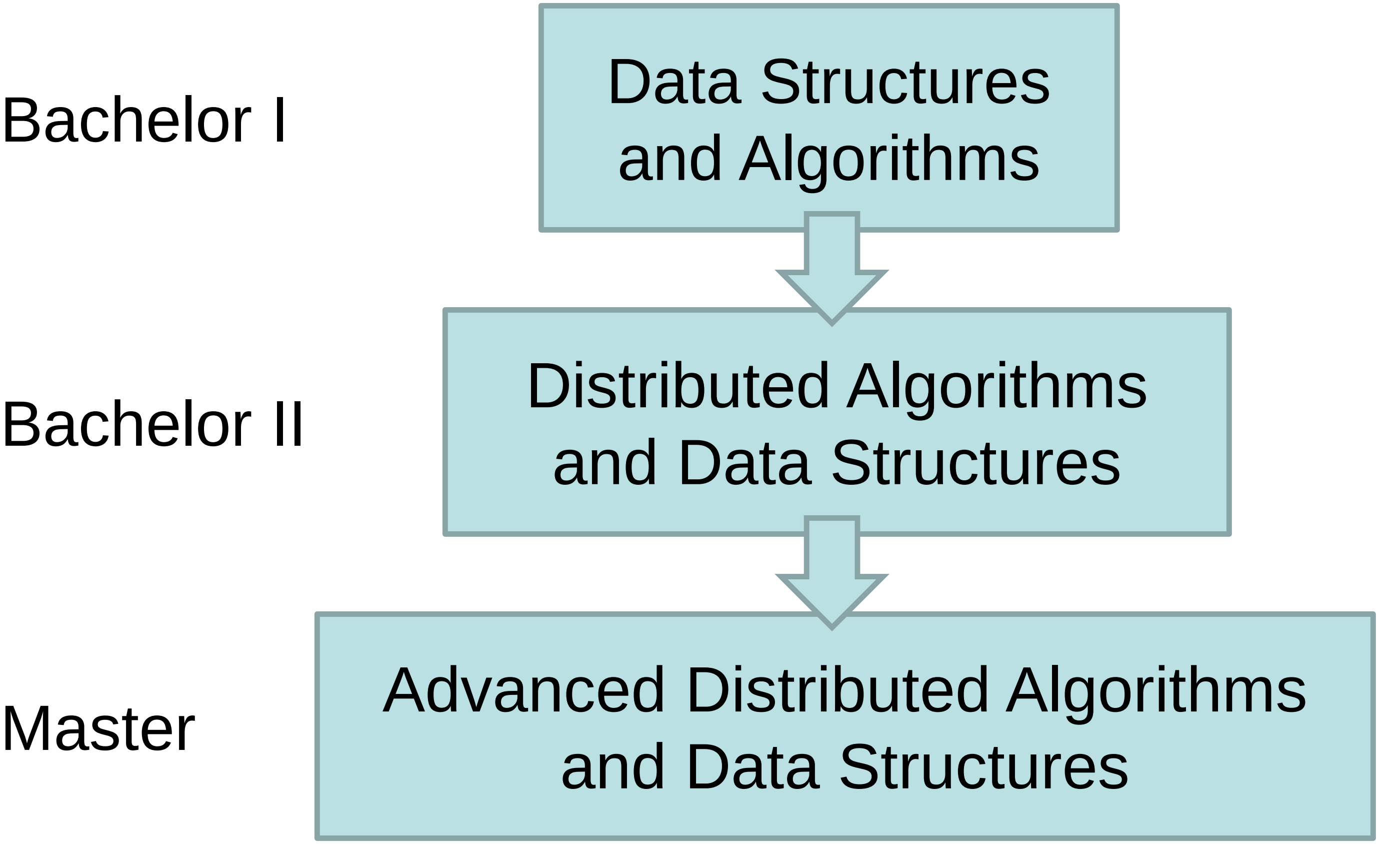
Homework assignments:

- Weekly assignments each Monday on the website (starting with this week)
- Theoretical and practical problems

Slides and assignments: course website

Book recommendations: no book available (lecture is based on newest results)

Embedding into CS Curriculum



Bachelor I

Data Structures and Algorithms

Bachelor II

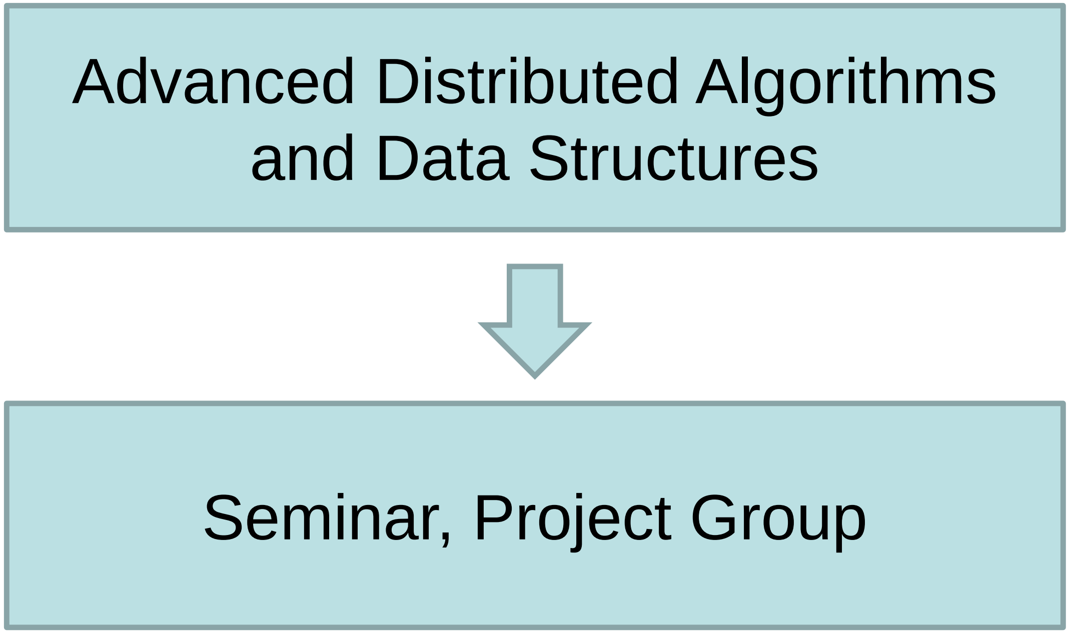
Distributed Algorithms and Data Structures

Master

Advanced Distributed Algorithms and Data Structures

Embedding into CS Curriculum

Master



Advanced Distributed Algorithms and Data Structures

Seminar, Project Group

Advanced Distributed Algorithms and Data Structures

Goals:

1. Introduction to advanced concepts in distributed algorithms and data structures.
2. Introduction to important design methods.
3. Introduction to important analytical methods.

Introduction

Sequential Algorithms
and Data Structures

Distributed Algorithms
and Data Structures

Introduction

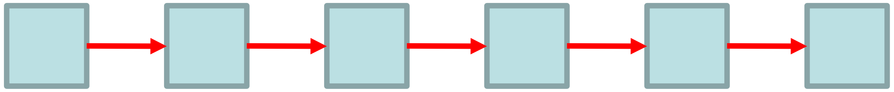
What are the basic problems for distributed algorithms and data structures?

Introduction

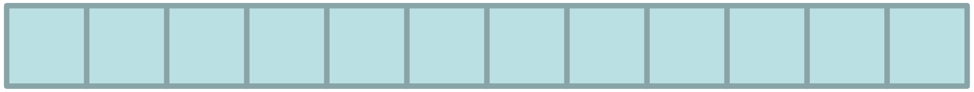
Definition 1.1: A *data structure* is a certain way to organize data in a computer so that operations like, for example, *search, insert,* and *delete* are simple and effective to realize.

Simple examples:

- Lists

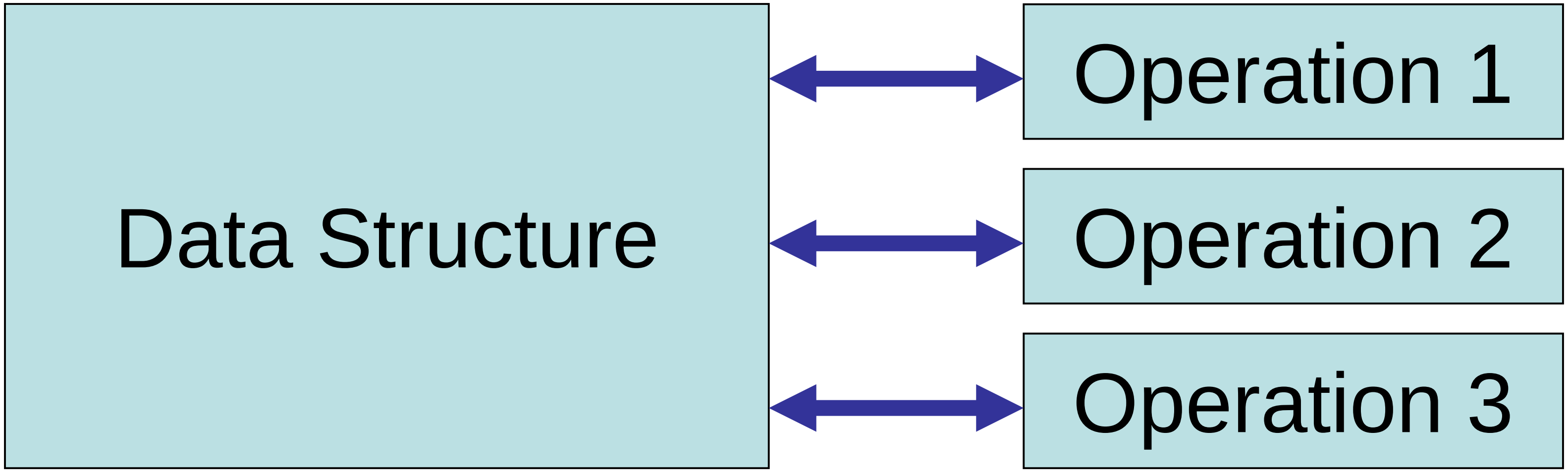


- Arrays



Introduction

Basic view:



Data Structure

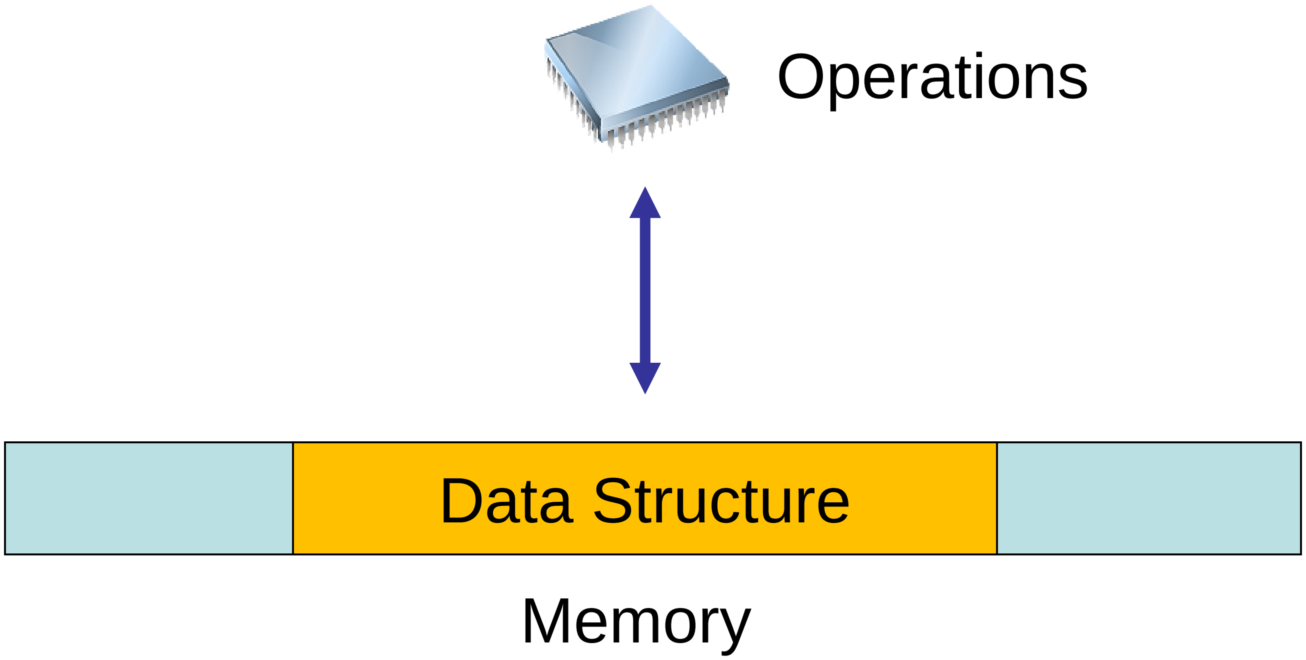
Operation 1

Operation 2

Operation 3

Introduction

Classical case: computer with one processor



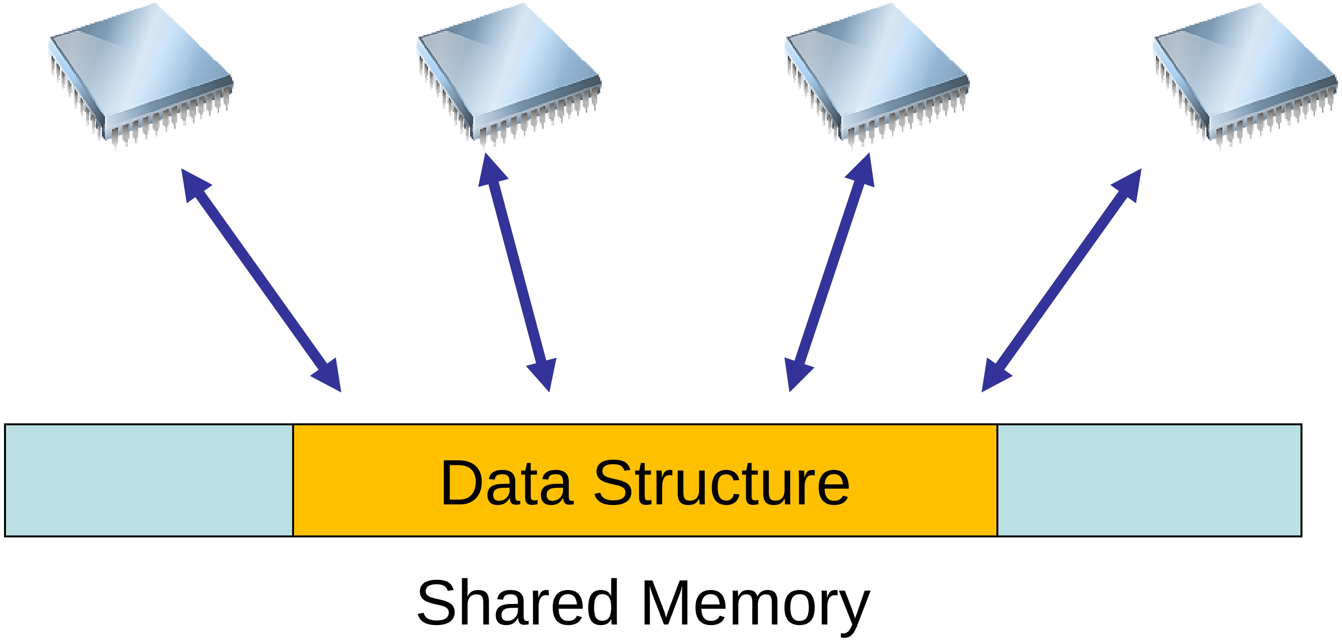
Operations

Data Structure

Memory

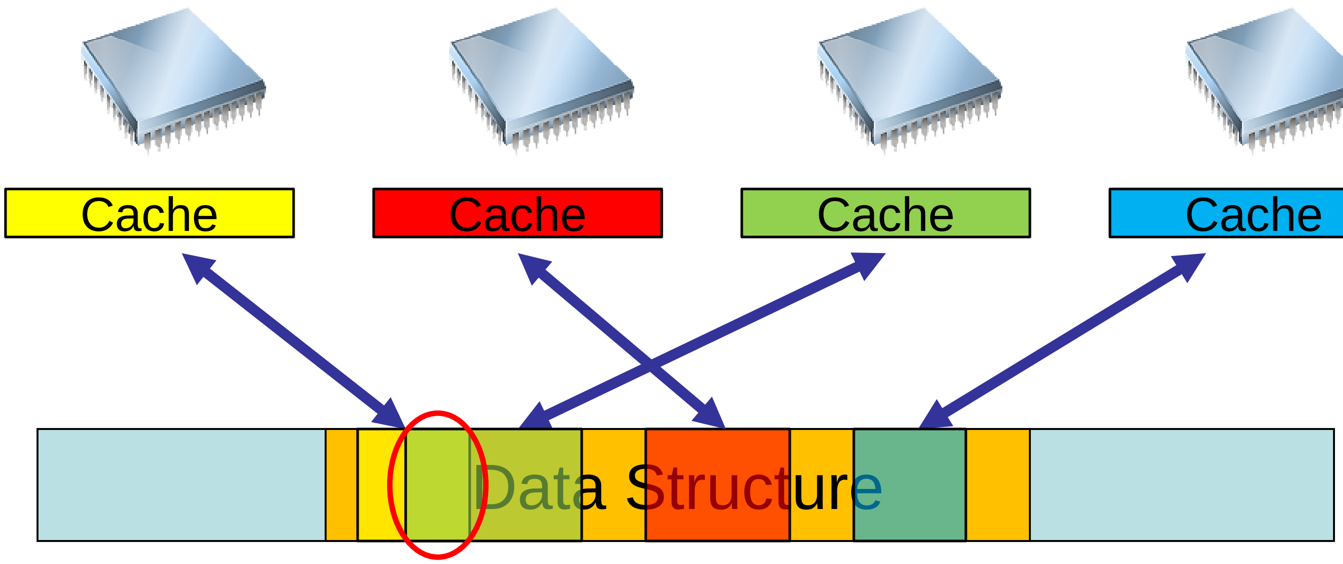
Introduction

Computer with several processors/cores:



Introduction

Computer with several processors/cores:

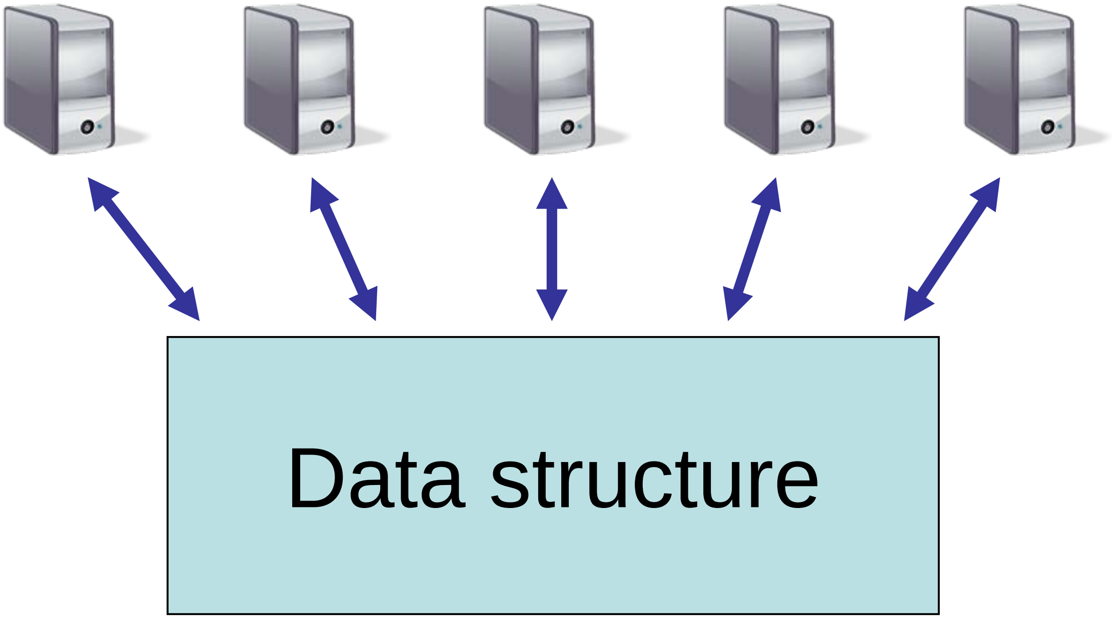


Overlaps:

- access conflicts (correctness)
- performance problems (efficiency)

Introduction

Multiple computers:

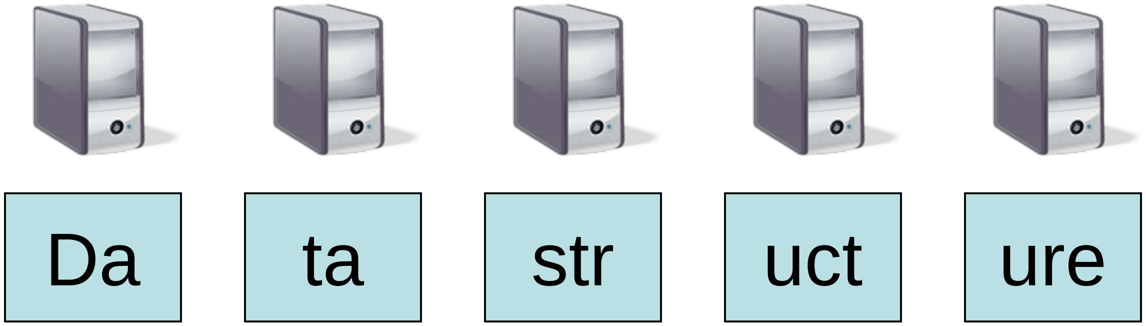


Data structure

Problem: distribution of DS among computers

Introduction

Multiple computers:



Da

ta

str

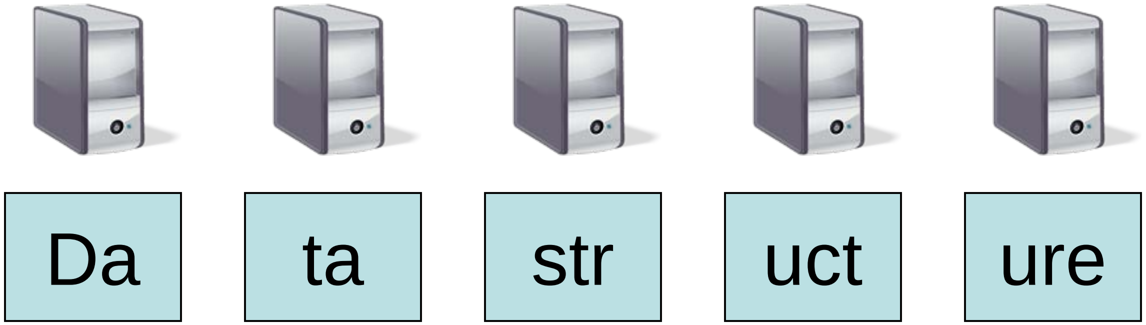
uct

ure

Problem: distribution of DS among computers

Introduction

Multiple computers:

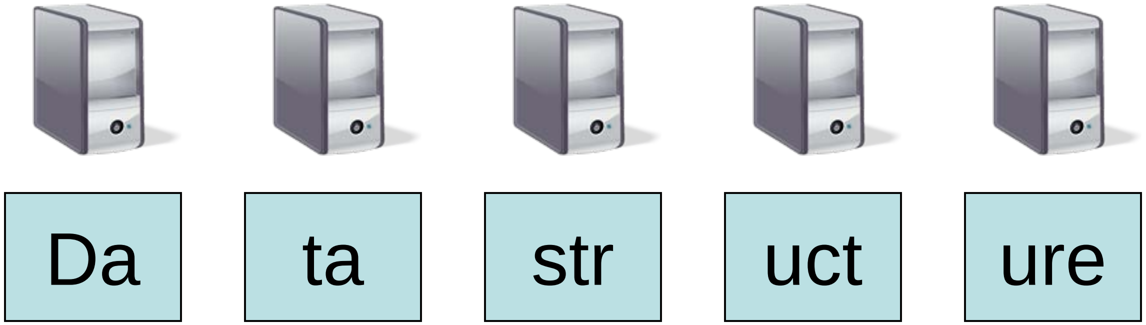


Basic problems:

- How to interconnect the computers?
- How to coordinate the management of the DS among the computers?

Introduction

Multiple computers:



Da

ta

str

uct

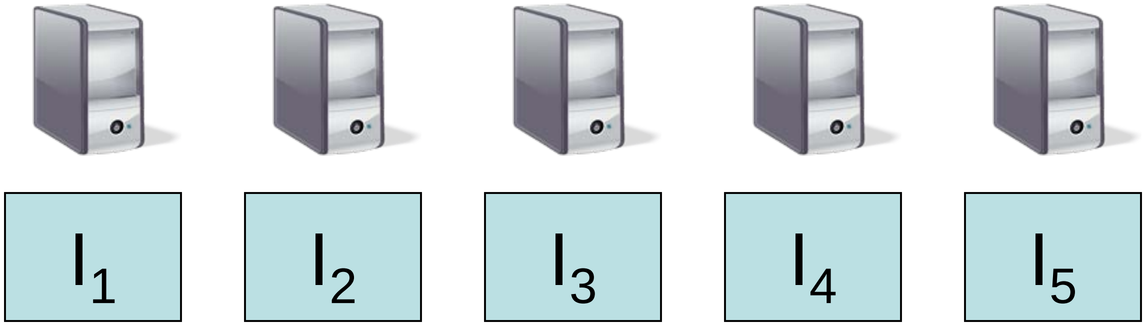
ure

How to manage the DS?

- We need redundancy to cope with failures.
- But then we need to maintain consistency!

Introduction

Distributed Algorithms:



Input I might be split into different pieces I_j that are distributed among many computers.

How to efficiently solve problems (minimum spanning tree, shortest paths,..) in this case?

Introduction

General problem:
Find solutions that are scalable, robust and secure (because participants might be faulty or adversarial, or might get attacked from outside!)

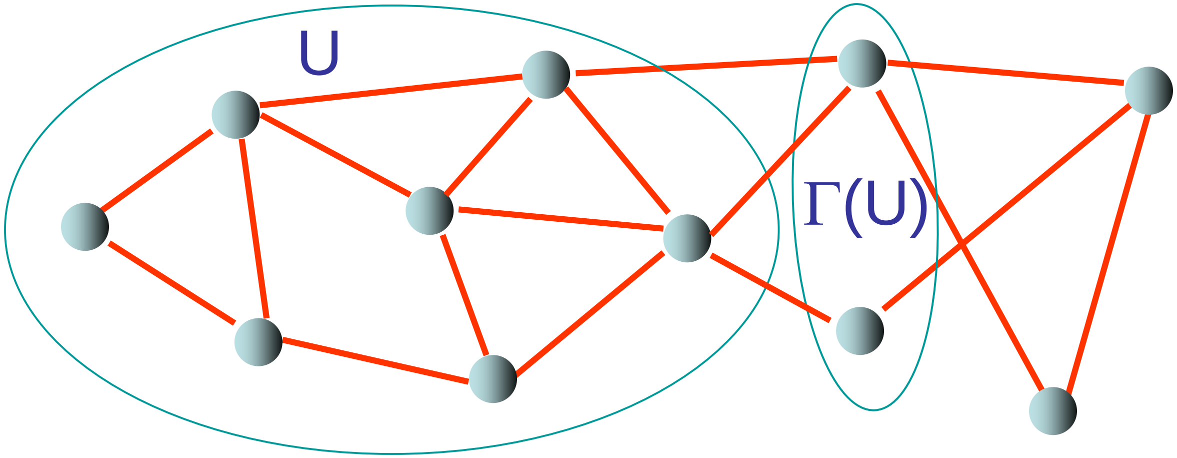
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Information Aggregation
8. Distributed Scheduling
9. Distributed Optimization

Foundations

Graphs and graph parameters, processes, …



Advanced distributed algorithms and data structures

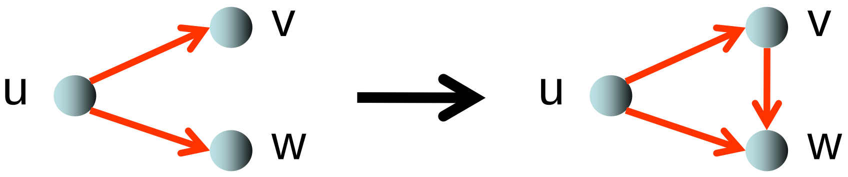
Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

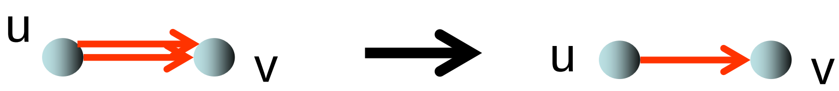
Link Primitives

Admissible link primitives w.r.t. connectivity:

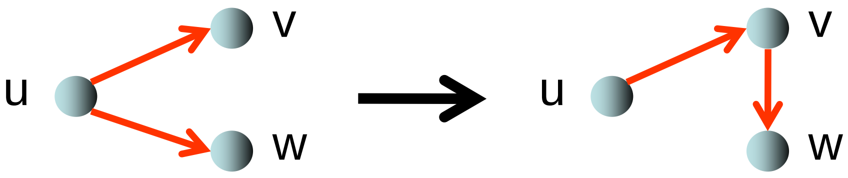
Introduction



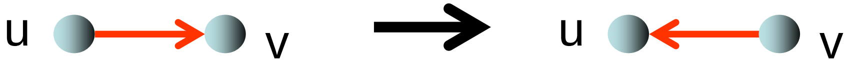
Fusion



Delegation



Reversal



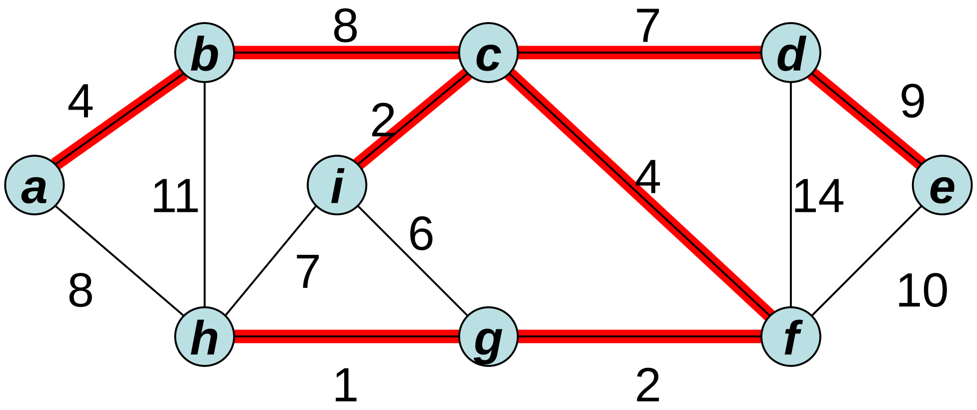
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

Networks

Minimum spanning tree:



New approach: hybrid networks

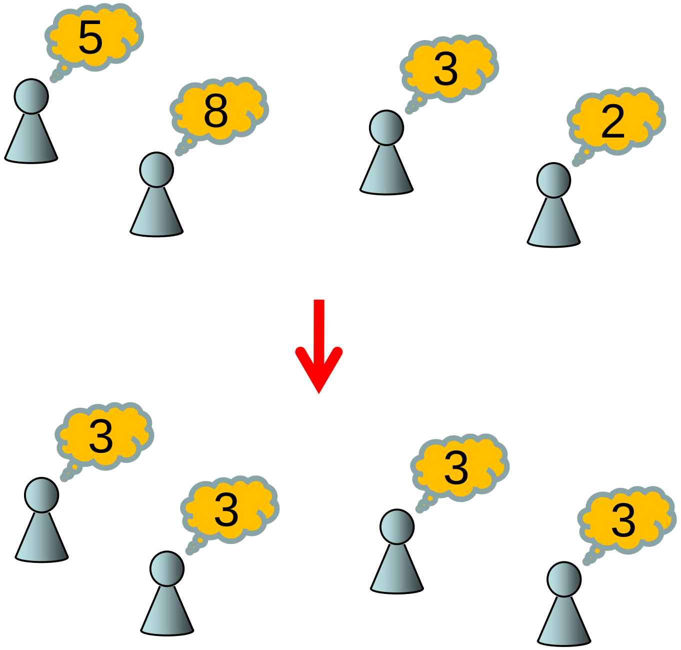
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

Consensus and Blockchains

Consensus:

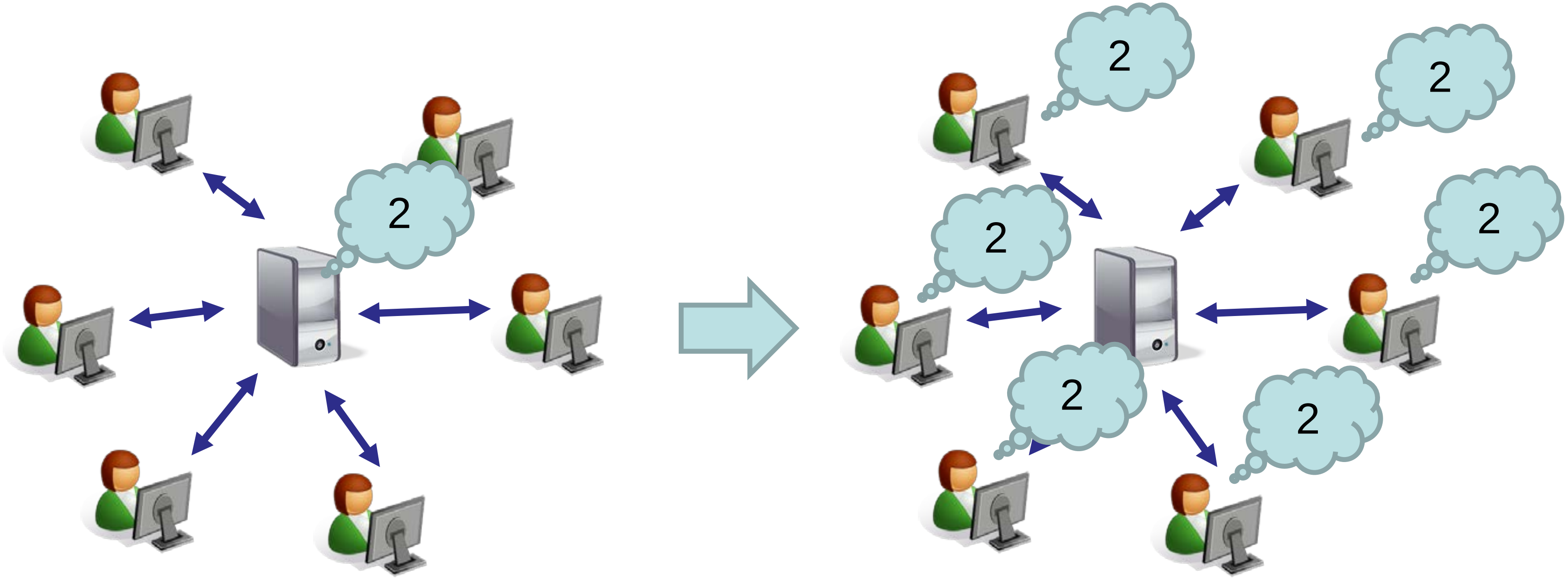


Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

Information Dissemination



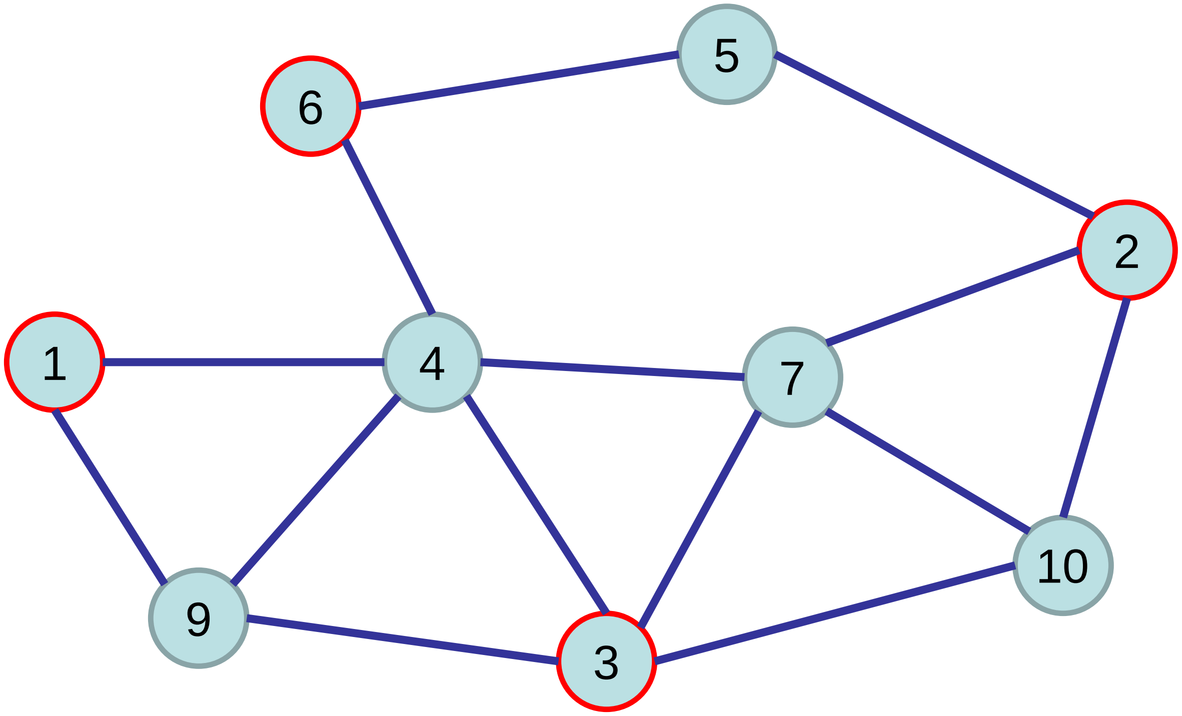
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

Distributed Scheduling

Independent set, matching, coloring,…



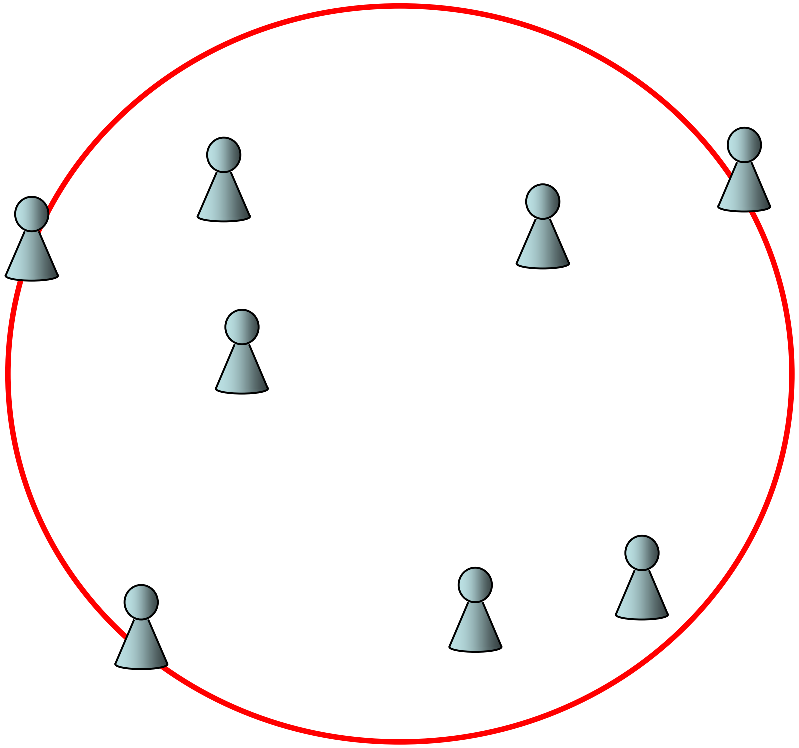
Advanced distributed algorithms and data structures

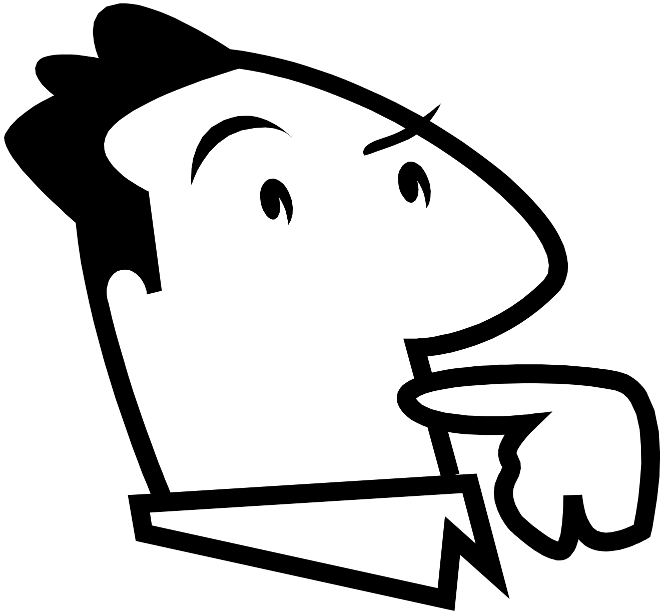
Contents:

1. Introduction
2. Foundations
3. Link primitives
4. Networks
5. Consensus and Blockchains
6. Information Dissemination
7. Distributed Scheduling
8. Distributed Optimization

Distributed Optimization

Smallest enclosing ball problem:





Questions?