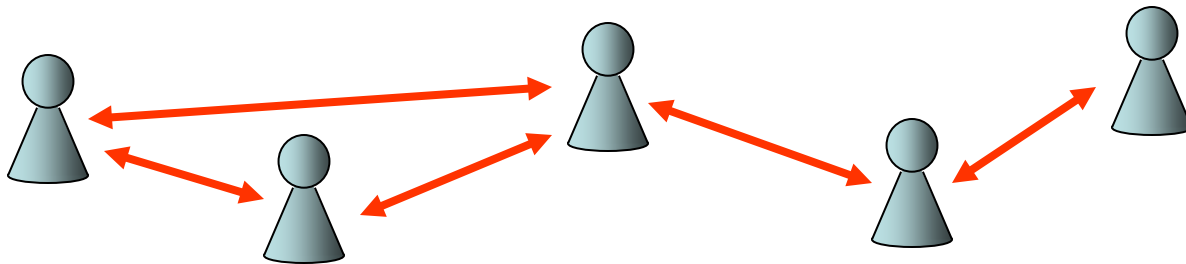


Advanced Distributed Algorithms and Data Structures



Christian Scheideler
Institut für Informatik
Universität Paderborn

Advanced Distributed Algorithms and Data Structures

Lecture: Mon 13:15-15:45, F1.110

Tutorial: Mon 09:15-10:45, F2.211 (starts 2nd week)

Website:

<https://cs.upb.de/ti/lehre/ws-2026-2027/>

Focus Areas and Grading:

- Focus areas “Classical and Quantum Algorithm Design” and “Computer and Communication Systems” and “Security”
- Prerequisites for oral exam: presentation of two solutions to homework problems
- Grading: oral exam

Prerequisites:

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

Advanced Distributed Algorithms and Data Structures

Deadlines:

- Registration deadline for course: Oct 31
 - Registration for course achievement: Oct 22 – Nov 26
 - (Course achievement: present two solutions)
 - Deregistration deadline: Dec 31
 - Registration for 1st exam: Oct 22 – Nov 26
 - Registration for 2nd exam: Mar 1 – Mar 12
 - Deregistration from exam: at least 2 days before exam
- All (de)registrations are done in PAUL.

General rule: If you see anything in PAUL that you can register for, you should do so.

Advanced Distributed Algorithms and Data Structures

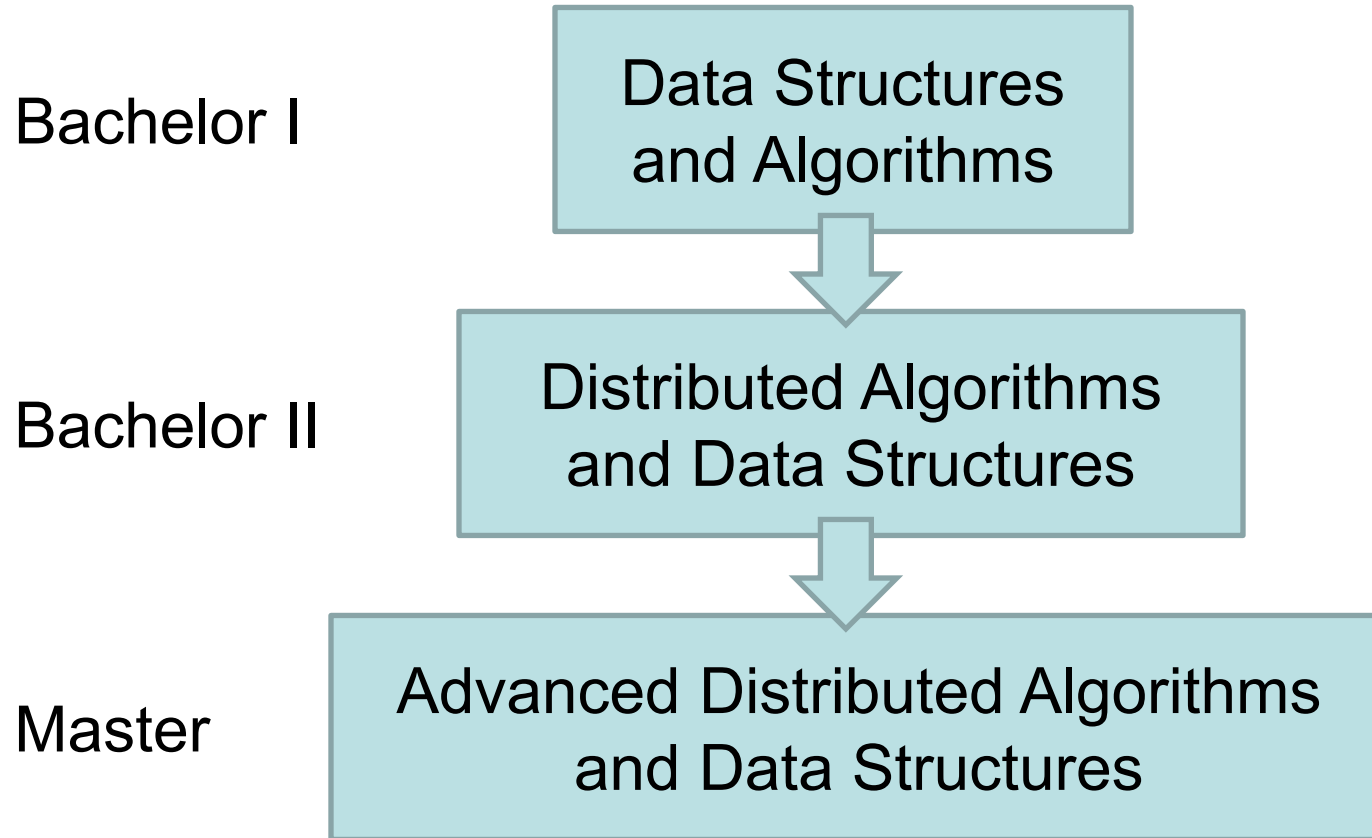
Homework assignments:

- Weekly assignments each Monday in PANDA (starting with this week)
- Theoretical and practical problems
- Homework solutions will be presented / discussed in the tutorial one week later

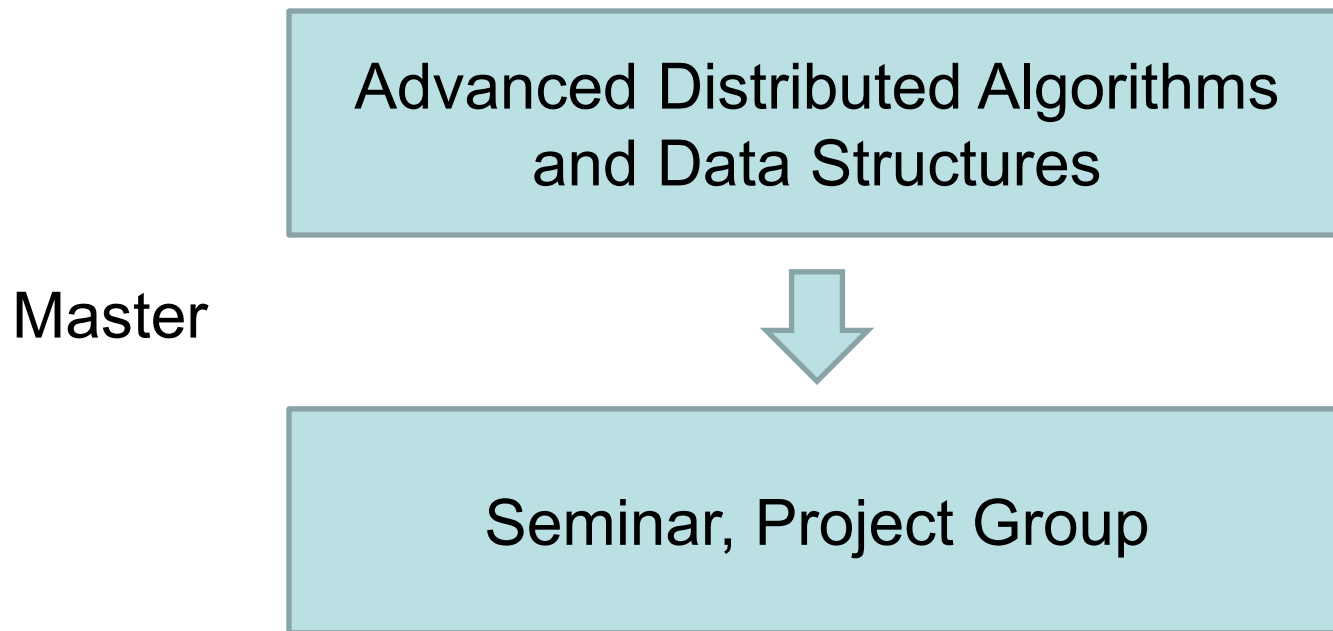
Slides and assignments: PANDA

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

Embedding into CS Curriculum



Embedding into CS Curriculum



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

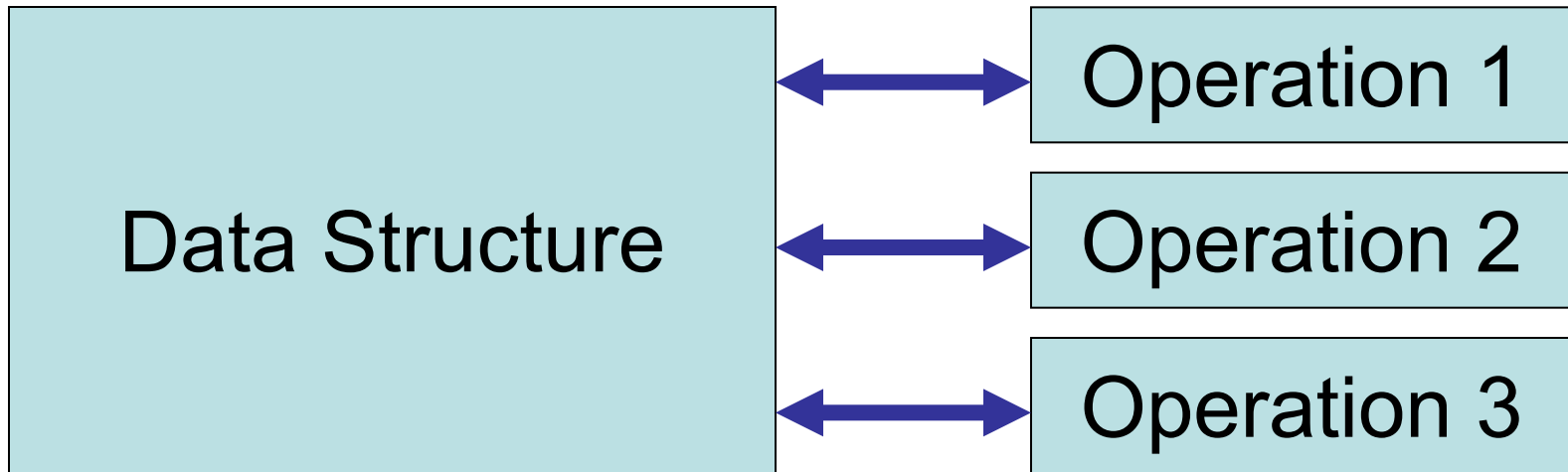


- Arrays



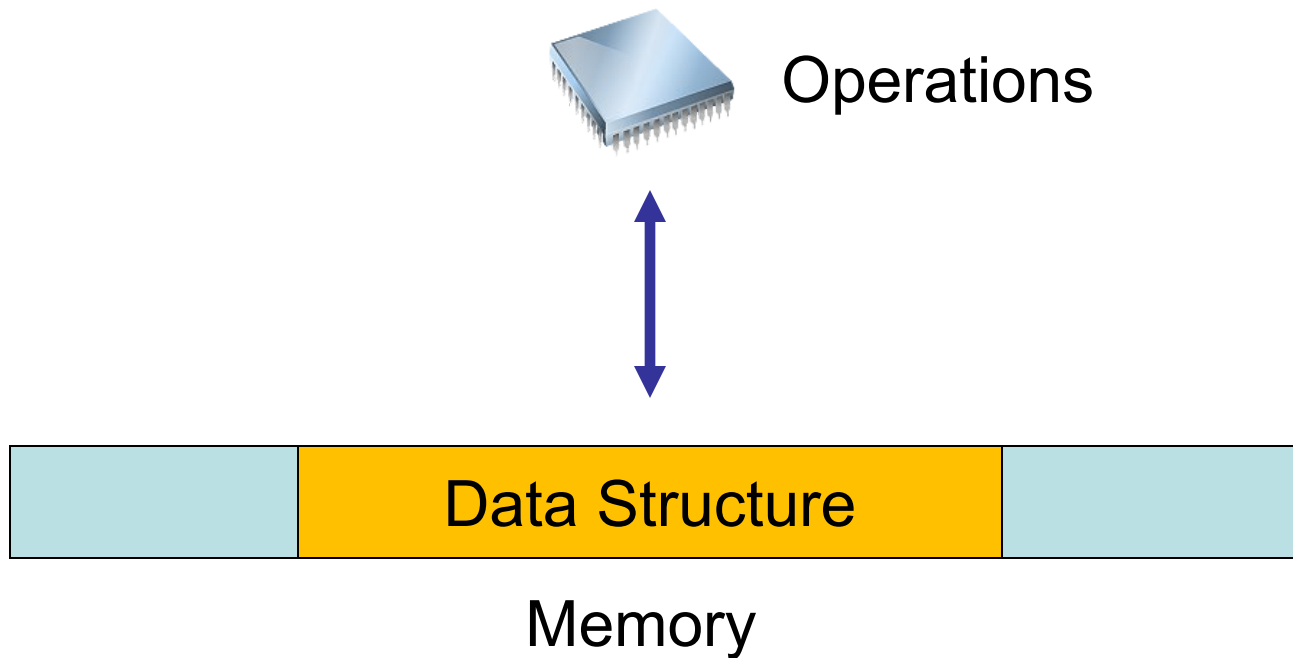
Introduction

Basic view:



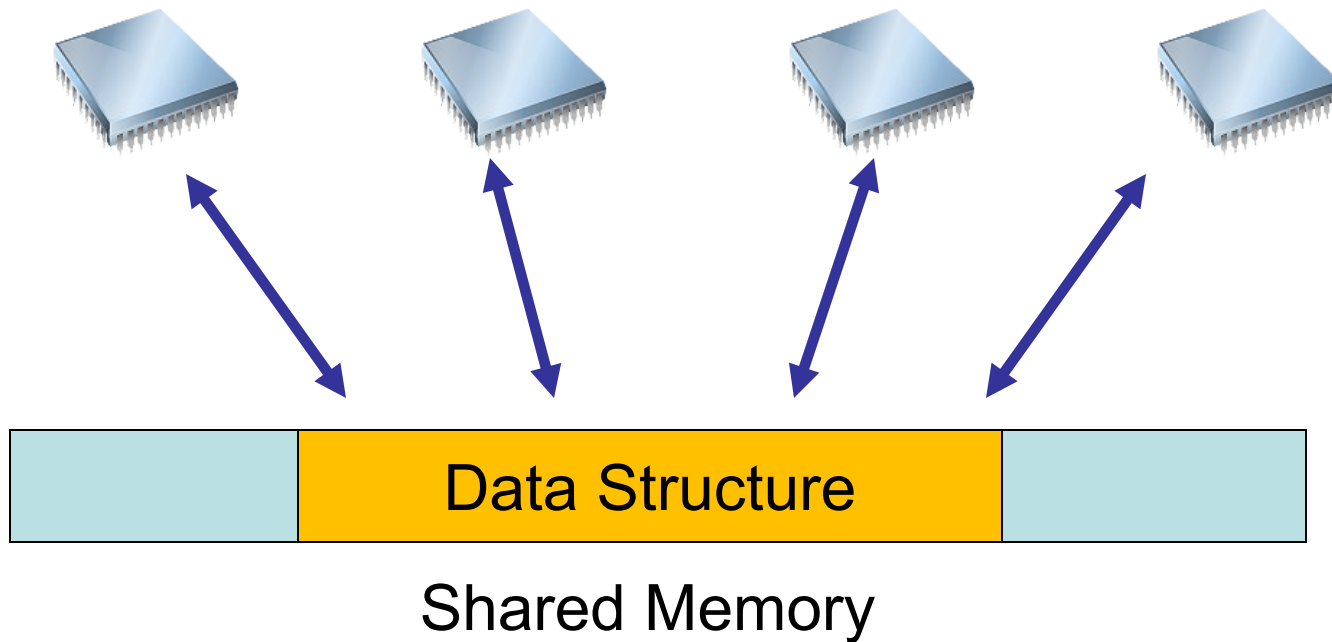
Introduction

Classical case: computer with one processor



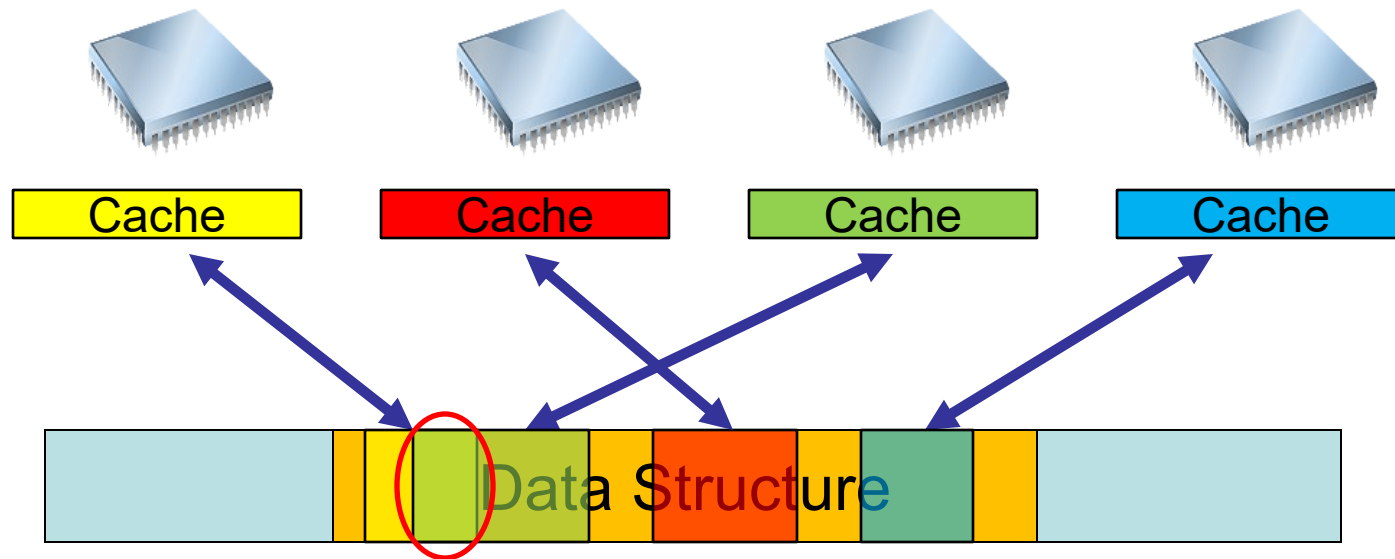
Introduction

Computer with several processors/cores:



Introduction

Computer with several processors/cores:

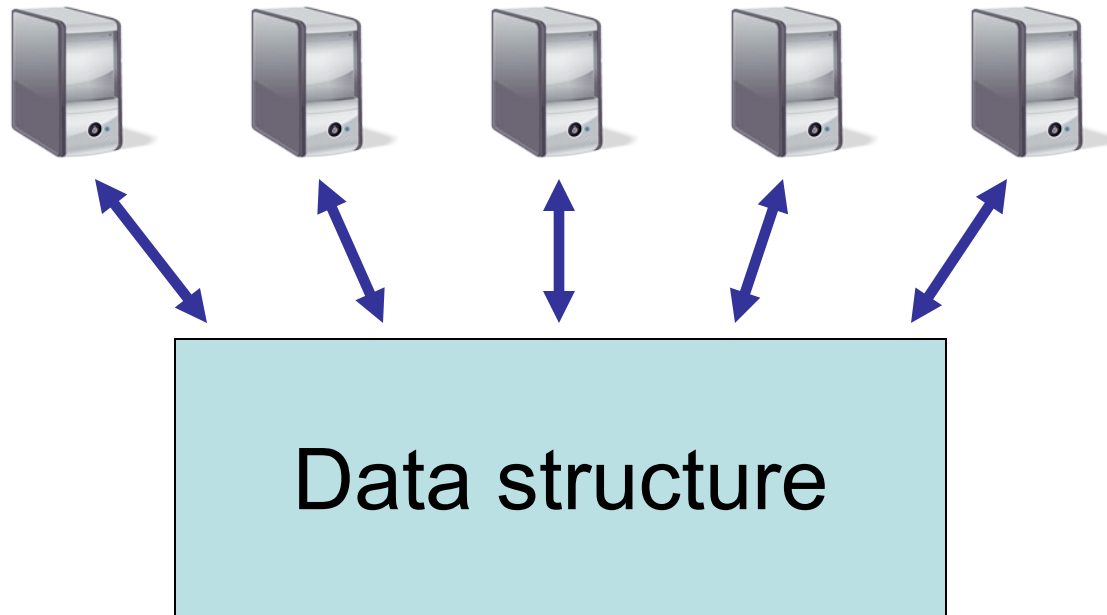


Overlaps:

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

Introduction

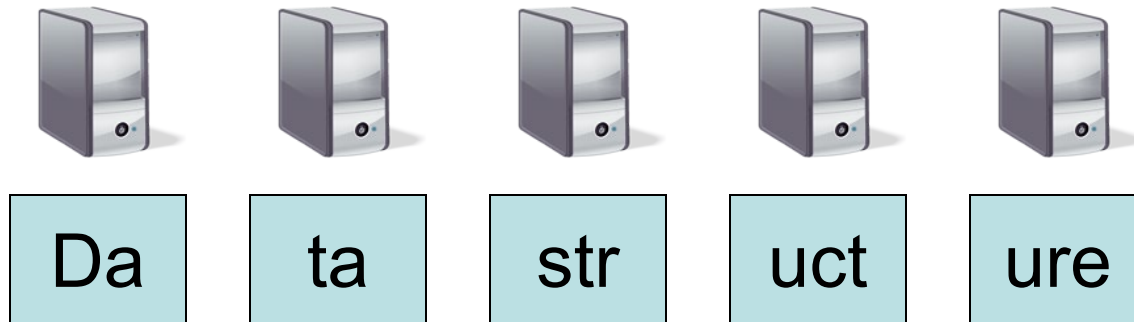
Multiple computers:



Problem: distribution of DS among computers

Introduction

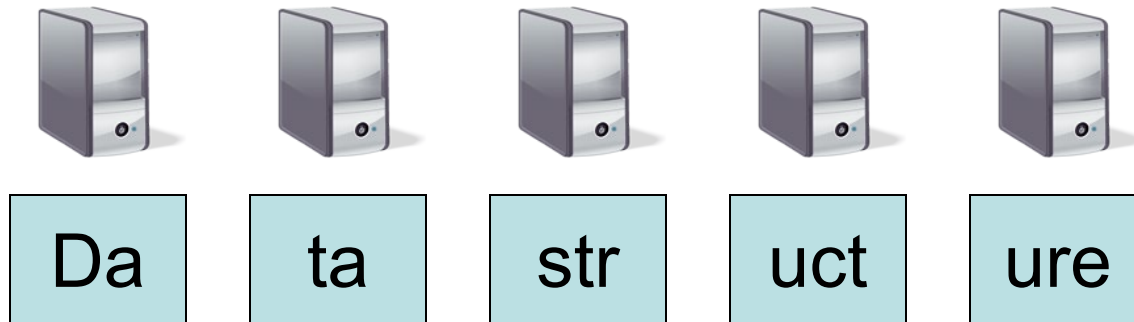
Multiple computers:



Problem: distribution of DS among computers

Introduction

Multiple computers:

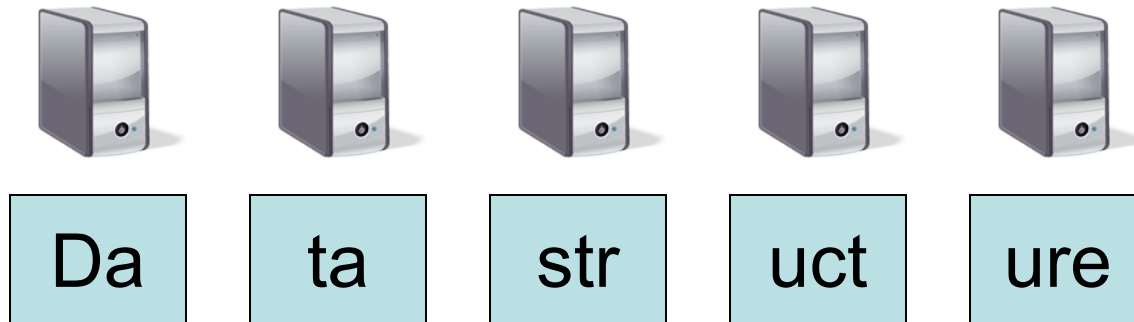


Basic problems:

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

Introduction

Multiple computers:

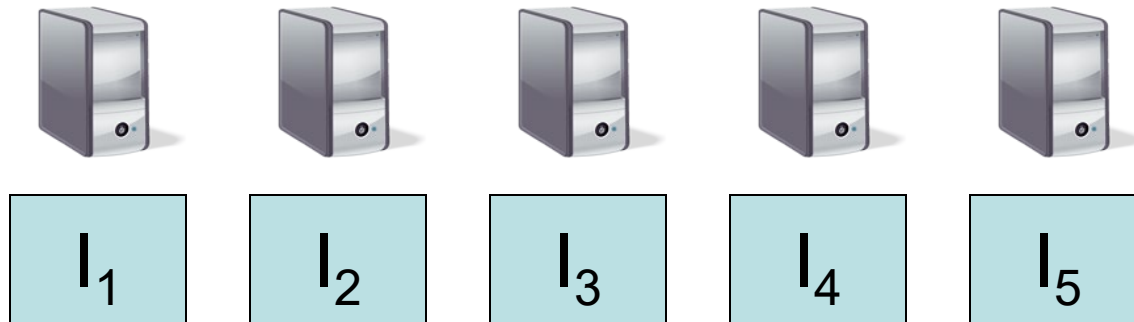


How to manage the DS?

- 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!)

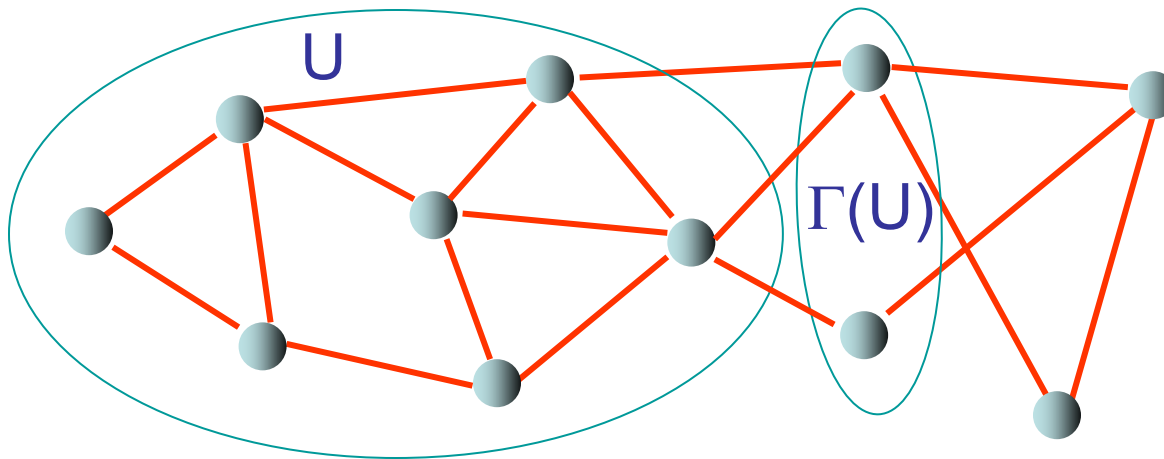
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Foundations

Graph theory, probability theory, process model



Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Access Control

Important access control requirements:

- **Integrity**: It should not be possible to construct, tamper with or steal an access right.
- **Propagation**: There should be mechanisms for controlling the transfer of access rights.
- **Revocation**: It should be possible to revoke an access right.



Access control problems are usually ignored in theory of distributed computing.

But without that important types of attacks like DoS attacks cannot be studied!



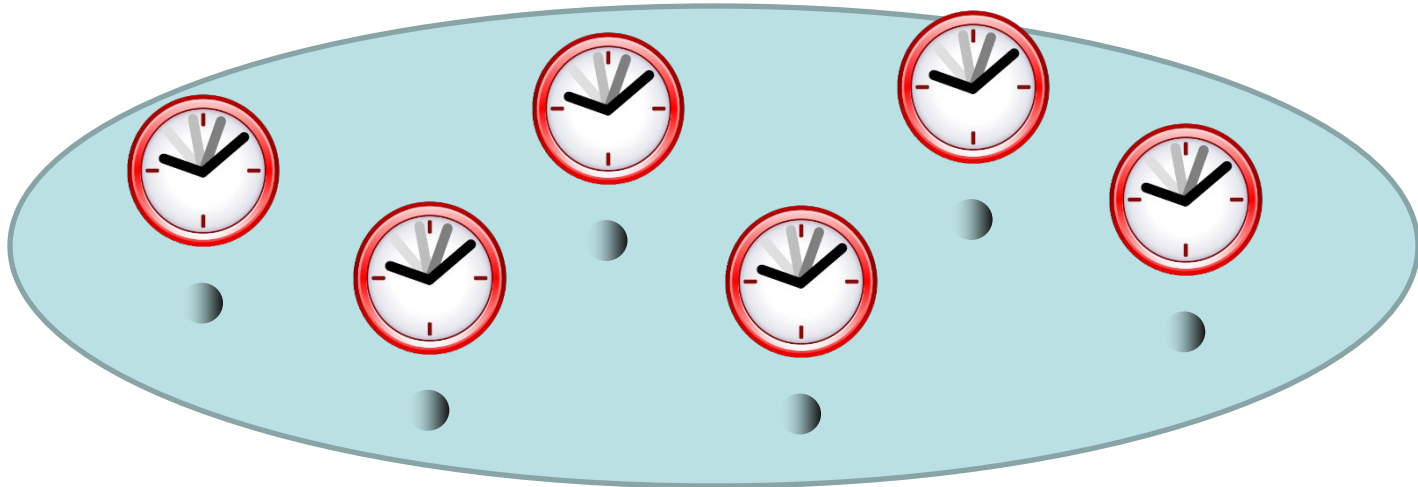
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Synchronization

The physical / logical clocks of the processes should show the same time.

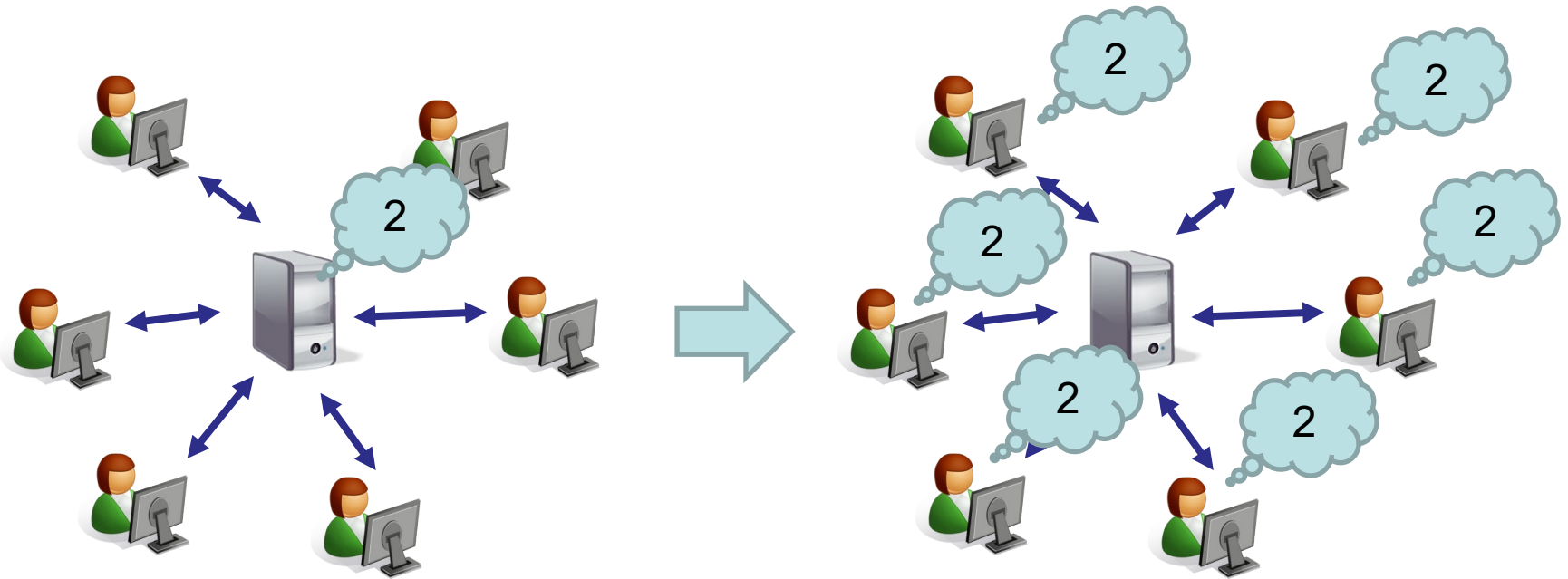


Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
- 5. Information Dissemination**
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Information Dissemination



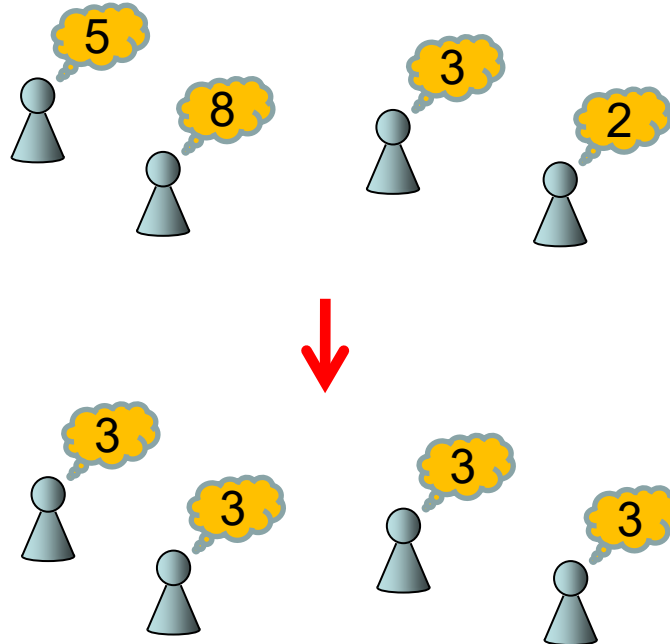
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Consensus

Arrive at a common decision.



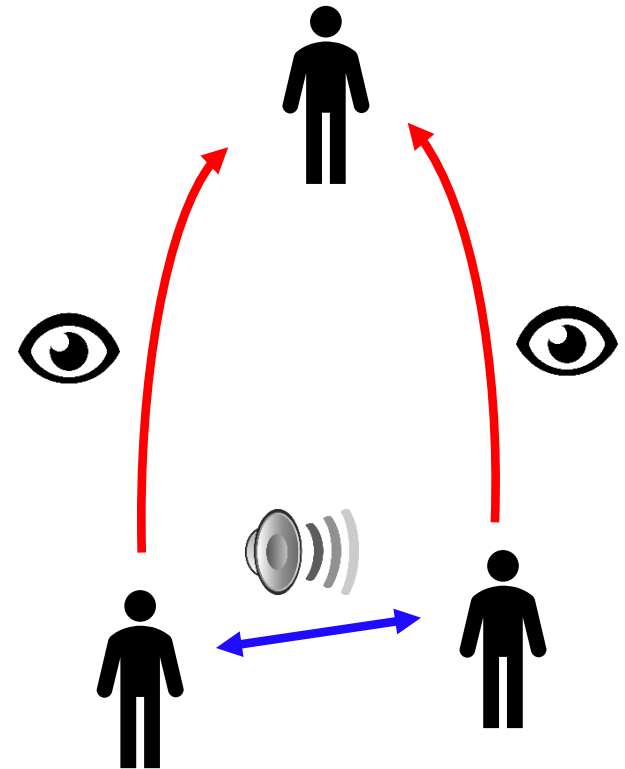
Advanced distributed algorithms and data structures

Contents:

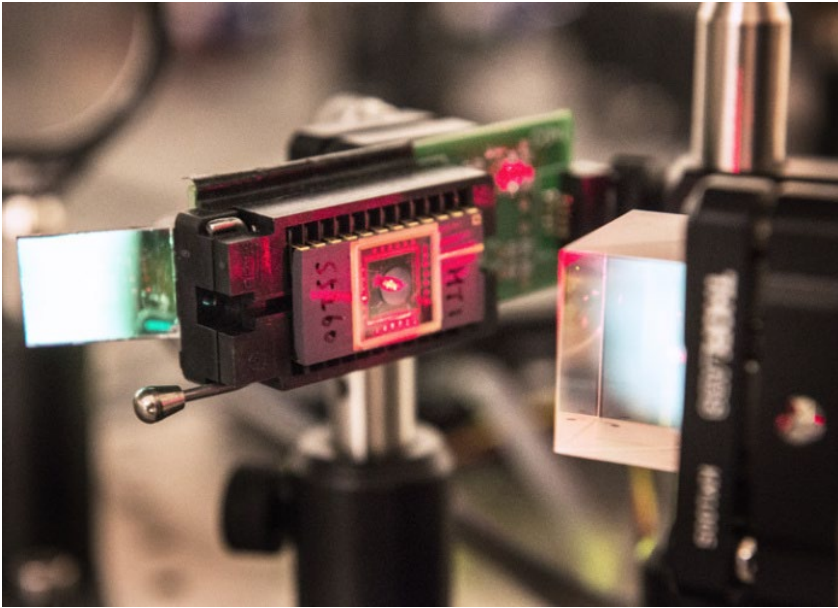
1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Hybrid Networks

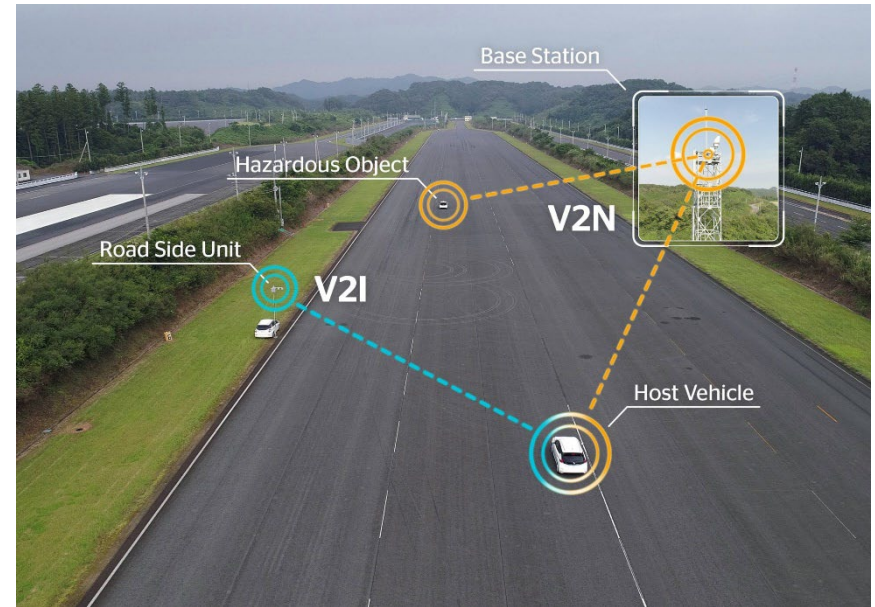
We all communicate in a hybrid fashion.



Hybrid Networks



Communication via cable network and lasers in data centers



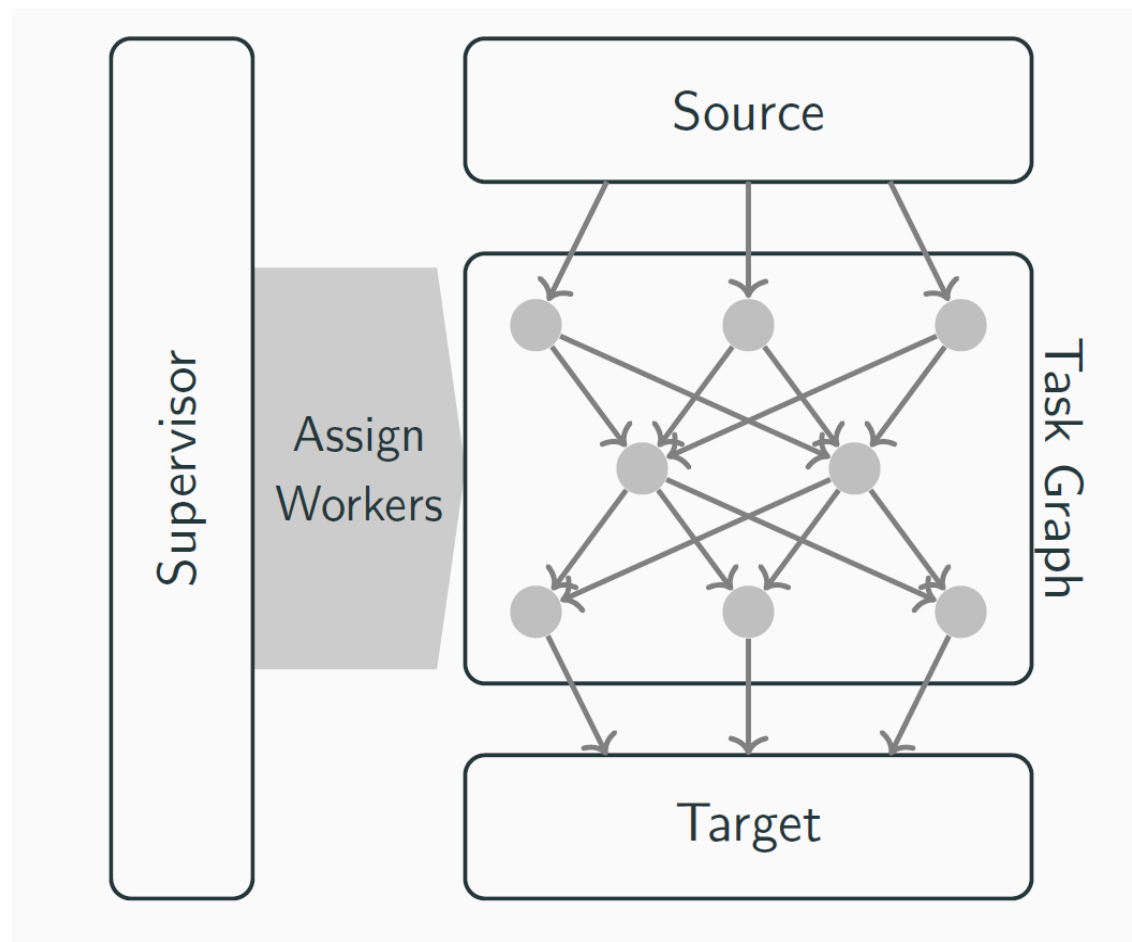
Car-2-Car communication via ad-hoc connections and cellular infrastructure

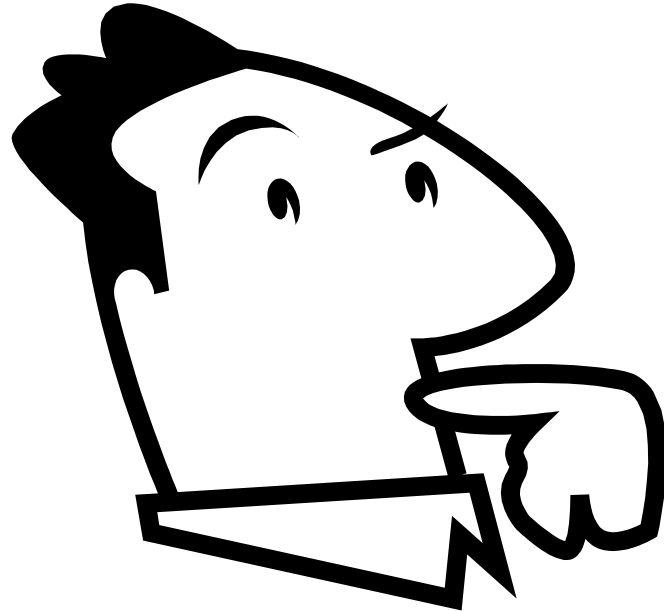
Advanced distributed algorithms and data structures

Contents:

1. Introduction
2. Foundations
3. Access Control
4. Synchronization
5. Information Dissemination
6. Consensus
7. Hybrid Networks
8. Supervised Distributed Computing

Supervised Distributed Computing

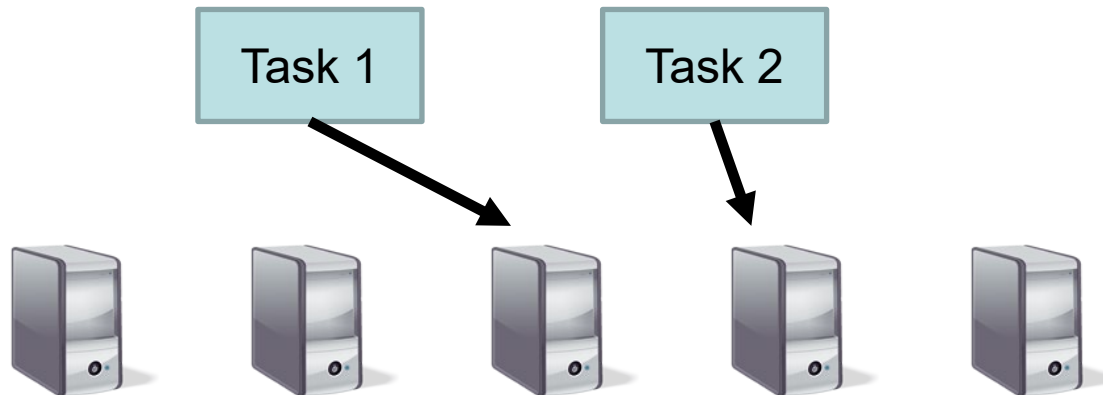
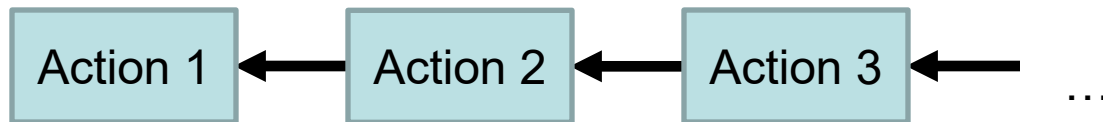




Questions?

If time permits: Distributed Scheduling

Ordering and matching problems



If time permits: Distributed Optimization

Smallest enclosing ball problem:

