Distributed Systems & Network Principles

Introduction

Dr Vincent Gramoli | Lecturer
School of Information Technologies

THE UNIVERSITY OF SYDNEY

Copyright

COMMONWEALTH OF AUSTRALIA Copyright Regulations 1969 WARNING

- This material has been reproduced and communicated to you by or on behalf of the University of Sydney pursuant to Part VB of the Copyright Act 1968 (**the Act**).
- The material in this communication may be subject to copyright under the Act. Any further copying or communication of this material by you may be the subject of copyright protection under the Act.
- **Do not remove this notice.**

THE UNIVERSITY OF SYDNEY

Outilne

- Administration
  Syllabus, goal, plagiarism, etc.
- Definition
  What is a distributed system?
- Examples
  Several distributed systems you might have heard of
- Challenges
  Scalability, Consistency, Fault-Tolerance

Administration

Timetable

- Title: Distributed Systems & Network Principles
- UOS code: COMP2121
- Credit point: 6
- Lecture:
  - Wednesday 12h-14h, weeks 1-13
  - School of IT Lecture Theatre
- Lab:
  - Wednesday 15h-16h, weeks 2-13
  - W15A Link Building 222 (north and south)

Myself

- Vincent Gramoli
  - Office: 417
  - Phone: (02) 903 69270
  - Vincent.Gramoli@sydney.edu.au
  - http://sydney.edu.au/engineering/it/~gramoli/
  - Office hour: Wednesday 14h-15h

- Background
  - Distributed storage, multicore programming.
  - France: IRISA, INRIA Futurs, Univ Rennes
  - USA: U. Connecticut, MIT, Cornel U.
  - Switzerland: U. Neuchatel, EPFL

eLearning

- eLearning: https://elearning.sydney.edu.au/
- Do not send email to the lecturer, but post questions online
- Notify us in advance if something unexpected will happen
- Usual things otherwise: notify us 7 days in advance
- Send proof (certificate) and you will be evaluated through an alternative mark assessment

THE UNIVERSITY OF SYDNEY

Description

- The unit will provide a broad introduction to the principles of distributed systems and their design; provide students the fundamental knowledge required to analyse and construct various types of distributed systems; explain the common architectural principles and approaches used in the design of networks at different scales (e.g. shared medium access and routing); introduce the programming skills required for developing distributed applications, and will cover the use of Java class libraries and APIs; cover common approaches and techniques in distributed resource management (e.g. task scheduling).

What you need to know

Distributed systems

- What is a distributed system?
- How it works?
- How to run yours?

Programming

- Most labs will require some programming
- All programming will be done in Java

Overall Goal

Distributed systems

- W5: Programming assignment 1: 10%
- W11: Programming assignment 2: 20%
- W7: Mid-semester exam: 20%
- Exam period: Final exam: 50%

- You must have at least 40% of each.

THE UNIVERSITY OF SYDNEY

Prerequisites

- Prerequisites: Java (INFO1103 or INFO1903) and (INFO1105 or INFO1905)
- Corequisites: Algorithms and complexity (COMP2007 OR COMP2907)

- You must have passed INFO1103 Introduction to Programming
- INFO1105 Data structures is not required but would help

- COMP2129 would help for concurrency

THE UNIVERSITY OF SYDNEY

Advice

- Lectures notes are for help
- You should understand in-depth
- Practice your reasoning by re-doing the examples at home
- Think about implications, ask questions
- Re-read your notes or the lecture notes at home after the class to memorize easily

Syllabus

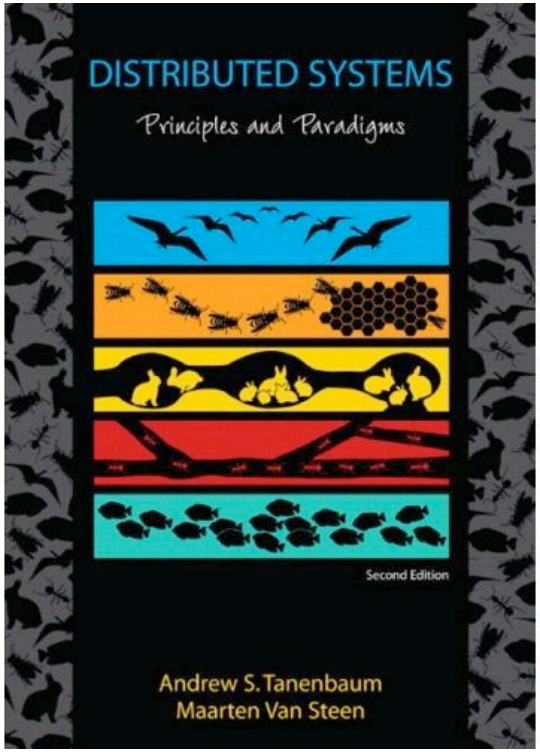
1. Introduction (Motivations) and architecture
2. Concurrency (Processes and Java threads)
3. Communication 1/2 (Network layers, routing, sockets)
4. Communication 2/2 (TCP/IP, RPC, RMI)
5. Naming (DNS, distributed file system)
6. Synchronization 1/3 (Physical time, logical time)
7. Synchronization 2/3 (Mutual exclusion, leader election, atomic broadcast)
8. Consistency (Causal consistency, database transactions)
9. Synchronization 2/3 (Chip Multiprocessors, transactional memory)
10. Failures (Crash and Byzantine failures, consensus problem)
11. Dynamism (Gossip-based protocols)
12. Security (Cryptography secured channels)
13. Review

Plagiarism

- *Plagiarism:* Copying all or parts of another work with or without their knowledge
- Time consuming
- Unfair to other students
- Do not encourage plagiarism, showing your own work makes you guilty
- Make sure to cite your source appropriately, so that one can have a look

Textbook

Distributed Systems – Principles and paradigms Tanenbaum and Van Steen. 2nd Edition.



Definition

What is a Distributed System?

A collection of independent computers that appears to its users as a single coherent system.

What is a Distributed System?

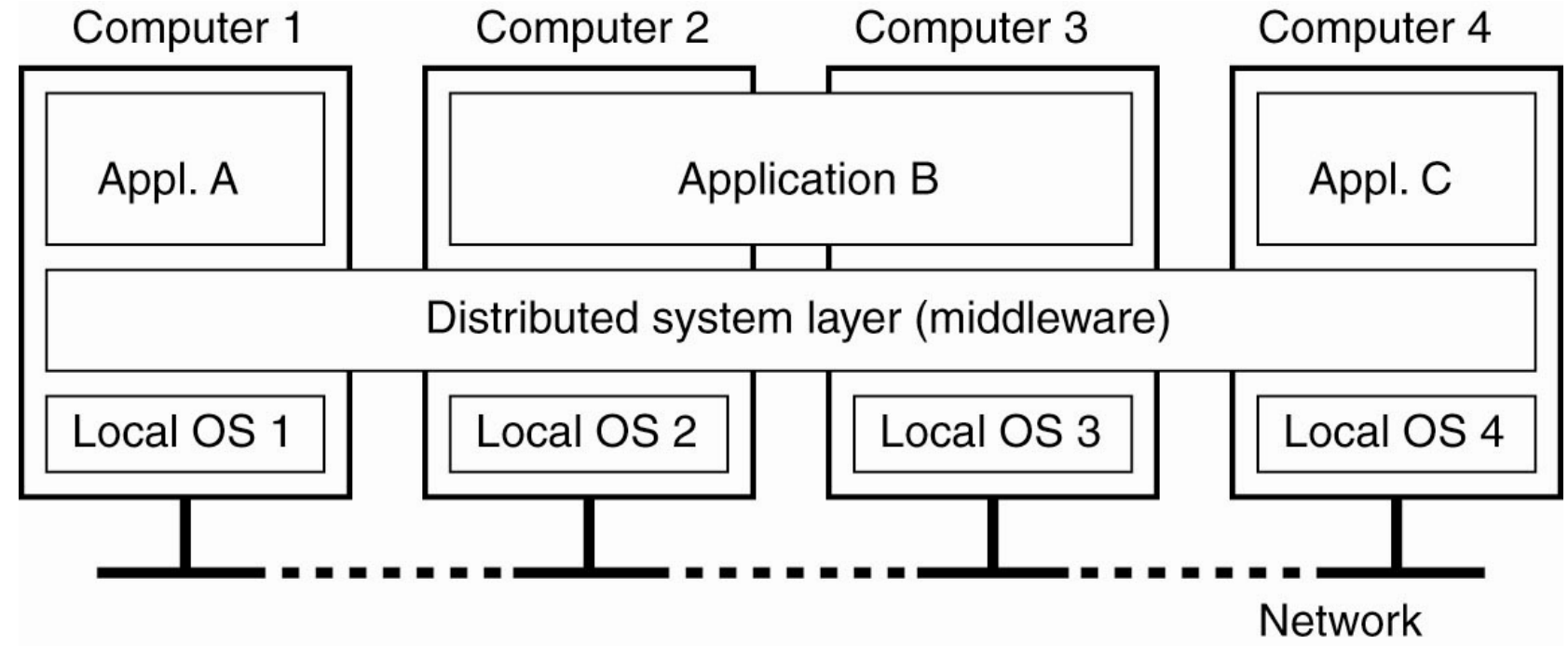
Transparency helps the users observe a single coherent system

Transparency	Description
Access	Hide differences in data representation and how a resource is accessed
Location	Hide where a resource is located
Migration	Hide that a resource may move to another location
Relocation	Hide that a resource may be moved to another location while in use
Replication	Hide that a resource is replicated
Concurrency	Hide that a resource may be shared by several competitive users
Failure	Hide the failure and recovery of a resource

› The different forms of transparency in a distributed systems

Middleware

A distributed systems can be viewed as a *middleware*



› The middleware layer extends over multiple machines and offers each application the same interface

Examples

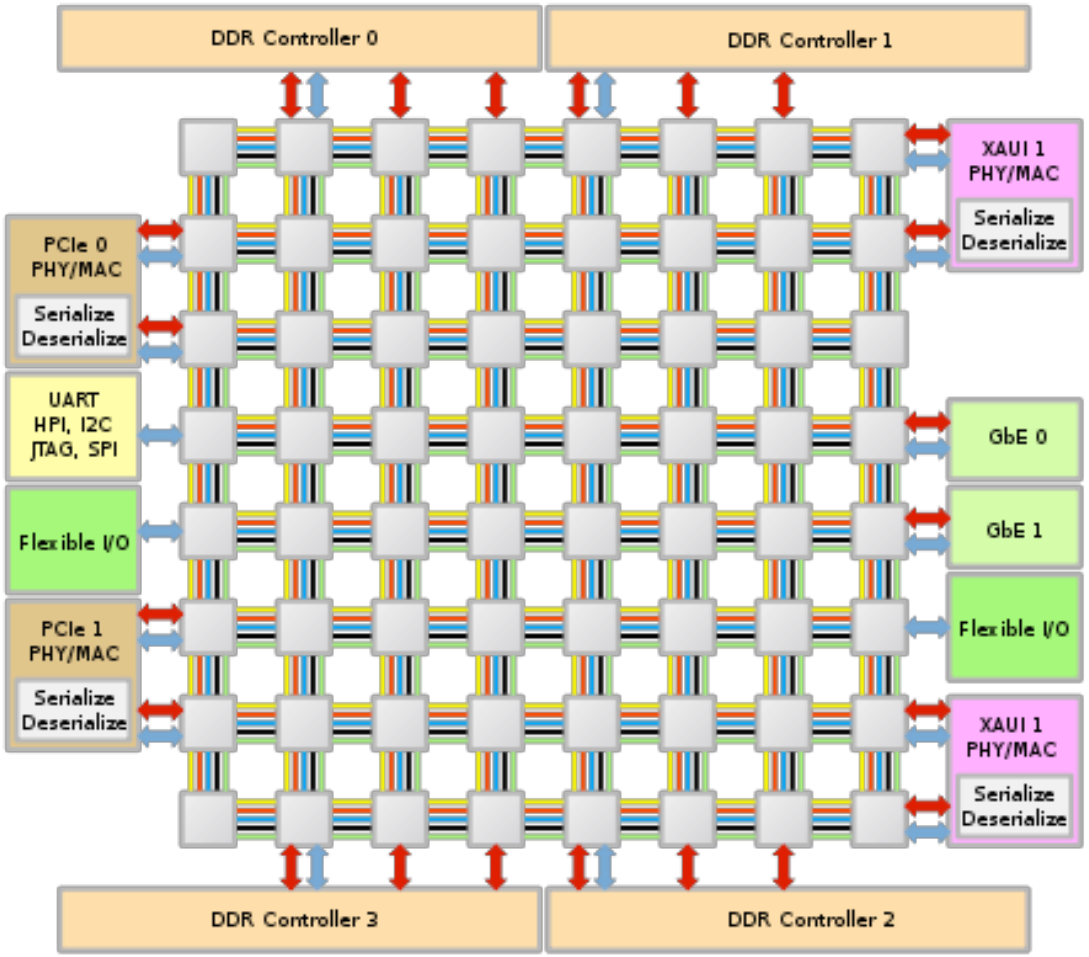
Cloud computing

THE UNIVERSITY OF SYDNEY

Cluster

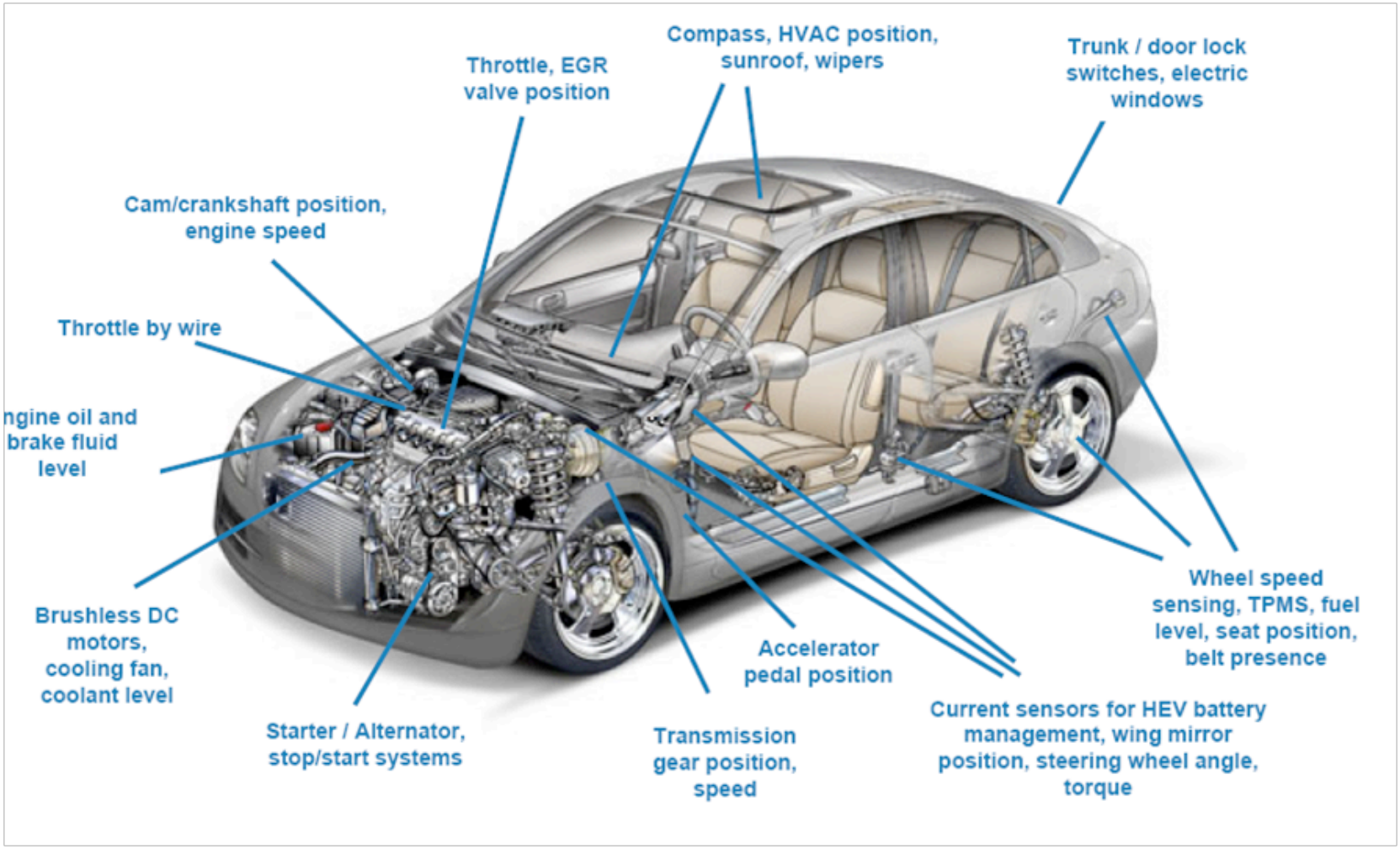
Chip Multiprocessors

TILERA®

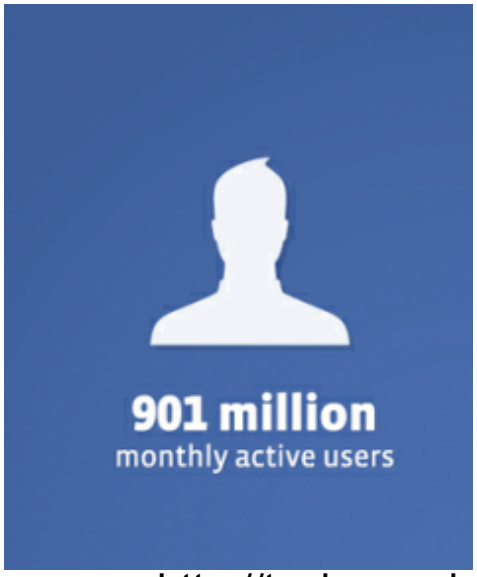


THE UNIVERSITY OF SYDNEY

Sensor Networks



Social Networks



source: http://techcrunch.com/2012/04/23/

INTERNET USERS AND POPULATION STATISTICS FOR OCEANIA

SOUTH PACIFIC REGION	Population (2011 Est.)	% Pop. of World	Internet Users, 31-Dec-11	Penetration (% Population)	Users % World	Facebook 31-Dec-11
Total for Oceania	35,426,995	0.5 %	23,927,457	67.5 %	1.1 %	13,353,420
Rest of the World	6,894,628,159	99.5 %	2,243,306,285	32.5 %	98.9 %	785,738,740
WORLD TOTAL	6,930,055,154	100.0 %	2,267,233,742	32.7 %	100.0 %	799,092,160

source: http://internetworldstats.com/stats.htm

Challenges

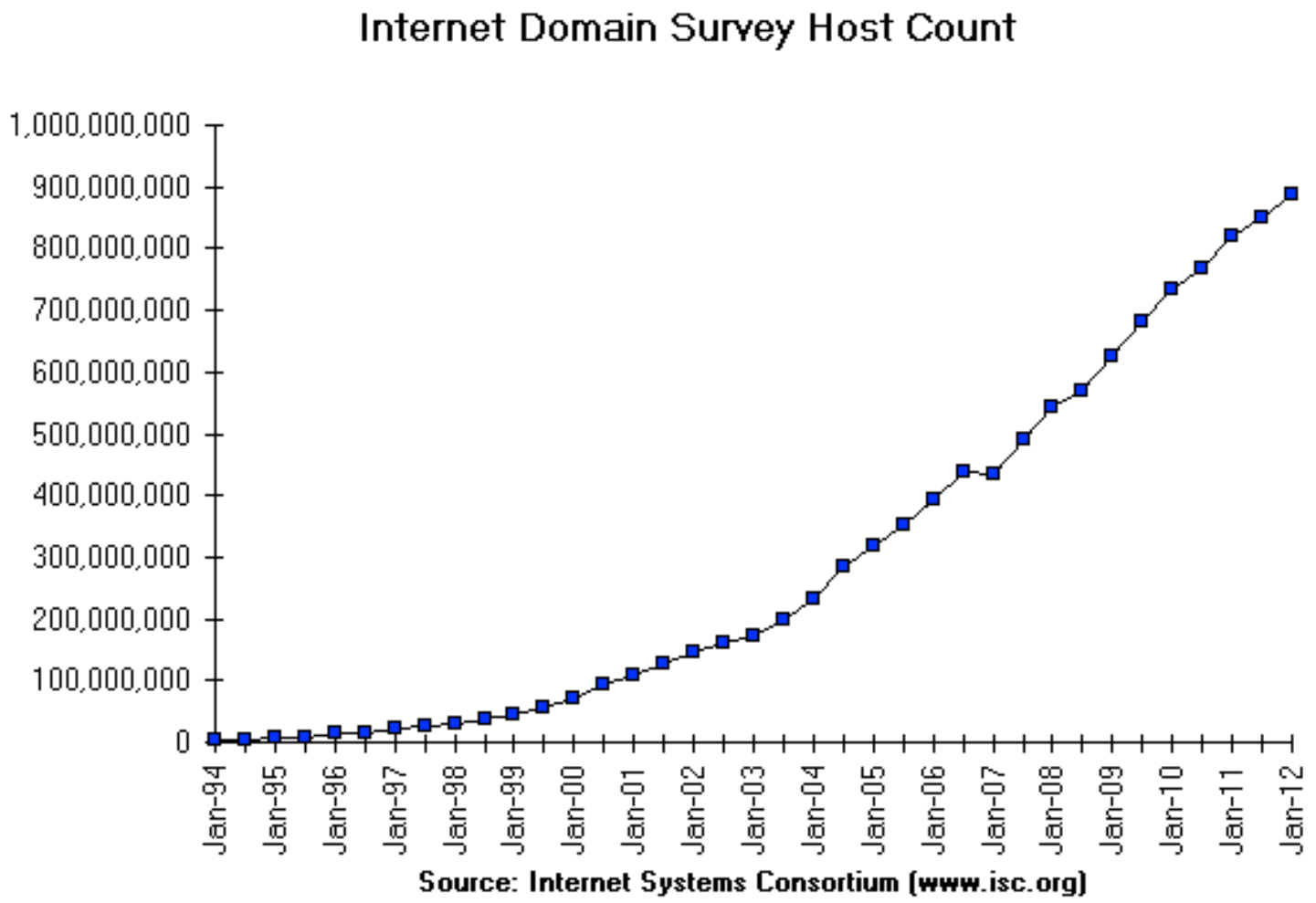
Scalability

- *Scalability of a distributed system:* the ability for the system to preserve some properties as the system grows in
  - the number of requests or participants,
  - the distance between resources and users, or
  - the heterogeneity.

- Scalability limitations:

Concept	Example
Centralized services	A single server for all users
Centralized data	A single on-line telephone book
Centralized algorithms	Doing routing based on complete information

Scalability



Scalability

- **Burst of load:**
  - **456** tweets per second (TPS) when Michael Jackson died (June 25, 2009).
  - **6,939** TPS after midnight in Japan on 2011 New Year's day.
- **Increase in participation:**
  - **+182%**: Increase in number of mobile users in 2010.
  - **>500,000** new accounts created on a single day.

Source: http://blog.twitter.com/2011/03/numbers.html

THE UNIVERSITY OF SYDNEY

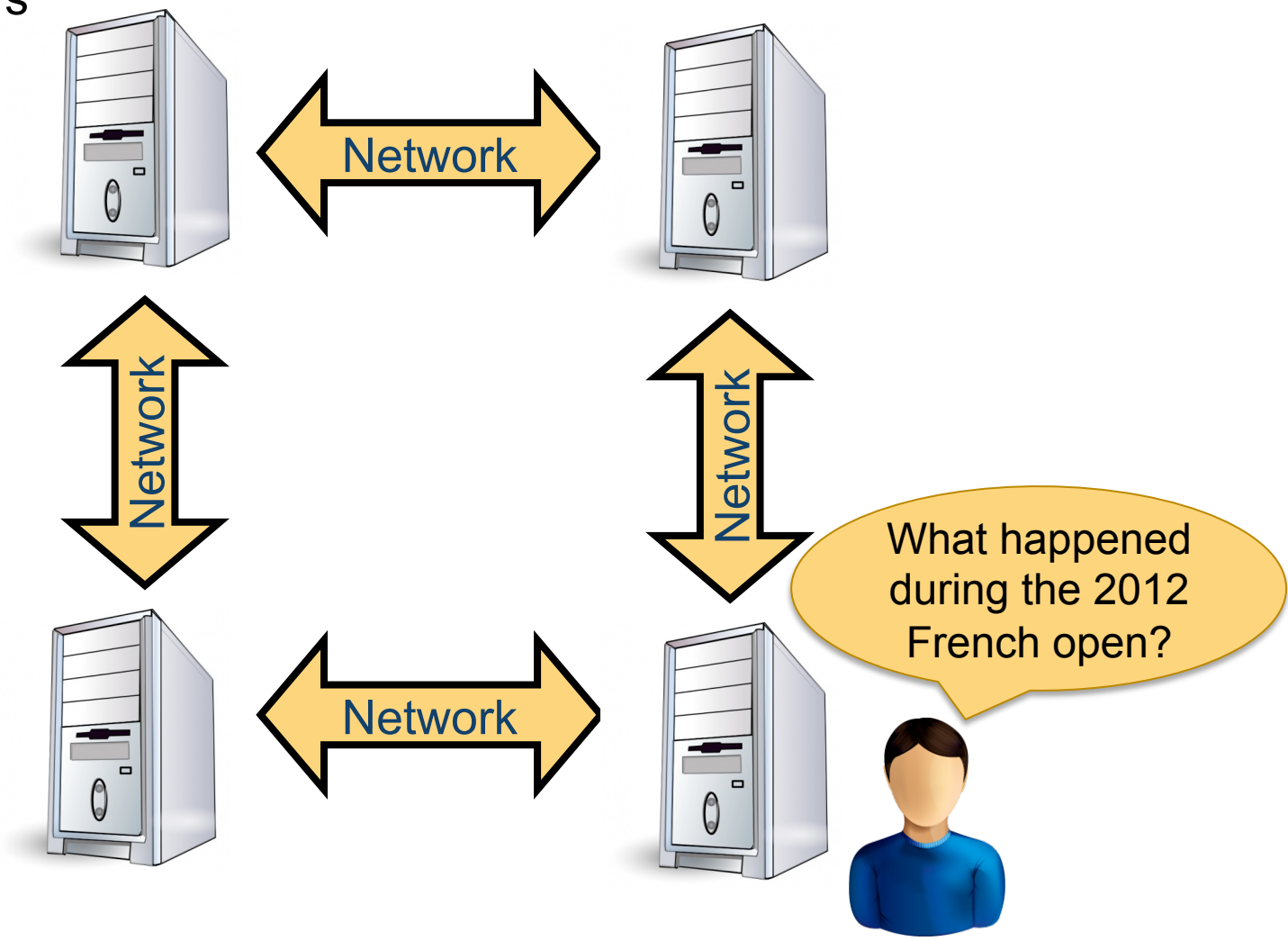
Consistency

- *Consistency;* a property applying to a collection of data items that are accessed by distributed participants.

- Examples of inconsistencies: As a participant, I observe that Djokovic lost against Nadal but then Djokovic won against Federer in the 2012 French Open.

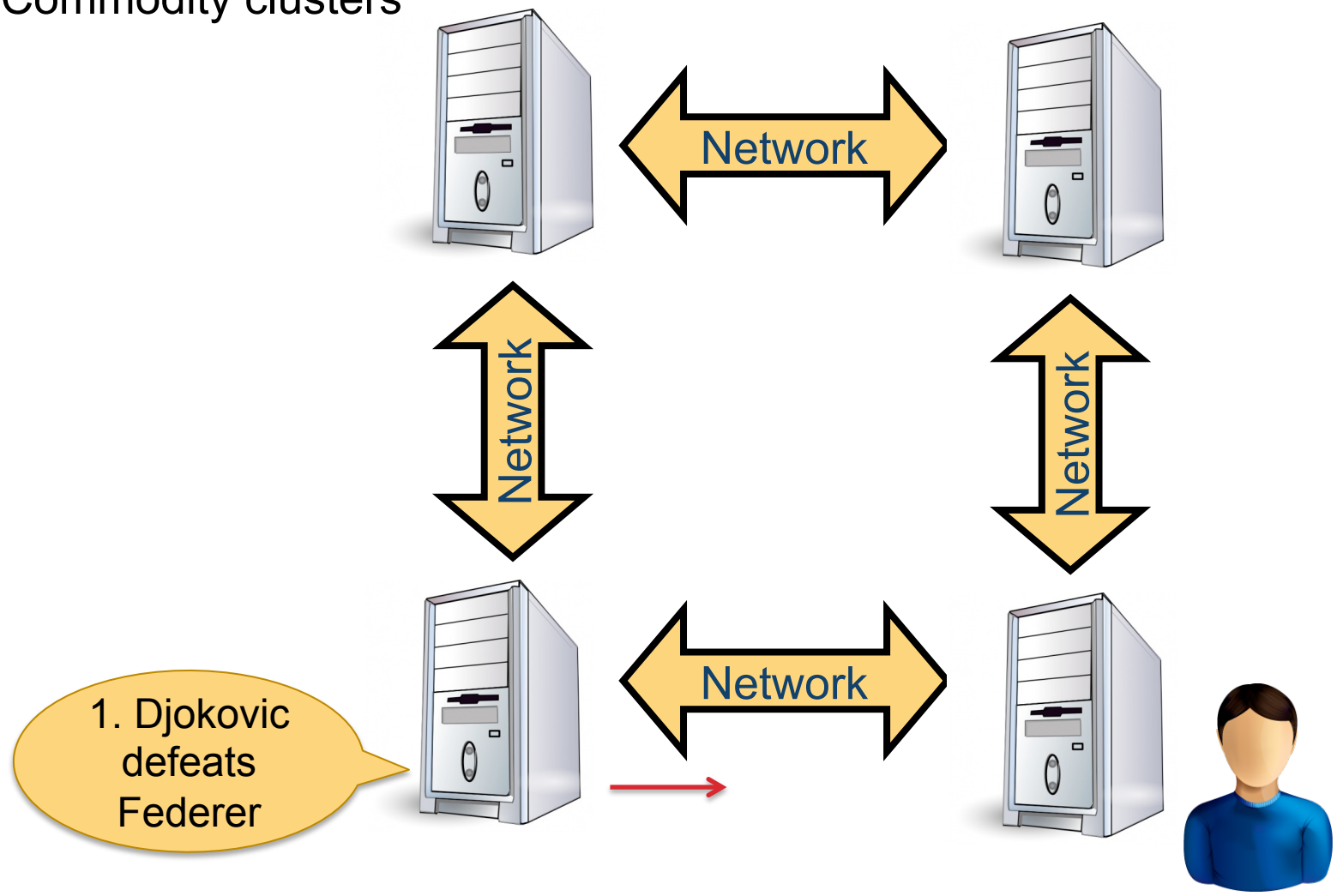
Consistency

Commodity clusters



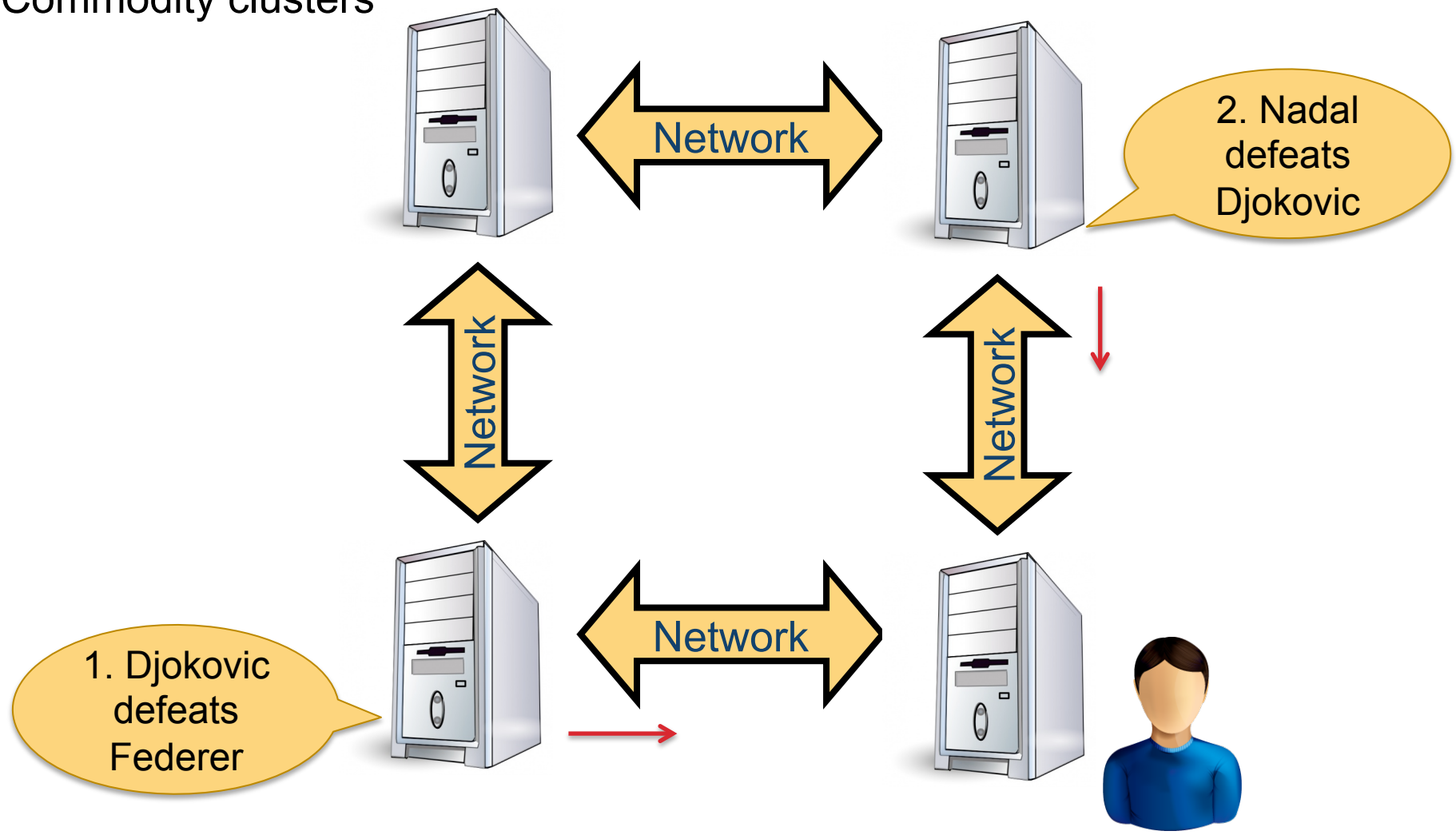
Consistency

Commodity clusters



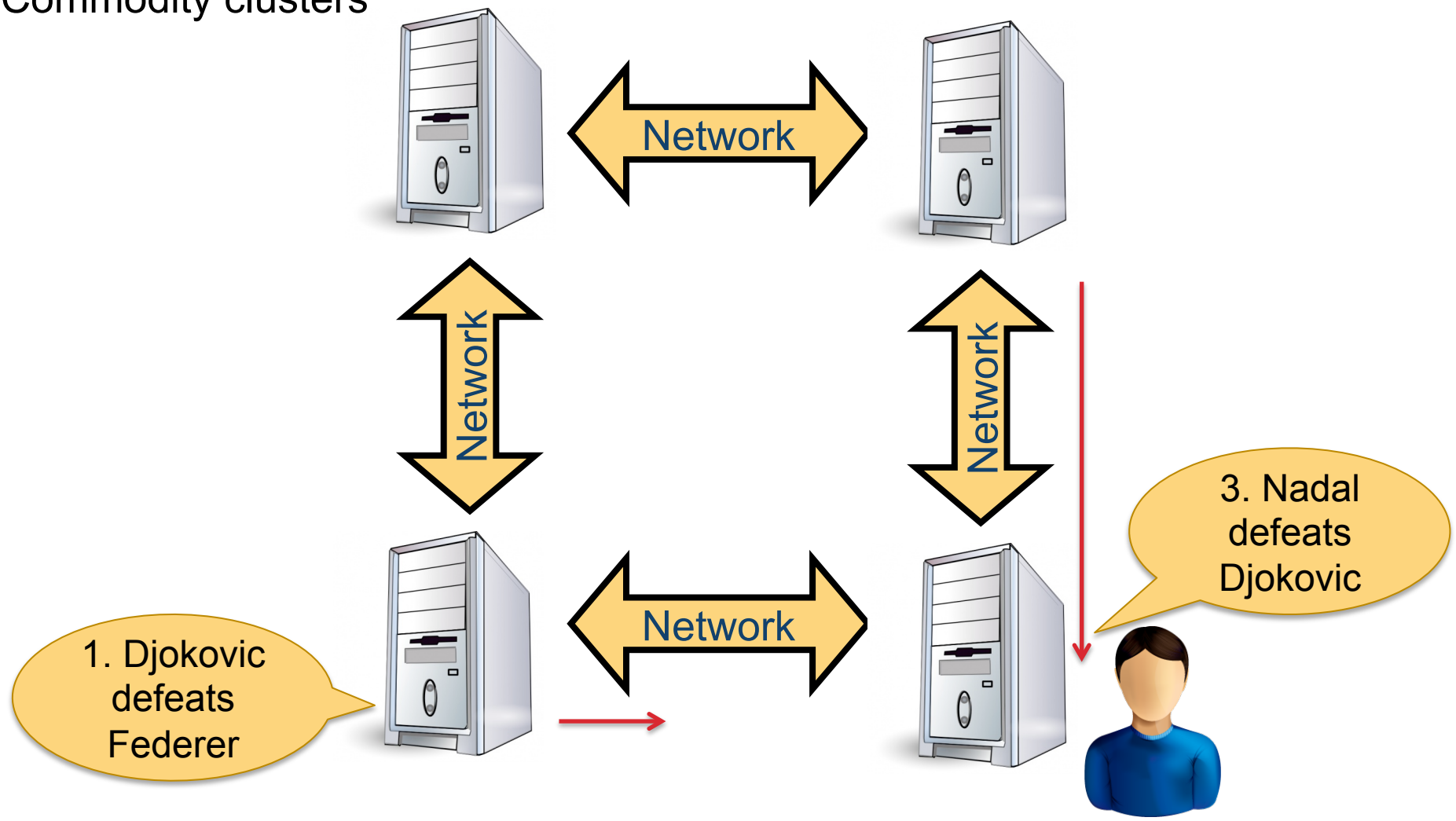
Consistency

Commodity clusters



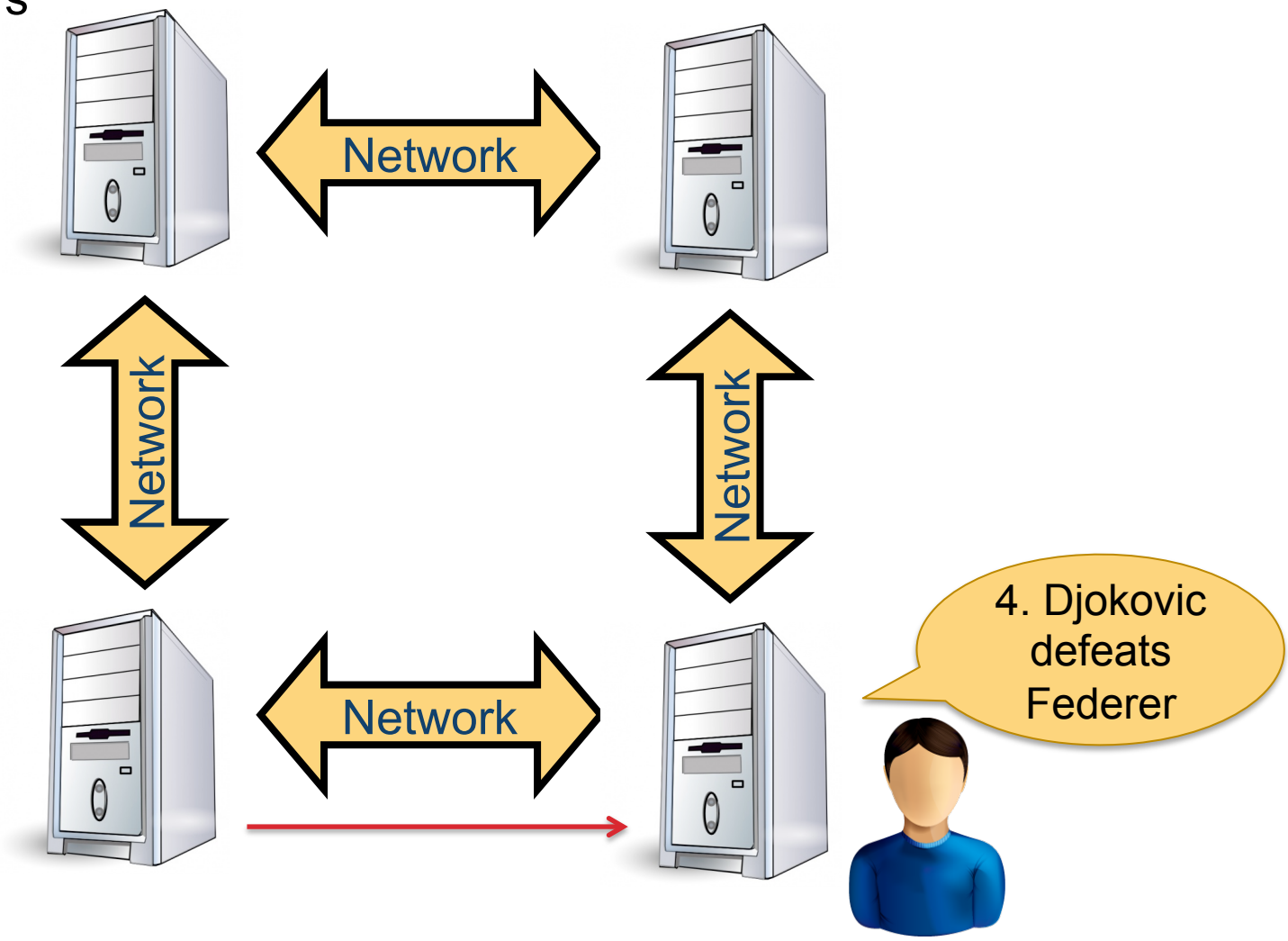
Consistency

Commodity clusters



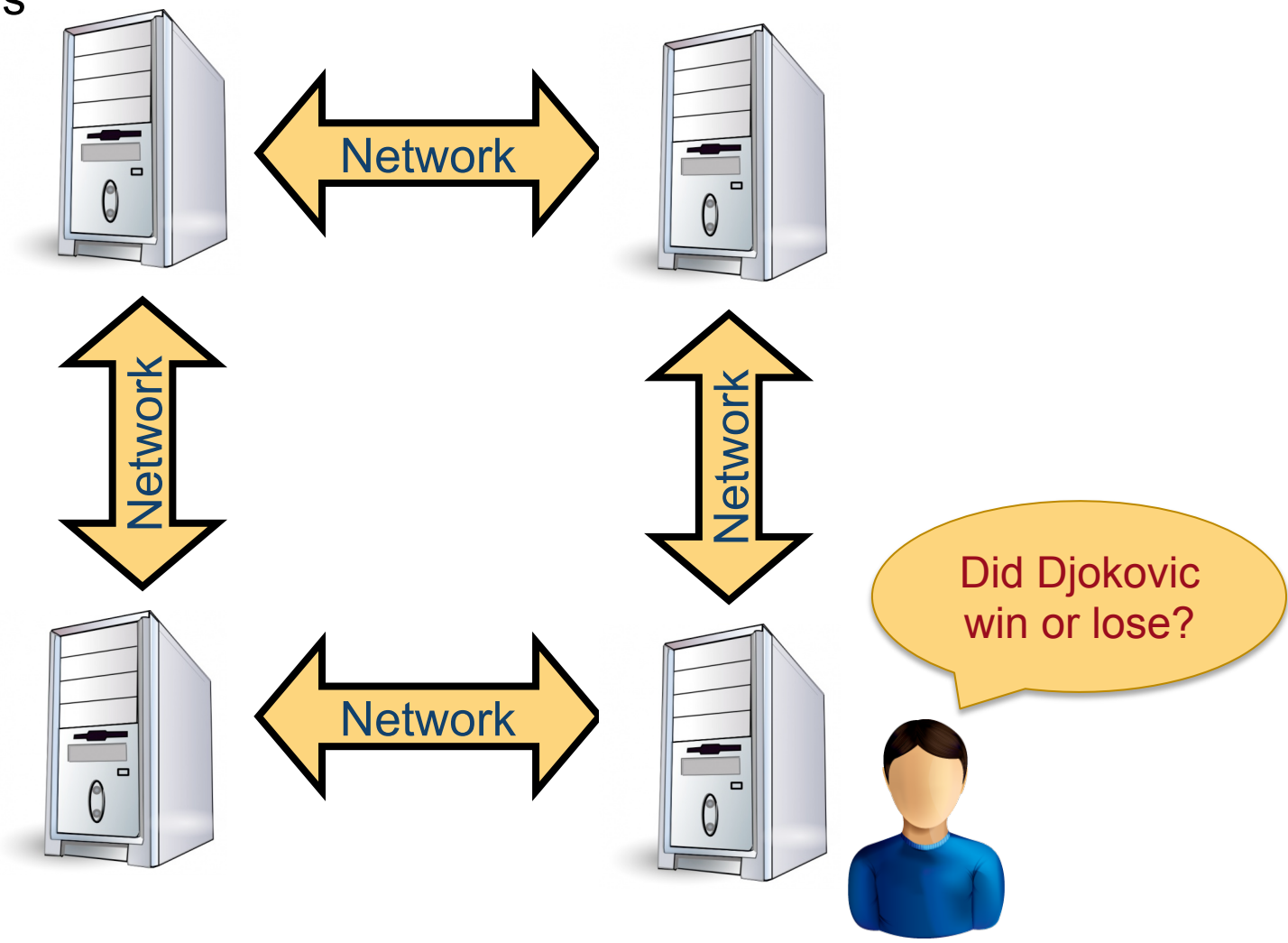
Consistency

Commodity clusters



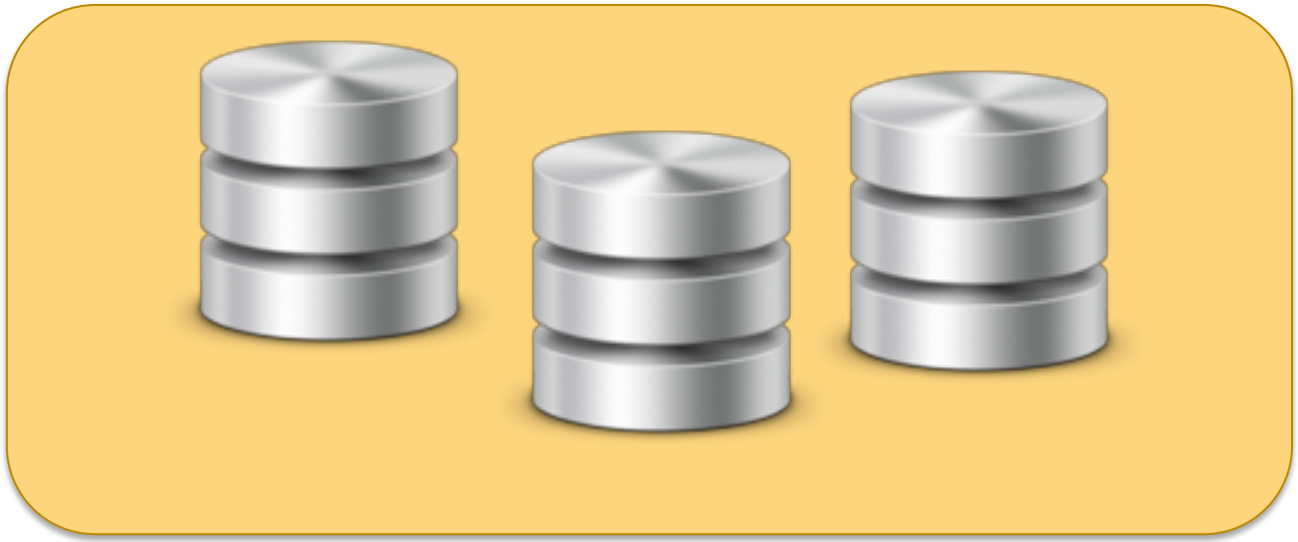
Consistency

Commodity clusters



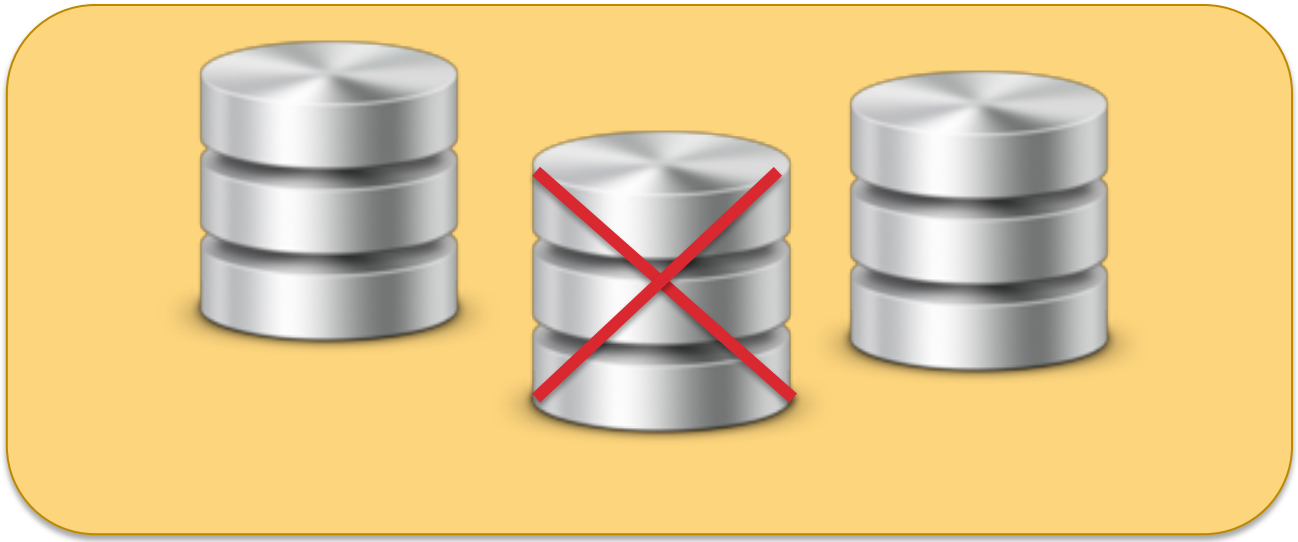
Fault-Tolerance

› *Fault-tolerance of a distributed system:* the ability for the system to recover from partial failures.



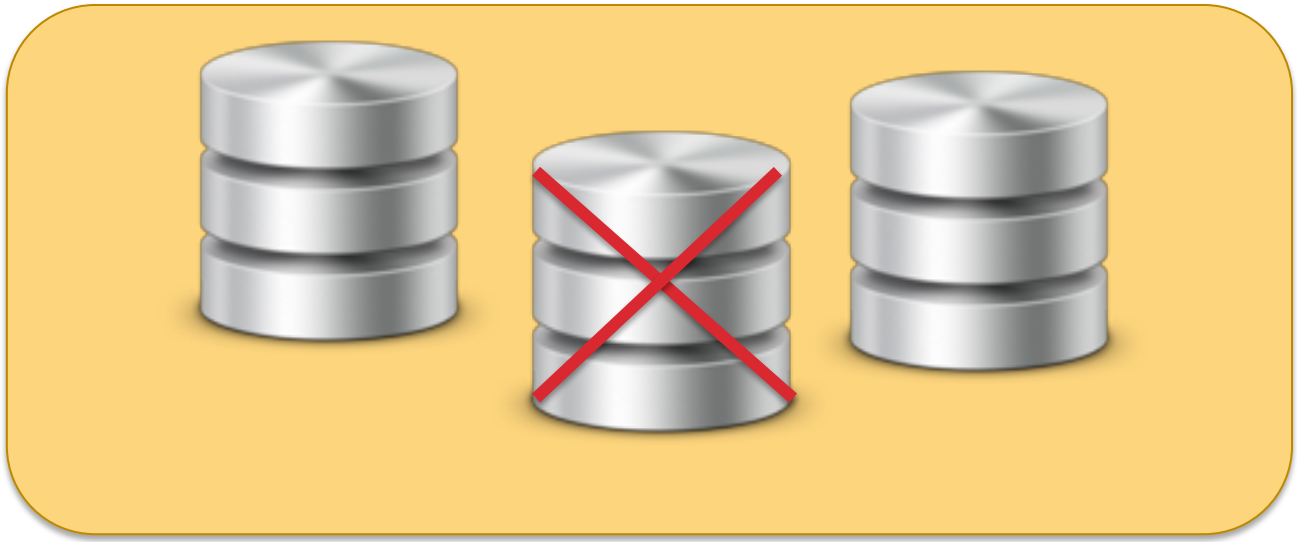
Fault-Tolerance

› *Fault-tolerance of a distributed system:* the ability for the system to recover from partial failures.

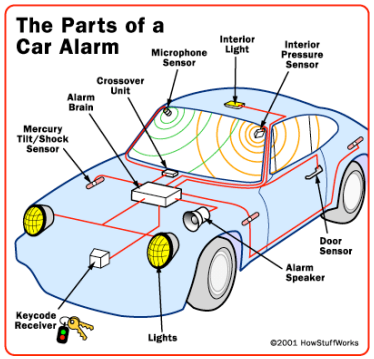


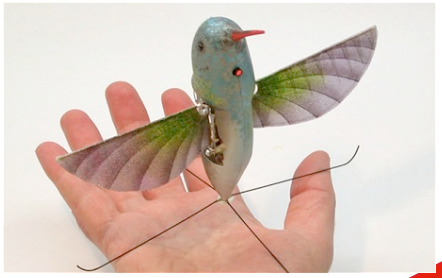
Fault-Tolerance

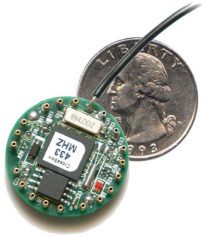
- *Fault-tolerance of a distributed system:* the ability for the system to recover from partial failures.
- How to keep the distributed system up and running, thereby appearing as a single running system to its users?

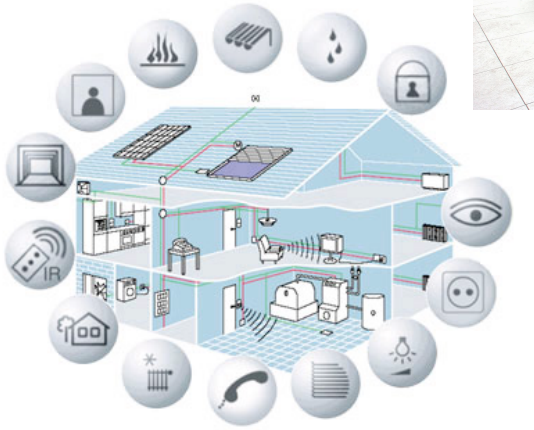


Societal Scale









Conclusion

- A distributed system comprises multiple entities but appears as a single system
- There are various kinds of distributed systems
- There are three main challenges to address in distributed systems:
  - Scalability
  - Consistency
  - Fault-tolerance