7.3.2006 / ta
15.1.2007 /LK

Distributed systems

Topic	Prerequisites	Just passing boundary	Achieves goals	Extensive results	Litterature [1]
Goals and problems of distribution	Operating systems and data communication - knows the basic concepts - is able to explain main services of operating systems and data communication and is able to explain what happens in the system when each of these services are used - knows the basic architecture of these systems	Based on case studies is able to describe the basic goals, challenges and problems of distribution Is able to describe, using examples, the concepts of transparency, heterogeneity, openness, scalability, and consistency Is able to define and motivate the concept of "distributed system".	Is able to describe and motivate concepts of concurrency, partitioning, replication and their effects on the availability, trustworthiness, performance and scalability of a service. On a qualitative level, is able to estimate the benefits and drawbacks of distributing an application. Understands[2] distributed time, concurrency and non-determinism as a sources of distribution problems.	Is able to evaluate different distribution solutions in terms of performance and scalability. Is able to evaluate fault tolerance and trustworthiness of different solutions. Knows security threats.	TaSte, ch1.2 CoDoKi, ch 1
Structure of distributed systems	Knows the principles of Internet DNS and www (TilPe) On the level of algorithms, is able to use RPC, RMI,	Knows the main differences between hardware architectures . Is able to characterize differences between network operating systems, distributed operating	On the level of principles, is able to describe relevant - hardware architectures - operating system and middleware architectures - application level	Is able to evaluate benefits and costs of each solution.	TaSte, ch 1 CoDoKi, ch 2

[1] Abbreviations :
TaSte Tanenbaum, van Steen, Distributed Systems Principles and Paradigms, Prentice Hall, 2002
CoDoKi Coulouris, Dollimore, Kindberg, Distributed Systems, Concepts and Design, 4th ed., Addison - Wesley , 2005

[2] The term "understands" means the ability of explaining and motivating the challenge, solution and relevant interdependencies and consequences between concepts, to someone new to the topic.

	message passing as a tool for remote communication (RiO: RMI & KJ)	systems, and middleware; and describe the shared goals of those approaches Understands ”vertical distribution” (multitiered architectures). Knows about mobility, the risks and potential thereof	architecture models Understands the dependencies between different solutions, and understands the consequences that different solutions cause to the application layer (performance, reliability)		
Coordination of the systems and communication Distributed decision-making	Is able to explain the principles of using message passing systems (RIO) and the implementation principles of IPC (KJ) Knows about the non-determinism and unreliability aspects of message passing (TilPe) Knows about transactions (Info)	Is able to explain how clocks are synchronized. Understands the need of logical scalar and vector clocks and is able to explain their implementation algorithms. Is able to describe total ordering, FIFO ordering, and causal ordering for multicast and understands their implementation principles and algorithms. Understands problems of determining global state and explain them with examples. Is able to describe an example case where distributed snapshot	Is able to implement on algorithm level - total and causal sequencing of events - distributed checkpointing - distributed mutual exclusion - distributed decision-making Knows several solutions for decision-making (use of coordinator, ring algorithms, other distributed algorithms) and is able to choose a suitable solution for an application. Is able to motivate the correct behaviour of the algorithms and knows the prerequisites for applicability.	Knows the new literature on the field. Is able to use formal methods for showing the correctness of the algorithms on the area. Knows the more advanced methods of transaction serialisation in distributed systems.	TaSte, chapter 5 CoDoKi, chapters11 – 13

		is taken (algorithm). Understands problems of distributed decision-making. Is able to describe at least one implementation algorithm for distributed mutual exclusion and election. Is able to describe management of transactions in a distributed system. Is able to explain concepts of serialisability and how it can be reached using locks.	Is able to evaluate the efficiency of the algorithms in different kind of environments. Is able to describe the behaviour of distributed transactions and serialisability using locks and timestamps.		
Replica manage-ment	Knows the structural basis for Internet DNS ja www services (TilPe)	Knows why objects are replicated and how replication of objects is implemented. Is able to explain how consistency problems emerge. Understands the difference of data centric and user centric consistency models. Is able to describe the essential models and compare them.	Is able to describe the main concept features of consistency models. Is able to apply suitable consistency models in different kind of environments (ranging from fast fixed networks to uncertain ad hoc networks) Is able to describe the use of both epidemic and majority-based update algoritms in replica management.	Is able to design and implement replica management (basic) solution for fixed networks. Is able to design replica management main features for mobile ad hoc networks. Is able to design and implement epidemic and majority-based update algoritms in replica management.	TaSte, chapter 6 CoDoKi, chapter 15

		Is able to choose a consistency model for an application situation. Is able to describe the main methods of implementation for replications and replicate updates. Knows epidemic and majority-based update methods.		Knows the modern litterature on the area.	
Fault tolerance methods	Knows the parity based and redundancy based error-detection and error-recovery mechanisms for data storage and data communication (TiTo + TilPe)	Is able to use the basic terminology correctly. Knows the failure models and is able to describe their differences and their usage. Is able to describe the fundamental fault tolerance mechanisms. Knows on what prerequisites it is possible to establish mutual agreement in a unreliable group (is able to describe the problem of Byzantine generals). Knows how reliable multicast can be implemented in a dynamic group environment.	Is able to describe dependencies and interactions between faults and fault management mechanisms. Is able to describe how unreliable group members can reach agreement. Is able to describe an algorithm that implements reliable multicast in a dynamic group and motivate the correct behaviour of that algorithm. Is able to motivate the correct behaviour of the 2PC algorithm in different failure situations. Knows methods for	Knows the modern literature on the area. Is able to use probabilities for estimating fault tolerance solutions.	TaSte, chapter 7 CoDoKi, chapters 2, 12, 14, 15

			distributed snapshot and is able to motivate its correct behaviour.		
Distributed file systems	Knows how file systems are structured (KJ)	Is able to explain the founding differences between NFS and CODA. Is able to describe the following features for files systems such as NFS and CODA - file sharing - use of caching - fault tolerance	Is able to motivate the correct behaviour of NFS and CODA and explain the limitations of the specified behaviour. Is able to explain how consistency control and conflict resolution is arranged in CODA. Is able to apply the methodological parts of the course to the design of distributed file systems and understanding their behaviour.	Is able to design the basic structure of a file system for mobile ad hoc network.	TaSte, chapter 10 CoDoKi, chapter 8