Blockchain 1

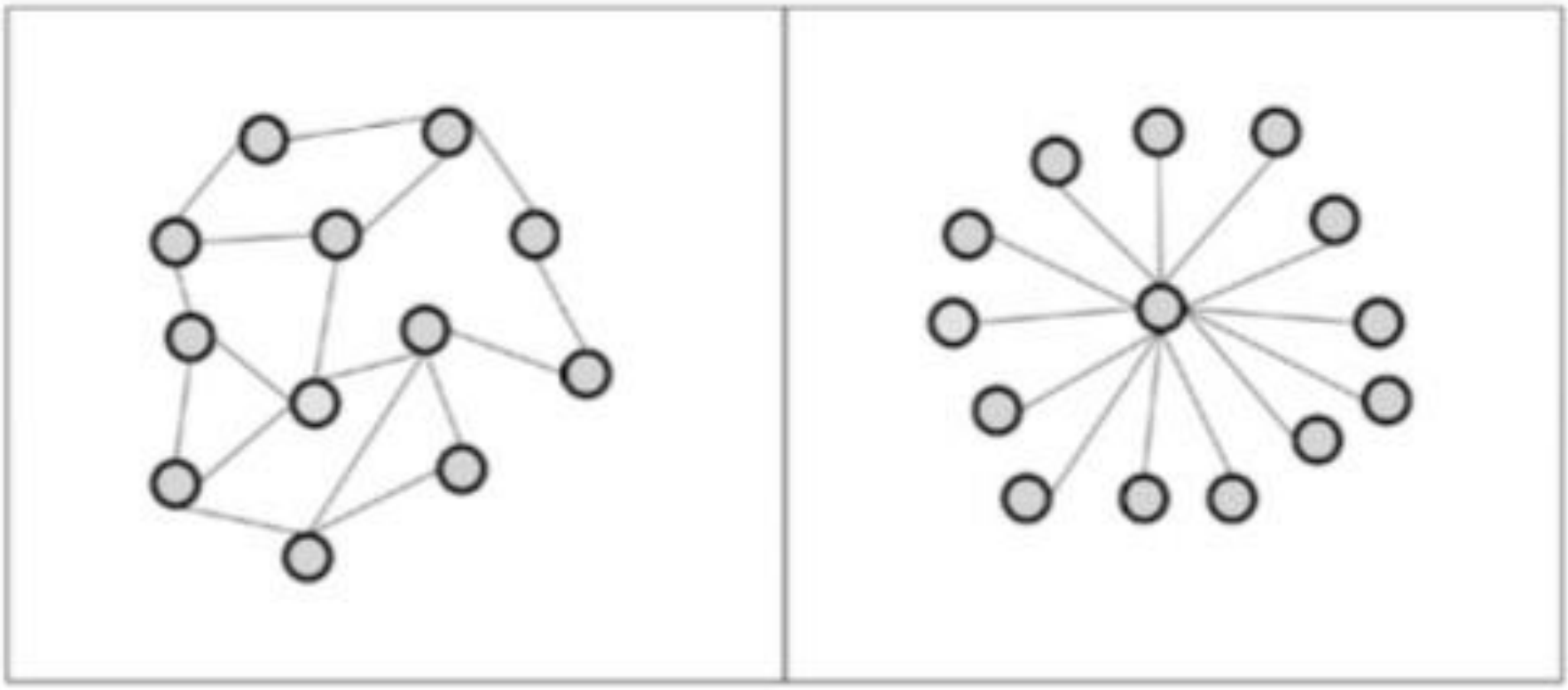
This set of slides was developed based on the reference books as listed in the course document.

Seeing the Big Picture

A Payment System

Layer	Functional Aspects	Nonfunctional Aspects
Application	Deposit money	The graphical user interface looks beautiful
	Withdraw money	Easy to use
	Transfer money	Transfer of money is done fast
	Monitor account balance	System has many participants
Implementation	?	Available 24 hours a day
		Fraud resistant
		Maintaining integrity
		Ensure user privacy

A Payment System - 2 Types of Software Architecture



Distributed (left) vs. centralized (right) system architecture

A Payment System - The Advantages of Distributed Systems

Higher Computing Power: The computing power of a distributed system is the result of combining the computing power of all connected computers.

Cost reduction: the costs of creating, maintaining, and operating a super computer are still much higher than the costs of creating, maintaining, and operating a distributed system.

Higher reliability: single point of failure

Ability to grow naturally: scalability

A Payment System - The Disadvantages of Distributed Systems

Coordination overhead:

Communication overhead:

Dependency on networks: without any network, there will be no distributed system, no communication, and therefore no coordination among the nodes, thus the dependency on networks.

Higher program complexity: software developed for a distributed system must address the additional problems such as coordination, communication, and utilization of networks ==> increase in complexity

Security issues: e.g. authentication and authorization problems

Blockchain?

Each of the two architectural concepts has its own advantages and disadvantages and their own specific way of doing things. Choosing a specific architecture has consequences on how you will achieve the functional and nonfunctional aspects of a system.

In particular, both architectural concepts have very different approaches to ensure integrity.

And this is the point where the blockchain enters the picture. **The blockchain is a tool for achieving integrity in distributed software systems.** Hence, it can be seen as a tool to achieve a nonfunctional aspect of the implementation layer.

Discovering the Core Problem

The Core Problem

Maintaining integrity in distributed systems is the major purpose of the blockchain. Why is maintaining integrity in distributed systems is a challenge?

In answering this question, we will discover the subtle relation between ***trust*** and ***integrity***.

Trust and integrity are two sides of the same coin. In the context of software systems, ***integrity*** is a nonfunctional aspect of a system to be ***safe, complete, consistent, correct, and free of corruption and errors***. ***Trust*** is also the firm belief of humans in the ***reliability, truth, or ability of someone or something without evidence, proof, or investigation***.

Trust is given in advance and will increase or decline based on the results of interactions on an ongoing basis.

The Core Problem (cont.)

Due to the importance of trust for the existence of peer-to-peer systems, the major question is: How do we achieve and maintain integrity in a purely distributed peer-to-peer system?

With respect to peer-to-peer systems, this means that people will join and continue to contribute to a system if they trust it and if the results of interacting with the system on an ongoing basis confirm and reinforce their trust. Integrity of the system is needed in order to fulfill the expectations of the users and reinforce their trust in the system. If the trust of the users is not reinforced by the system due to a lack of integrity, the users will abandon the system, which, as a result, will eventually cause it to terminate.

The Core Problem to be solved by the Blockchain

The core problem to be solved by the blockchain is:

- **achieving and maintaining integrity**
  - **in a purely distributed peer-to-peer system**
    - **that consists of an unknown number of peers with unknown reliability and trustworthiness.**

Defining Blockchain

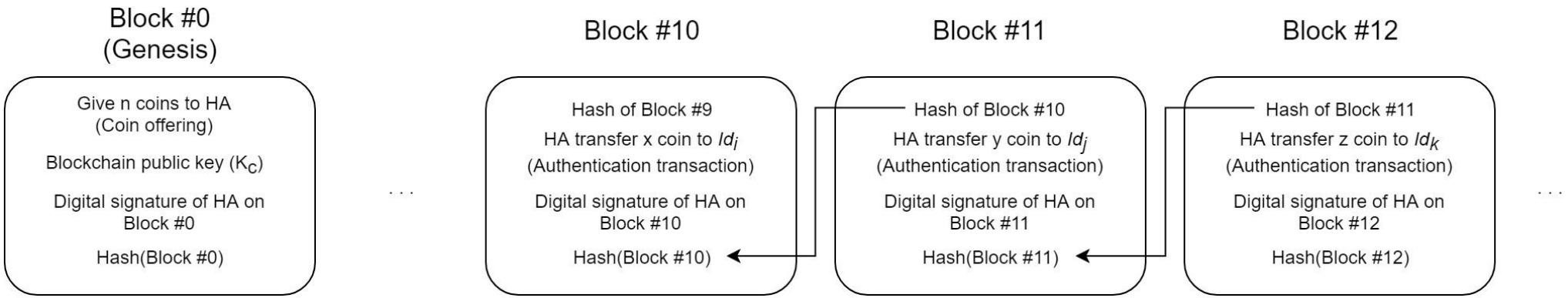
What is Blockchain?

The term blockchain is used as:

- A name for a data structure
- A name for an algorithm
- A name for a suite of technologies
- An umbrella term for purely distributed peer-to-peer systems with a common application area

Blockchain as a name for a data structure

When used as a name for a data structure, blockchain refers to data put together into units called blocks. One can think of these blocks much like pages in a book.These blocks are connected to one another like a chain, hence the name blockchain.



Blockchain as a name for an algorithm

When used as a name for an algorithm, blockchain refers to a ***sequence of instructions*** that ***negotiates the informational content of many blockchain-data-structures*** in a purely distributed peer-to-peer system

Blockchain as a name for a suite of technologies

When used to refer to a suite of technologies, blockchain refers to a **combination of the blockchain-data-structure, the blockchain-algorithm, as well as cryptographic and security technologies** that combined can be used to achieve integrity in purely distributed peer-to-peer systems, regardless of the application goal.

C. H. Lau, K. H. Yeung, and F Yan, “Blockchain-based Authentication in IoT Networks,” 2018 IEEE Conference on Dependable and Secure Computing.

Blockchain as an umbrella term for purely distributed peer-to-peer systems with a common application area

Blockchain can also be used as an umbrella term for purely distributed peer- to-peer systems of ledgers that utilize the blockchain-technology-suite.

Note that in this context blockchain refers to a purely distributed system as a whole instead of referring to a software unit that is part of a purely distributed system.

Provisional Definition of Blockchain

- This definition serves as an intermediate step toward a more complete understanding of the term:

- *The blockchain is a purely distributed peer-to-peer system of ledgers that utilizes a software unit that consist of an algorithm, which negotiates the informational content of ordered and connected blocks of data together with cryptographic and security technologies in order to achieve and maintain its integrity.*

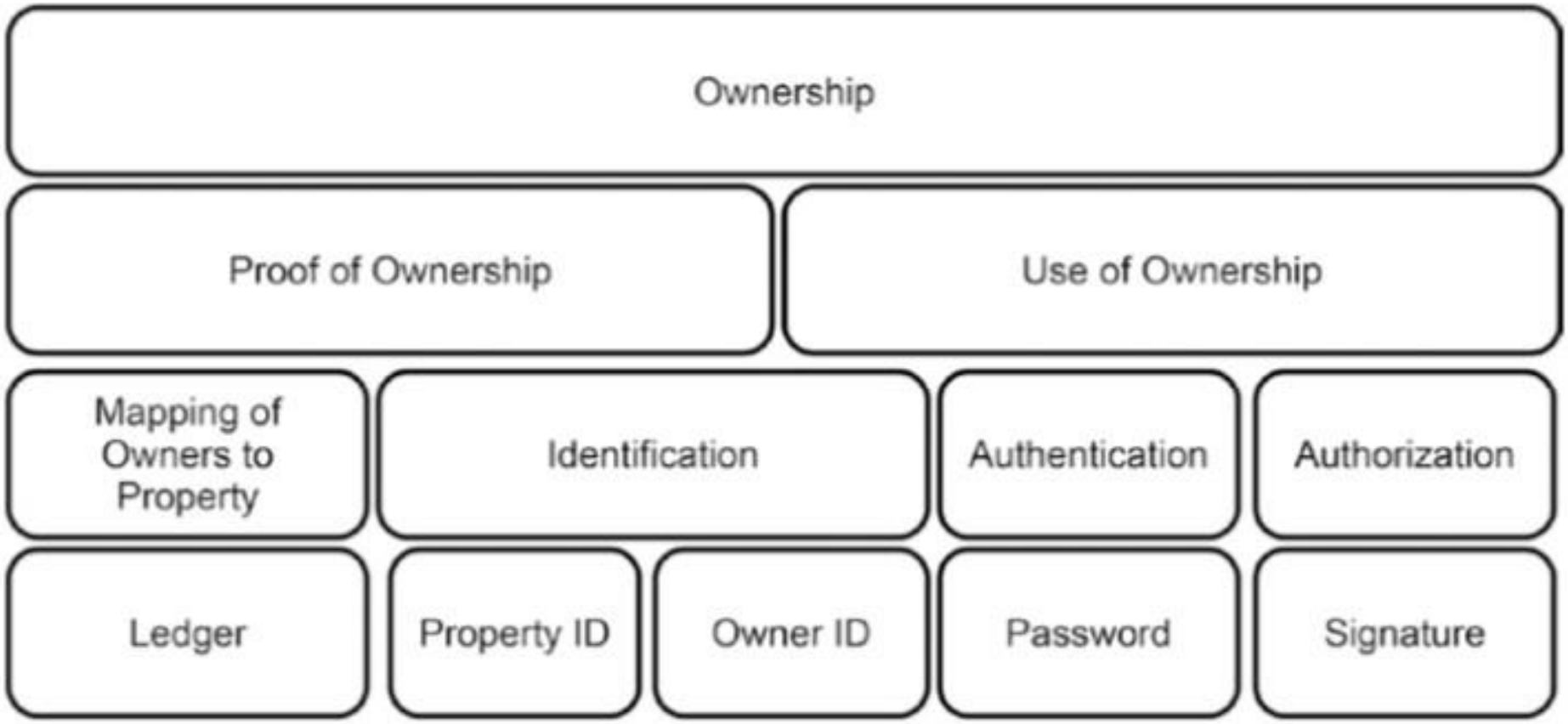
Understanding the Nature of Ownership

Foundations of Ownership

One can state that proving ownership involves three elements:

- An identification of the owner
- An identification of the object being owned
- A mapping of the owner to the object

Foundations of Ownership



Foundations of Ownership

The previous figure used three major security related concepts:

- **Identification**: Identification just means to claim to be someone by stating a name or any- thing else that could be used as an identifier.
- **Authentication**: The purpose of authentication is to prevent someone from claiming to be someone else. Authentication means verifying or proving that you really are who you claim to be. This proof can be provided by something you have or something you know that can serve as proof that you really are who you claim to be.
- **Authorization**: Authorization means granting access to specific resources or services due to the characteristics or properties of one's identity1. Authorization is the consequence of both a successful authentication and evaluation of one's characteristics or rights.

Ledger of Ownership

Ownership and Blockchain

The relation between managing ownership with a ledger and the blockchain is summed up as:

- An individual ledger is used for maintaining information about ownership, which is equivalent to one blockchain-data- structure storing ownership-related data.
- The individual ledgers are stored on the computers (nodes) of a peer-to-peer system.
- The blockchain-algorithm is responsible for letting the individual nodes collectively arrive at one consistent ver- sion of the state of ownership on which the final verdict is based.
- Integrity in this system is its ability to make true statements about ownership.
- Cryptography is necessary for creating a trustworthy means of identification, authentication, and authorization and ensuring data security.