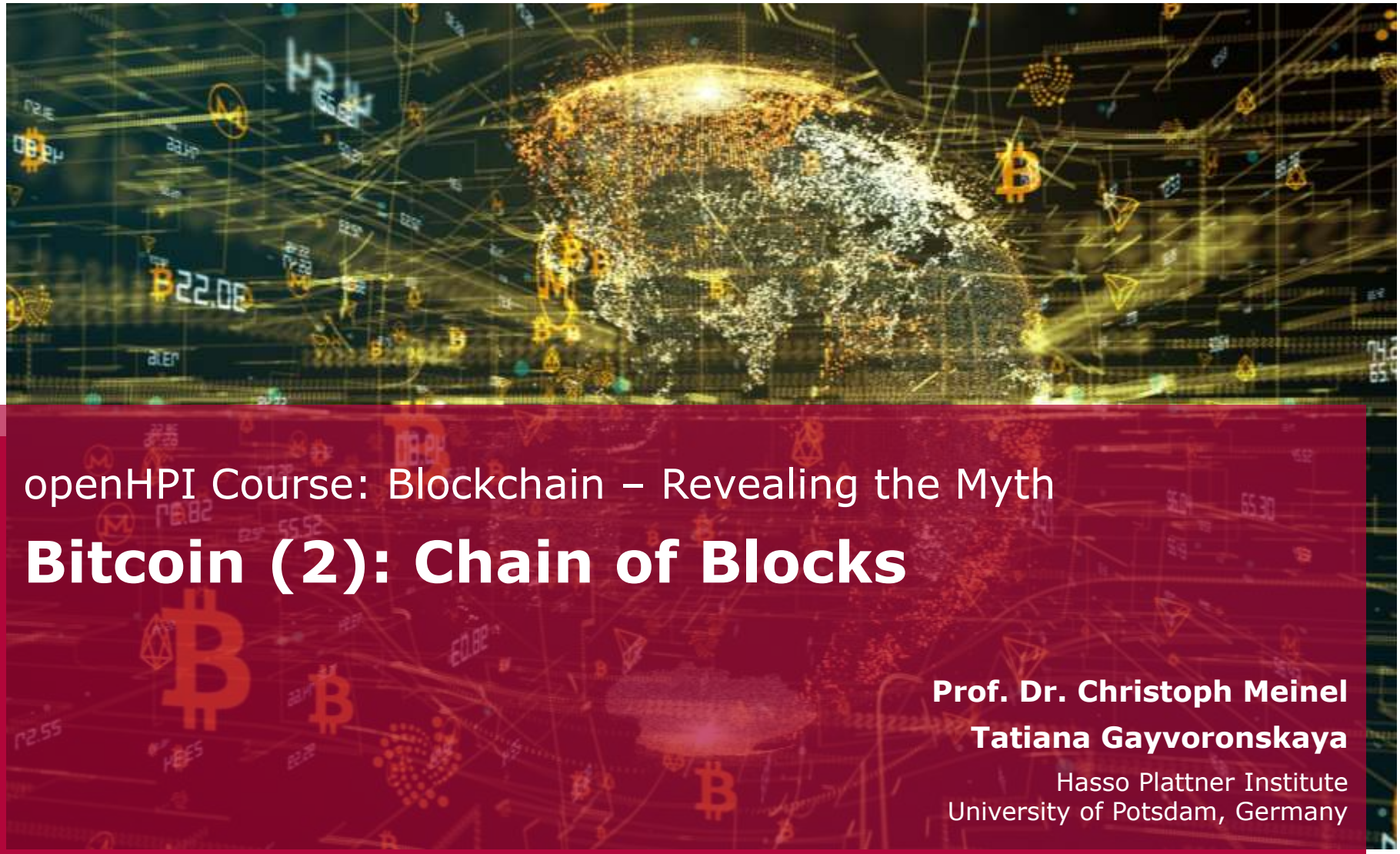


openHPI Course: Blockchain – Revealing the Myth

Bitcoin (2): Chain of Blocks

Prof. Dr. Christoph Meinel
Tatiana Gayvoronskaya
Hasso Plattner Institute
University of Potsdam, Germany

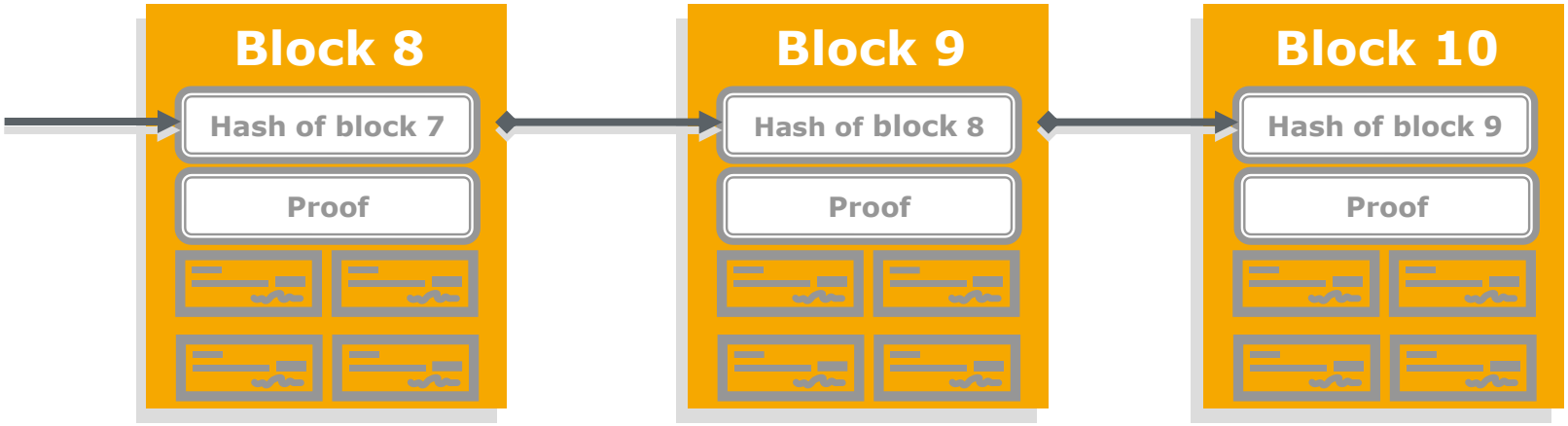
Majority Decision by the Longest Chain

Review:

- We have **no central authority** and **no trusted third party**
- All transactions are **publicly announced**
- Users are **pseudonymous**
- We have a system for participants to **agree on a single history of the order** in which transactions were received
- The majority decision is represented by the **longest chain**, in which the **greatest proof-of-work effort** is invested
- To prevent **double-spendings** a peer-to-peer distributed **timestamping** is used that generates a **computational proof of the chronological order** of transactions
- Transactions that are **computationally impractical** to reverse protect sellers from fraud

Block of Transactions
Chain of Blocks

- The system would be far **too slow** if, **every time a new transaction arrives**, users have to solve a **computationally complex task** and to cryptographically link (by generating hashes) all transactions individually with each other
- To make the process **more efficient**, each user first **collects received transactions** into a list **"block"** of a specified size
- Then work on finding a difficult **proof-of-work** for this **new block**. This **proof** together with the **hash** of the previous block are added to the new block

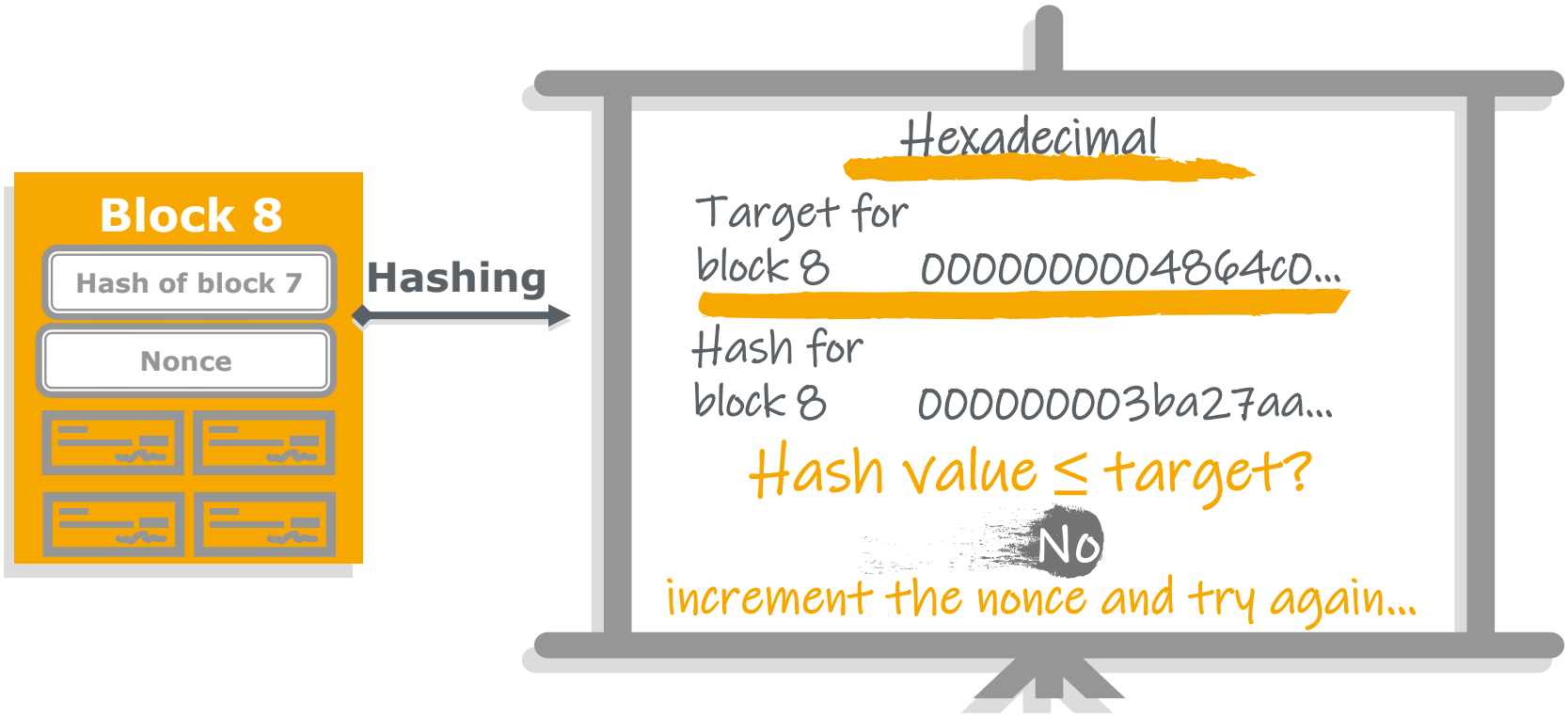


Proof-of-Work
Nonce

- Like previously mentioned, the **work** consists of a **computational task** demanding a huge **CPU effort**
- **Computational task** consists of simply trying out a number of hash values to **find a value** that corresponds to a given **target**
- To do this user computes the hash of the **block together with an arbitrary number** called **nonce** and checks whether the resulting number is **below given target**
  - **"nonce"** is a combination of symbols that is **only used once** in the respective context

Proof-of-Work
Incrementing the Nonce

- If the **resulted hash isn't below target**, user keeps trying by **incrementing the nonce** in the block until a value is found that gives the block's hash the **required zero bits**

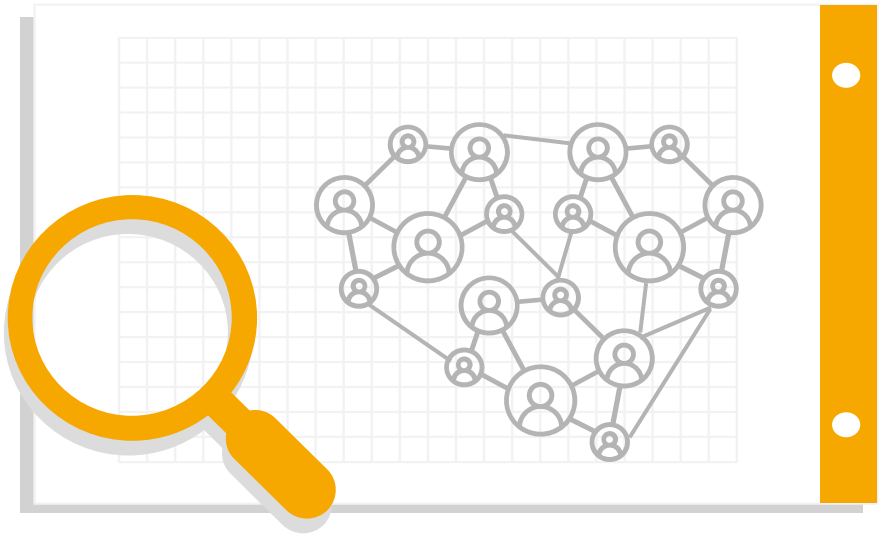


Proof-of-Work Target

- If the proof-of-work is to easy to solve, a large amount of **forks** would occur
- A gap between blocks of **10 minutes** is considered safe for the stability of the network
- The solution implemented in bitcoin is to **vary the proof-of-work difficulty (adjust the target)** according to the change in the hashrate of the network
- **Target** adjusts **every 2016 blocks** (roughly two weeks) such that **10 minutes** are required for the creation of a new block
- If the computing power of the entire network **increases** (or **decreases**) and the 2016 blocks are found in less (or more) than two weeks, then the level of difficulty is **raised or lowered** accordingly

What is the Incentive to Stay Honest?

- What is the **incentive** for user's to encourage them **to stay honest** and **expend the electricity** by the CPU effort?
- It would definitively be wrong to assume that all users **act rationally and follow the set rules** strictly only to have the option of doing **without a third party** and using a **peer-to-peer electronic cash system**
- It is time to take a **closer look at our network**



Chain of Blocks
Proof-of-Work

Summary

- To make the process of cryptographically linking more efficient, all user's first **collect transactions received** into a **block**
- Then they need to **generate a hash of the block** and work on finding a difficult **proof-of-work** for this **new block**.
  This **proof** together with the **hash** of the previous block are added to the new block
- The **proof** is represented by a **nonce** which, in combination with a block, after a hash function, gives a value that's **below current target**