CS 394B Introduction

Marco Canini

This Class

- Course is a combination: classes in a flipped classroom style and paper presentations/discussions

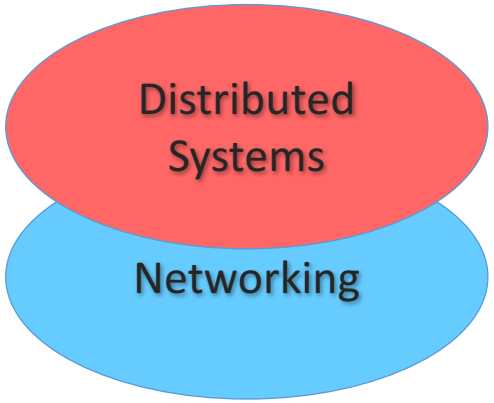
- Learn technical aspects of blockchain technologies and distributed consensus
- Have the conceptual foundations to engineer secure software that interacts with the blockchain
- Be able to integrate ideas from the blockchain in their own projects

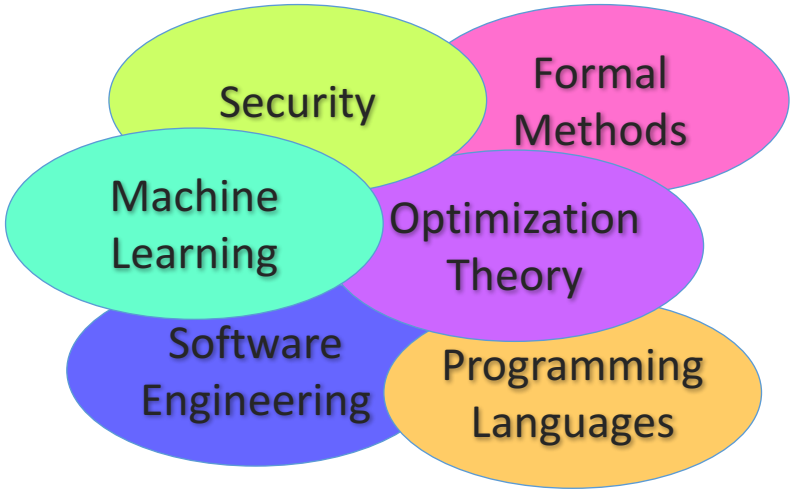
- Comprehend and critique relevant research papers in the area of blockchain systems
- Present research ideas both orally in a concise way and within the allotted time as well as in writing
- Defend the research approach, design decisions, and the evaluation methods in a discussion
- Moderate a discussion after a research presentation

About the Instructor

- Marco Canini
  - Assistant Professor at KAUST since Aug ’16
  - https://mcanini.github.io
- Research interests span
  - Distributed and Networked systems in the context of cloud computing, large-scale data analytics, and machine learning
- Head of SANDS Lab
  - Software-defined Advanced Networked and Distributed Systems Laboratory

My research





I design, build, measure and analyze large-scale networked systems that span multiple autonomous, potentially untrusted entities

Goal: *Discover and apply fundamental principles and valuable knowledge on how to build scalable, dependable and future-proof systems, worthy of society's trust*

Challenges

#1 Challenge: Complexity

Hard to reason about behavior as systems scale to large numbers of components and users

- Poorly understood connections
- Need predictability to ensure scalable performance, reliable operation, etc.

Systems Approach

- Formulate problem
- Get idea
- Build prototype
- Measure & analyze
- Adjust prototype … repeat previous step

Principles of system construction

- *modularity, hierarchy, layering, abstraction, end to end*

New approaches are needed

Systems based on design decisions made in the last decade can hardly cope with today's scale, volume or velocity, let alone the future

We need **new techniques, designs and solutions**:

- Improve performance by at least **10x**, in some cases **100x**
- Ensure **predictability of performance** and **high reliability**
- **Lower complexity** of managing **large-scale systems** and processing **big data**

SANDS Lab Vision

Make it easy to produce and manage key networked systems that are worthy of society's trust and achieve specific objectives:

- High performance and scalability
- High dependability and future-proof
- Low power
- …

We build **prototypes** that directly **improve the lives of real users**

We learn general **principles and lessons** of what **works in practice**

Example: Dynam-IX

- Dynamic Interconnection eXchange

- https://dynam-ix.github.io/

What About You?

- Please introduce yourself!
- Why are you in this class?
- How can we make it a very useful one?

About this class

Course Schedule

- Webpage: http://web.kaust.edu.sa/Faculty/MarcoCanini/classes/CS394B/S18/
  - Piazza: https://piazza.com/kaust.edu.sa/spring2018/cs394b
- Meetings
  - 4PM – 5:30 PM (Sun for lectures and discussions)
- Pay attention to the online announcements and schedule
  - On average, one meeting per week
  - Makeups will be added on a need-to-add basis, typically Wed

Prerequisites

- CS 240 (Computing Systems and Concurrency)
  - Basics of OS organization, threads, memory management, file systems, scheduling, networking, etc.
  - Equivalent course of CS 240 is acceptable as well
- Good programming skills
- Distributed systems – helpful
- Cryptography – helpful
- Probability – helpful

Flipped classroom

- Course lectures are online
  - We follow the course Bitcoin and Cryptocurrency Technologies by Arvind Narayanan, Joseph Bonneau, Edward Felten and Andrew Miller http://bitcoinbook.cs.princeton.edu/
- You must watch the video material before the meeting
- We go deep into each topic during meetings
- Send me questions in advance regarding the material

Textbook

- Textbook not required but can be very helpful
- Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, **Bitcoin and Cryptocurrency Technologies**
- A pre-publication draft of the book is available for download on the website: http://bitcoinbook.cs.princeton.edu/

Course Requirements

Paper Reviews	15%
Paper Presentation	15%
Active Participation	5%
Assignments	25%
Project	40%
Checkpoint #1: initial proposal	5%
Checkpoint #2: midterm progress report	10%
Presentation	10%
Final report	15%

Paper Reviews

- Paper reviews account for 20% of the total grade
- 6-8 summaries to write
- What goes in a good summary?
  - Highlight strengths
  - Highlight weaknesses
  - Describe the entire paper in 3-5 sentences

Papers

- Read critically!
- Is the problem real?
- What is the solution's main idea (nugget)?
- Why is solution different from previous work?
  - Are assumptions different?
  - Is workload different?
  - Is problem new?
- Does the paper (or do you) identify any fundamental/hard trade-offs?
- Do you think the work will be influential in 10 years?
  - Why or why not?

Paper Reviews

- Reviews must be submitted electronically 24 hours before the class
  - Send the review via Piazza as private note to Instructors
- You can miss at most two without any penalty
  - Each missing one beyond that will result in 25% decrease in grade for this segment
  - Meaning, missing six or more will result in 0% for the “Paper Reviews” segment of your grade
- Reviews are peer-reviewed
  - Another student will read and give constructive feedback on your review
  - The goal is to make better reviews
  - Good feedback will be considered positively during grading
- Read (if you haven’t already!)
  - How to Read a Paper by S. Keshav
  - How to read a research paper by Michael Mitzenmacher
  - Writing Reviews for Systems Conferences by Timothy Roscoe

How to Review

- You will see a section for describing a paper summary, its strengths, its weaknesses, and detailed comments
- In the summary section, please directly address:
  1. What problem the paper is addressing (1-2 sentences or bullets)
  2. The core novel ideas or technical contributions of the work (1-2 sentences or bullets). Put another way, what's the 30 second elevator pitch, or, five years from now, what should one remember about this paper?
  3. A longer description (3-5 sentences) that summarizes the paper's approach, mechanisms, and findings.
- For the other sections, please include 2-4 bullet points for the strengths and weaknesses, while a much longer exposition in the detailed comments.
- Remember to be constructive: don't only focus on the paper's shortcomings, but also on what it could have done differently or as the next steps. Imagine that you are having a conversation with the authors: What would you tell them?

Paper Presentation

- Each student must present at least two papers; possibly three
  - Paper presentation account for 15% of the total grade
- Two papers per class and 20-minutes presentation for each
  - Sharp 20 minutes; we start clapping!
  - Followed by discussion anchored by the presenters
- What should go in a useful presentation?
  - Motivate
  - Highlight the key ideas and insights; skip the details
- Lead the discussion
  - Go through the strengths and weaknesses from the paper review

Presentation Guidelines

Your oral description of the paper should follow a much similar format:

1. Title, authors and institutions, conference/journal
2. Problem
3. Core ideas
4. Descriptive summary and main results
5. Strengths
6. Weaknesses/limitations
7. Further discussion, including proposals for follow-up work

Paper Presentation

- Email your slides to the instructor 24 hours before the class
- Prepare early
- Practice a lot
- Also, read
  - How to Give a Bad Talk by David A. Patterson
  - Pointers for Leading Paper Discussions by Randy H. Katz

Participation

- Attend all meetings
  - Can miss at most two with legitimate reasons
- Read all the papers and participate
  - Ask questions!
- Send questions on video lectures (in advance)

Assignments

- 3 programming assignments
  - To be done individually; due 11:59pm on due date

- 20/2 Assignment 1 due
- 6/3 Assignment 2 due
- 21/3 Assignment 3 due

- No extensions given

Project

- The biggest component of this course
  - Pick an interesting open problem. Why is it important?
  - What has already been done? Why are they not enough?
  - Develop a hypothesis about how you'd improve it
    - Intuitively, why will your approach work?
  - Build a substantial prototype
    - Experiment, measure, and compare against the state-of-the-art
- Aim at producing a conference/workshop-quality research paper
  - Can be related to your research topic but it is expected to be distinct!

Projects

- The final project accounts for 40% of total grades
  - Done in groups of 2 students. Find your peers!
- What can and cannot be a project?
  - Just surveys are not allowed. In fact, each project must include a survey of related work and background
  - An ideal project should answer the questions you asked during paper reviews and points you cared about for presentations
  - Measurements of new environments or of existing solutions on new environments are acceptable upon discussion
- Will cover projects in more depth in the next meeting!

How to Approach it?

1. Find a problem and motivate why this is worth solving
2. Survey background and related work to get a sense of your (friendly!) competition
   - Might require you to go back to the first step
3. Form/update your hypothesis
4. Test your hypothesis
   - Go back to 3 until you are happy
5. Present your findings in a presentation and in writing
   - Discuss known limitations

Milestones

Date	Milestone	Details
ASAP	Form Group	Find like-minded students
14/2/18	Draft Proposal	Send your proposal by email
7/3/18	Finalize Proposal Checkpoint #1 (5%)	After a back-and-forth discussions with the instructor
21/3/18	Midterm Progress Report Checkpoint #2 (10%)	Should read like parts of a research paper
25/3/18	Midterm Presentations	Define and motivate a problem, survey related work, and form initial hypothesis and idea
13/5/18	Presentations (10%)	Present your findings in a presentation
16/5/18	Research paper (15%)	Submit a final report similar to the papers you read

Draft Proposal

- Two pages including references that *ideally* includes
  - What is the problem?
  - Why is it important to solve?
  - Any initial thoughts on what you want to do?
  - How would you evaluate your solution?
- Include team members
  - Meaning, form a group ASAP
- Schedule via email a 15-minute meeting to discuss

Read: The Heilmeier's Catechism

Finalized Proposal

- Two pages including references that **must** include
  - What is the problem?
  - Why is it important to solve?
  - Any initial thoughts on what you want to do?
  - How would you evaluate your solution?
- Approved by the instructor and agreed upon by you
  - Forms the basis of expectation

Midterm Presentation

- In-class short presentation over one day
  - This is to make sure you are making progress
- Must include
  - What is the problem?
  - Why is it important?
  - What are the most related work?
  - What's your hypothesis so far?
  - How are/will you evaluate it?

Final Presentation and Paper

- Presentation
  - It will follow a format similar to other presentations given in the class
- Research paper
  - The key part
  - Should be written similar to the papers you've read
  - Your goal is to do publishable quality systems research
  - Up to five "best projects" will be earmarked for expedited submission to a renowned conference, with the help of the instructor
  - How to Write a Great Research Paper by Simon Peyton Jones

Rough Outline

- Abstract
- Introduction (Highlight the importance and give intuition of solution)
- Motivation (Use data and simple examples)
- Overview (Summarize your overall solution so that readers can follow later)
- Core Idea (Main contribution w/ challenges and how you address them)
- Implementation (Discuss non-obvious parts of your implementation)
- Evaluation (Convince readers that it works and when it fails)
- Related Work (Let readers know that you know your competition!)
- Discussion (Know your limitations and possible workarounds)
- Conclusion (Summarize and point out future work)

Course Topics

- Introductory material (~10 video lectures)
  - basics of cryptography; Merkle tree
  - blockchain; distributed consensus
  - mining; incentives
  - proof of work; proof of stake
  - governance; economics
  - security
  - smart contracts; applications
- Advanced material
  - active research problems in the area, including attacks, network scalability, alternatives to proof of work/stake

Before We Move On…

- No extensions
- Everyone must watch lectures and read papers in **advance**
- Class meeting format
  - Quick summary by the instructor
  - Topic discussion and addressing questions
  - Presentation of one paper
  - Presenters lead discussions on the papers we've read and related topics

Developing a Cryptocurrency

Slide courtesy of Ittay Eyal

Bitcoin

2008: The Bitcoin white paper
2009: Reference implementation

Probably not this guy

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

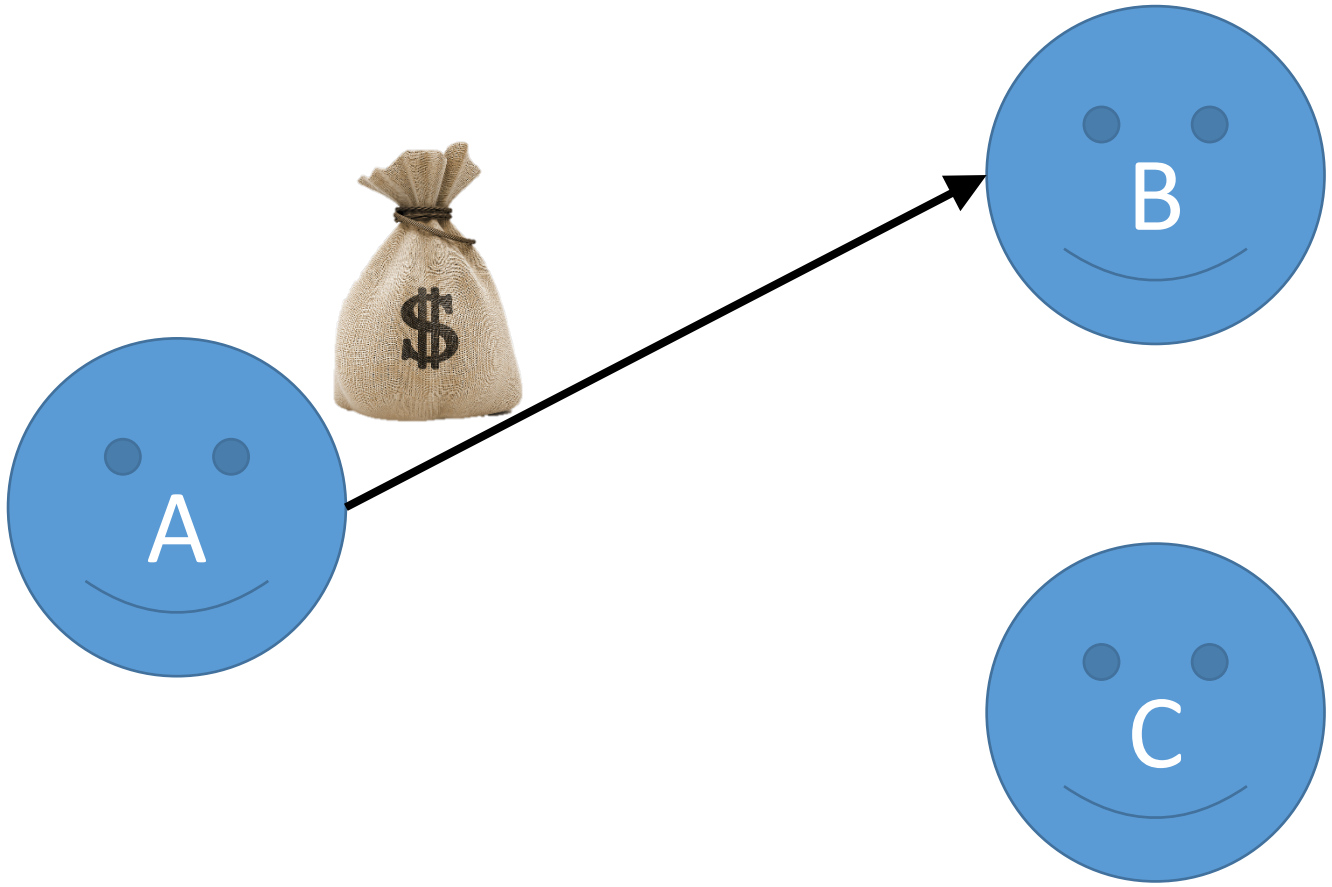
1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers

Slide courtesy of Ittay Eyal

Key Challenges



1. **No stealing: Only Alice can move her money**
2. Minting: Fair money creation
3. No double-spending: Alice cannot duplicate her money

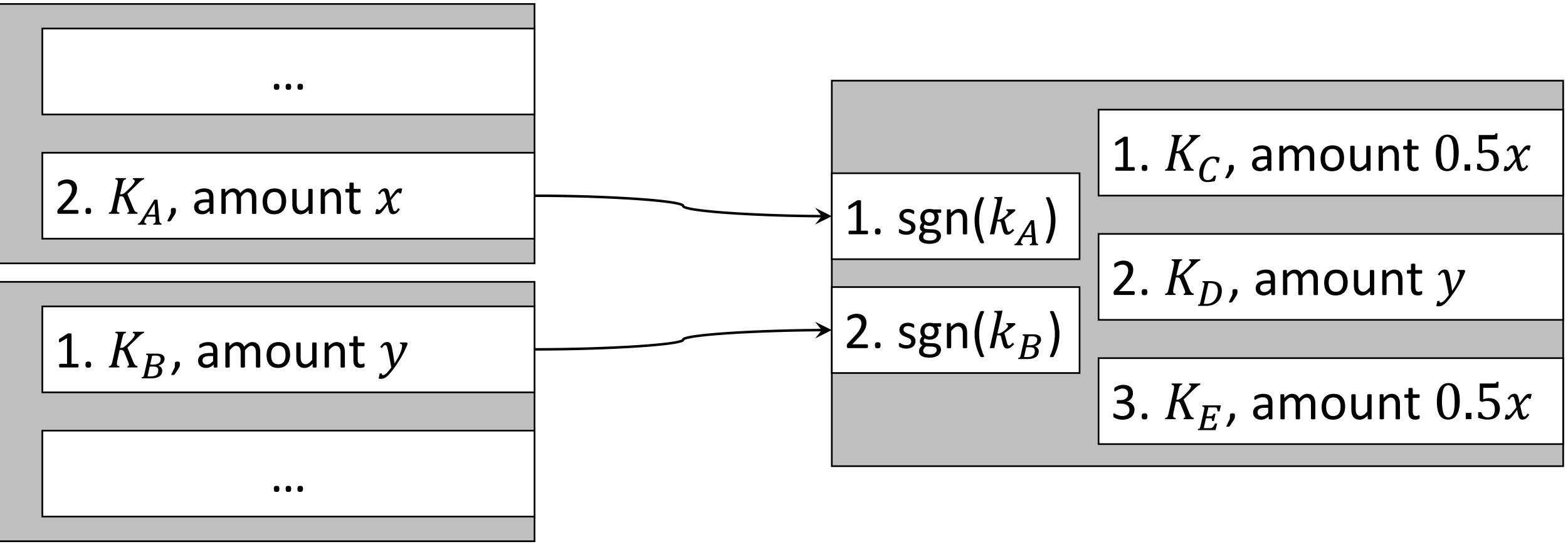
60 Seconds on Public Key Signatures

Alice generates key pair

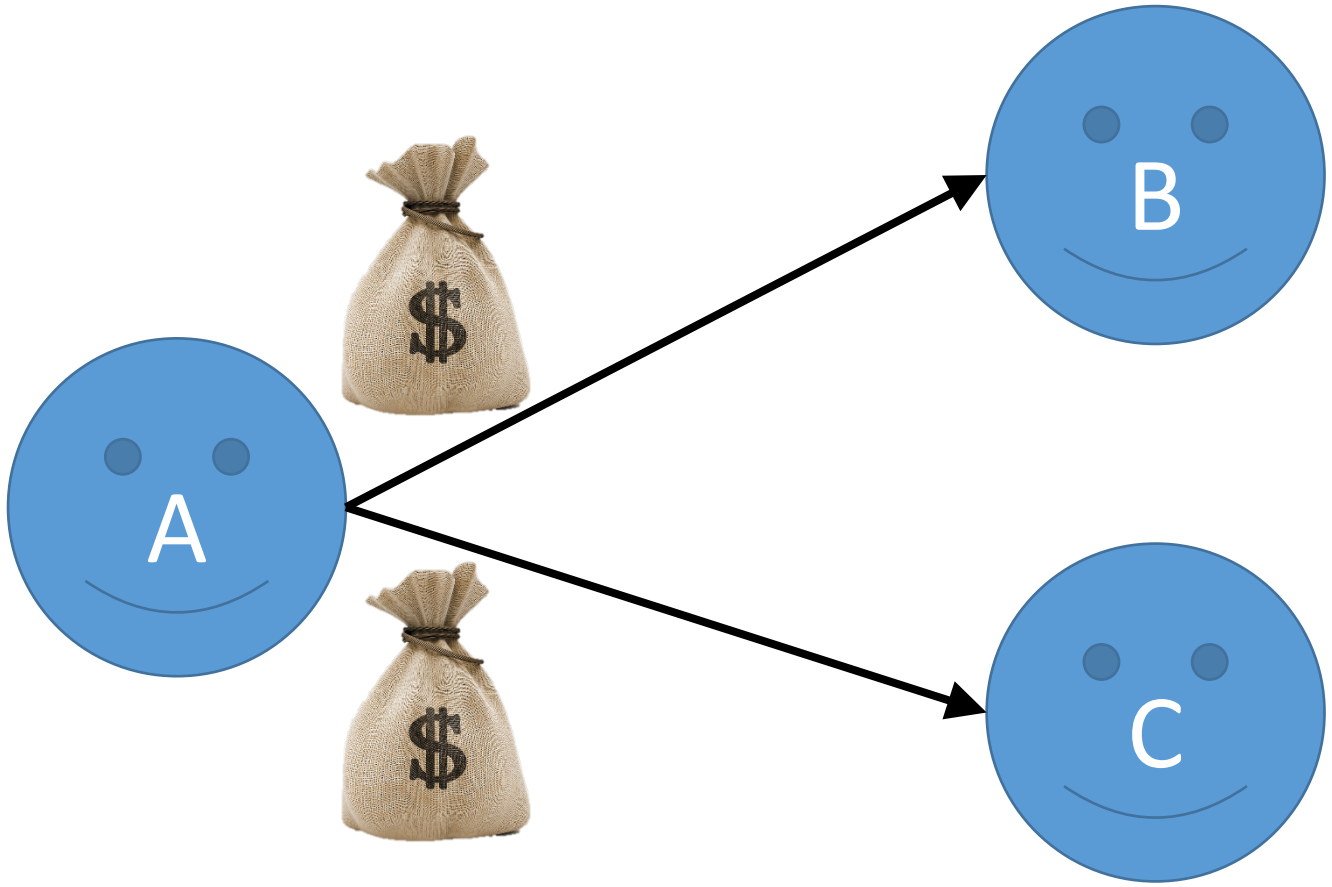
1. private key k_A, kept secret
2. public key K_A, published with ***public key infrastructure***

- Alice signs a message m with private key k_A, generating a signature s.
- Anyone can verify that s is a signature of m with key k_A given m and K_A.

Addresses and Transactions

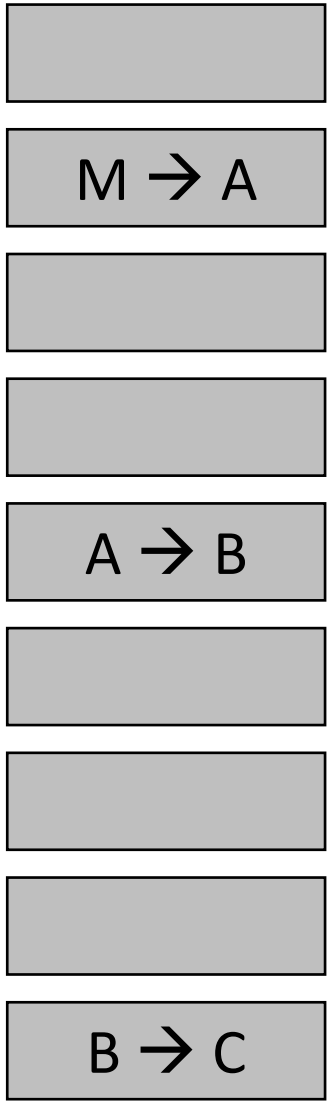


Key Challenges

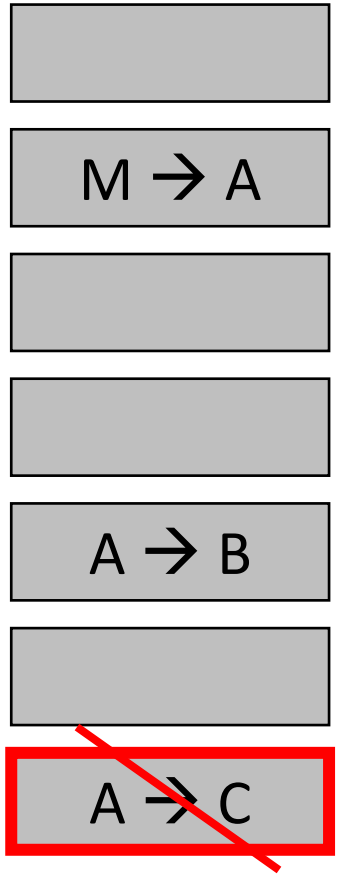


1. No stealing: Only Alice can move her money
2. **No double-spending: Alice cannot duplicate her money**
3. Minting: Fair money creation

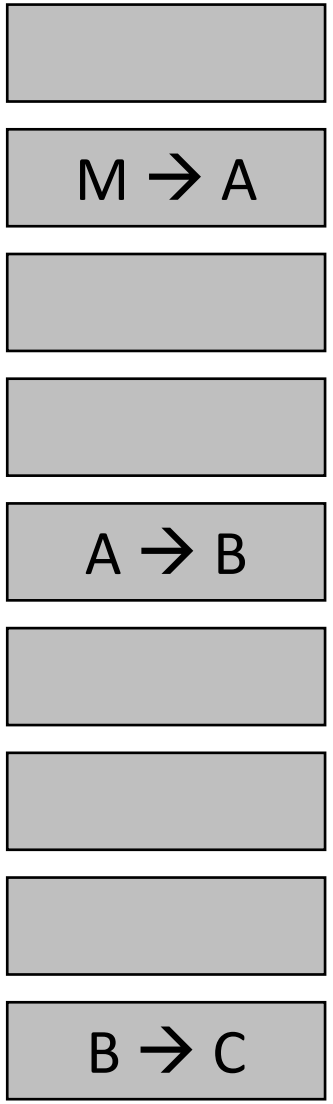
Global Ledger



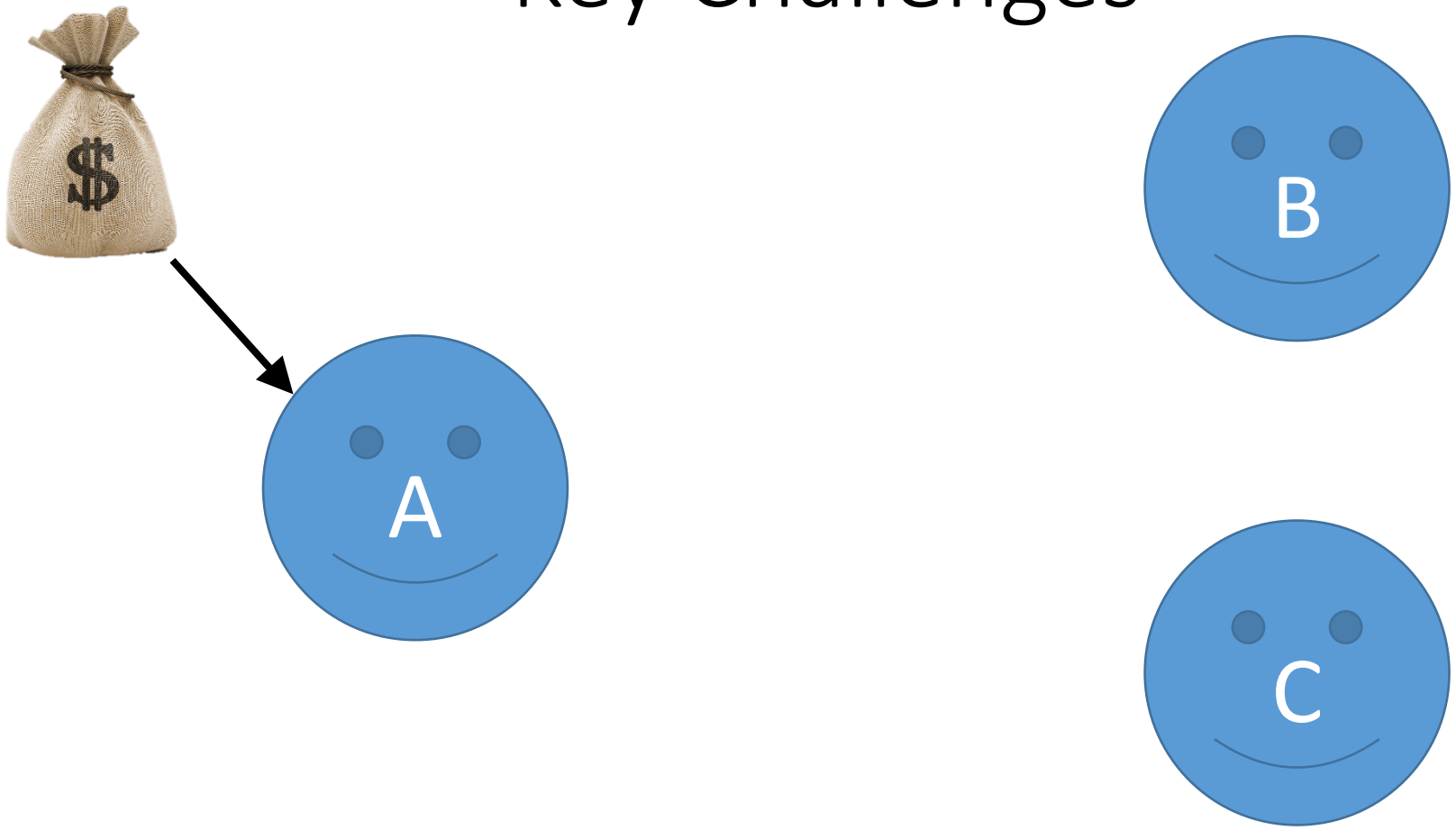
Global Ledger



Global Ledger

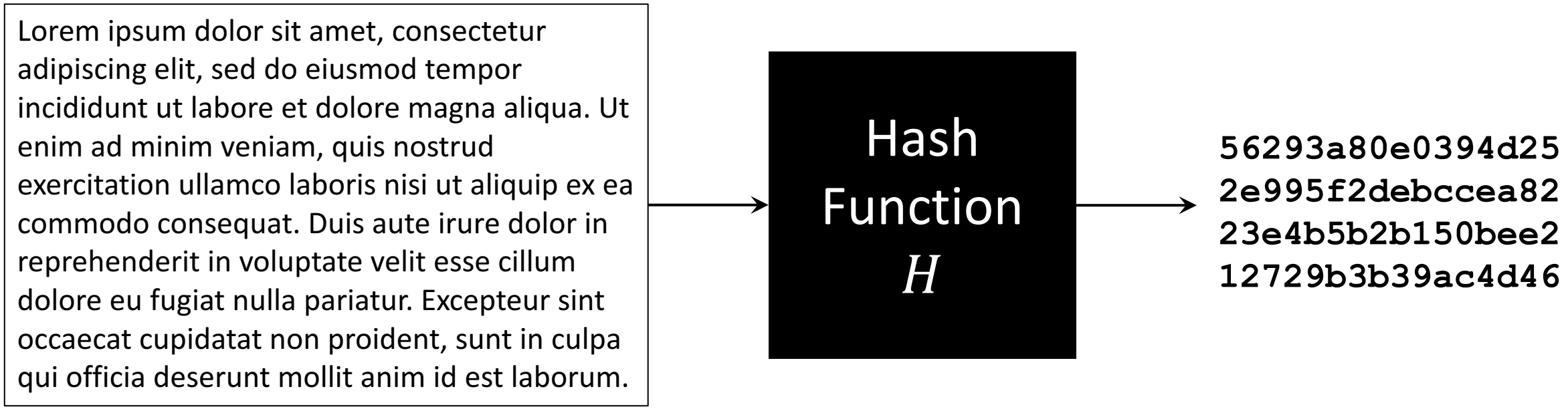


Key Challenges



1. No stealing: Only Alice can move her money
2. No double-spending: Alice cannot duplicate her money
3. **Minting: Fair money creation**

60 Seconds on Cryptographic Hashing



String input

256 bit number (for example)

Given a 256bit number h, one cannot find an input string that results in h faster than repeatedly guessing inputs x and calculating $H(x)$.

Mining – Minting for Proof of Work

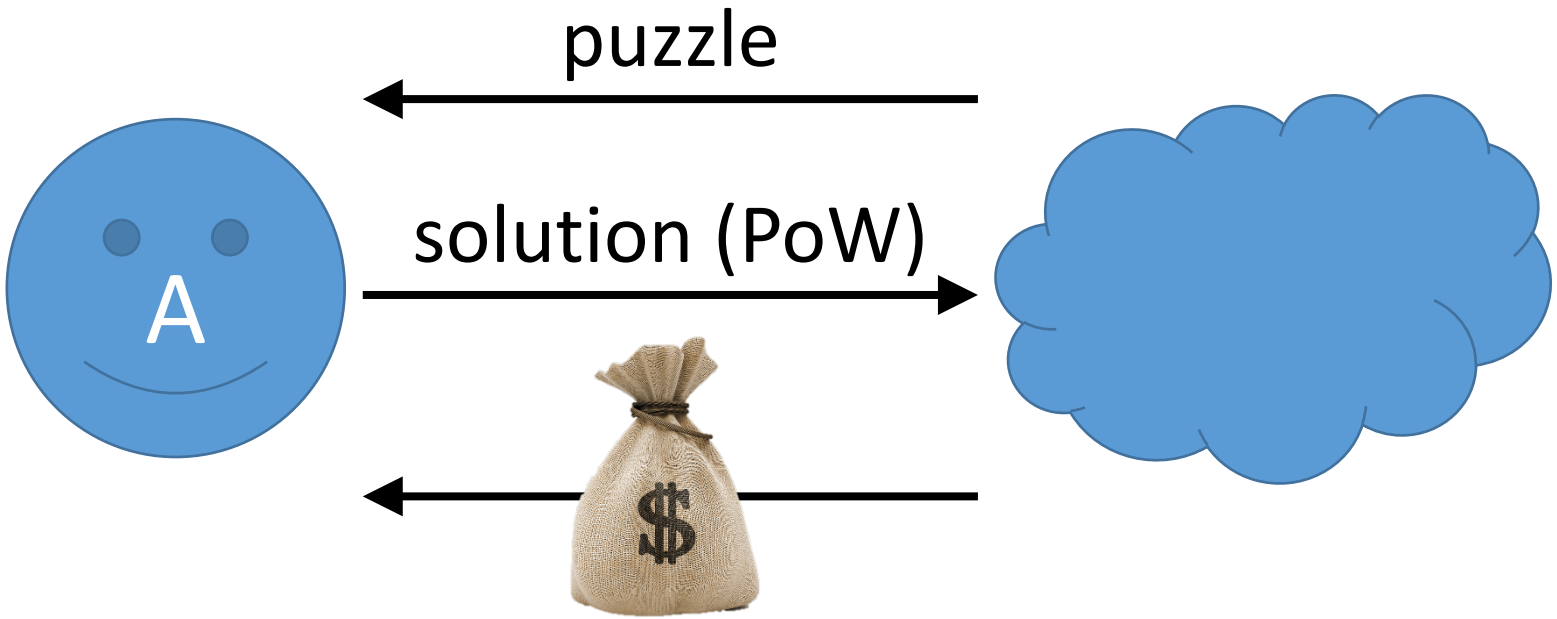
Computationally difficult puzzle:

$$\text{Find } x \text{ such that } H(x|y) < t$$

Solver guesses values for x until finding a valid one

- Different strings y for different puzzles
- The target t determines the difficulty, average time to solve

Mining – Minting for Proof of Work



Key Challenges

1. No stealing: Only Alice can move her money

   Cryptographic signatures
2. No double-spending: Alice cannot duplicate her money

   Global ledger
3. Minting: Fair money creation

   Mint for proof of work

Key Challenges

1. No stealing: Only Alice can move her money

   Cryptographic signatures
2. No double-spending: Alice cannot duplicate her money

   Global ledger
3. Minting: Fair money creation

   Mint for proof of work

Who runs the public key infrastructure?

Who maintains the public ledger?

Who gives money for puzzles?

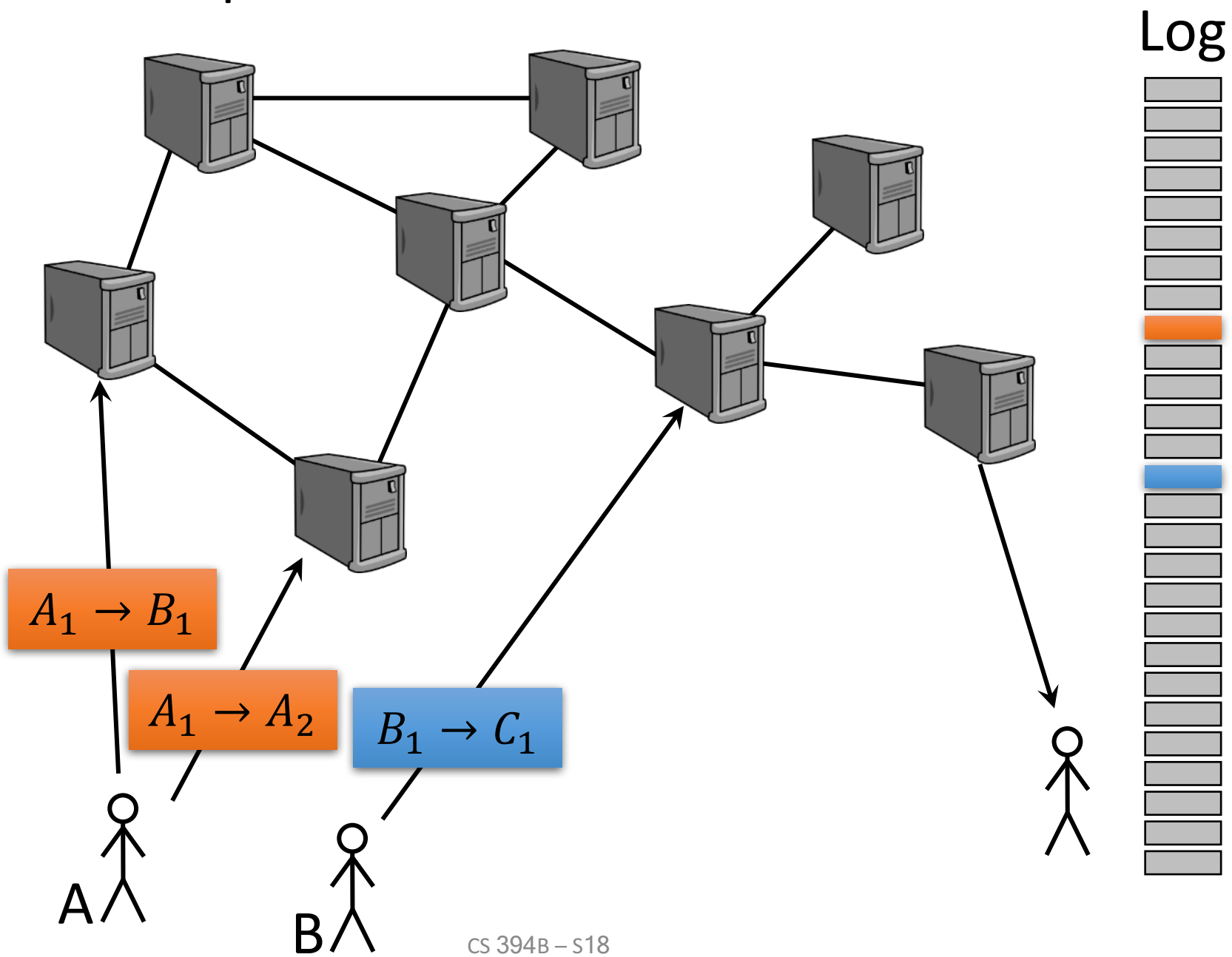
Can this be decentralized?

Replicated State Machine

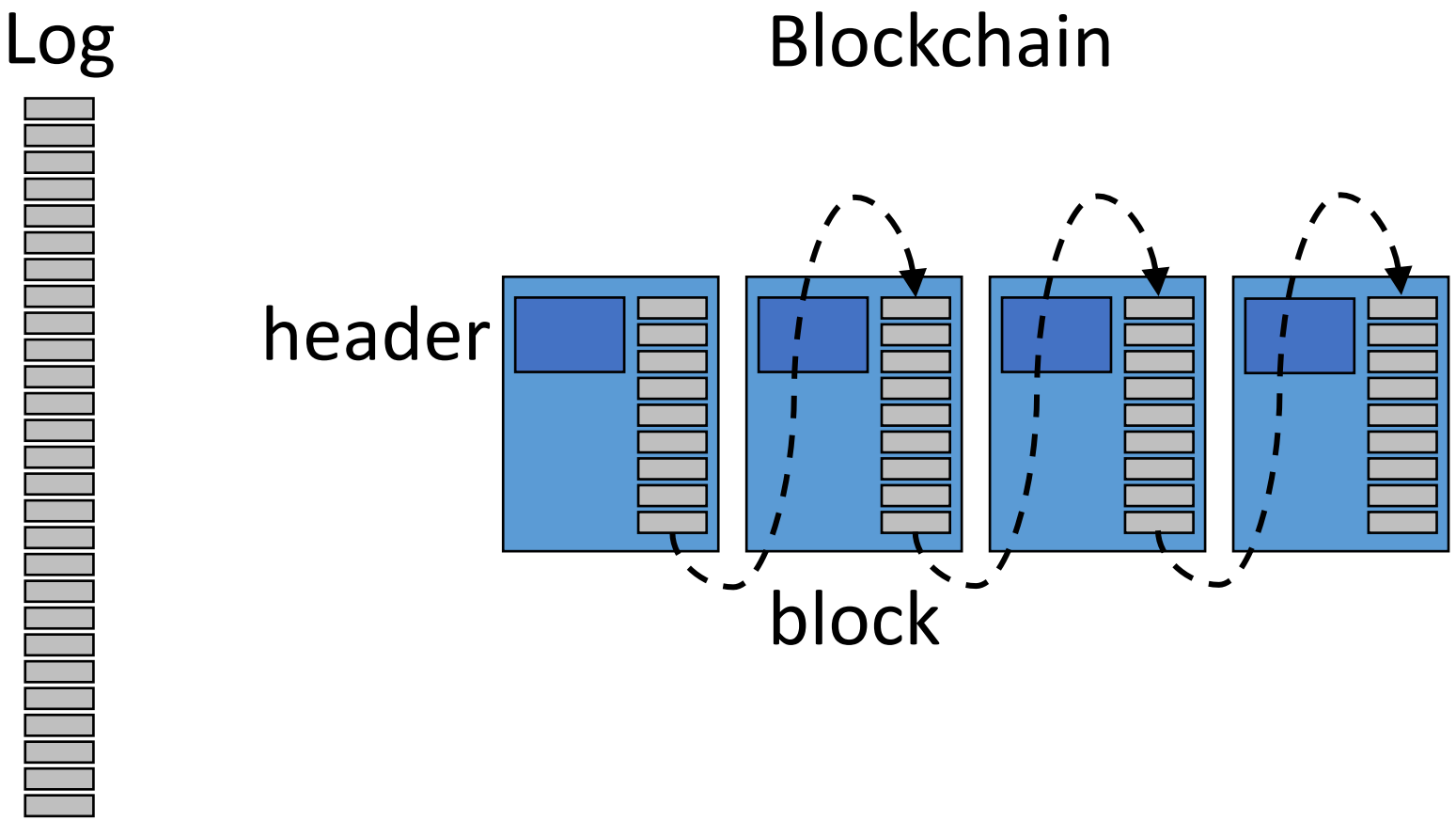
- Instead of one machine, use a ***replicated state machine***
- Multiple machines operate a single ledger, PKI, and mint fairly
- A subset can behave arbitrarily – aka ***Byzantine***

But who chooses the participating machines?

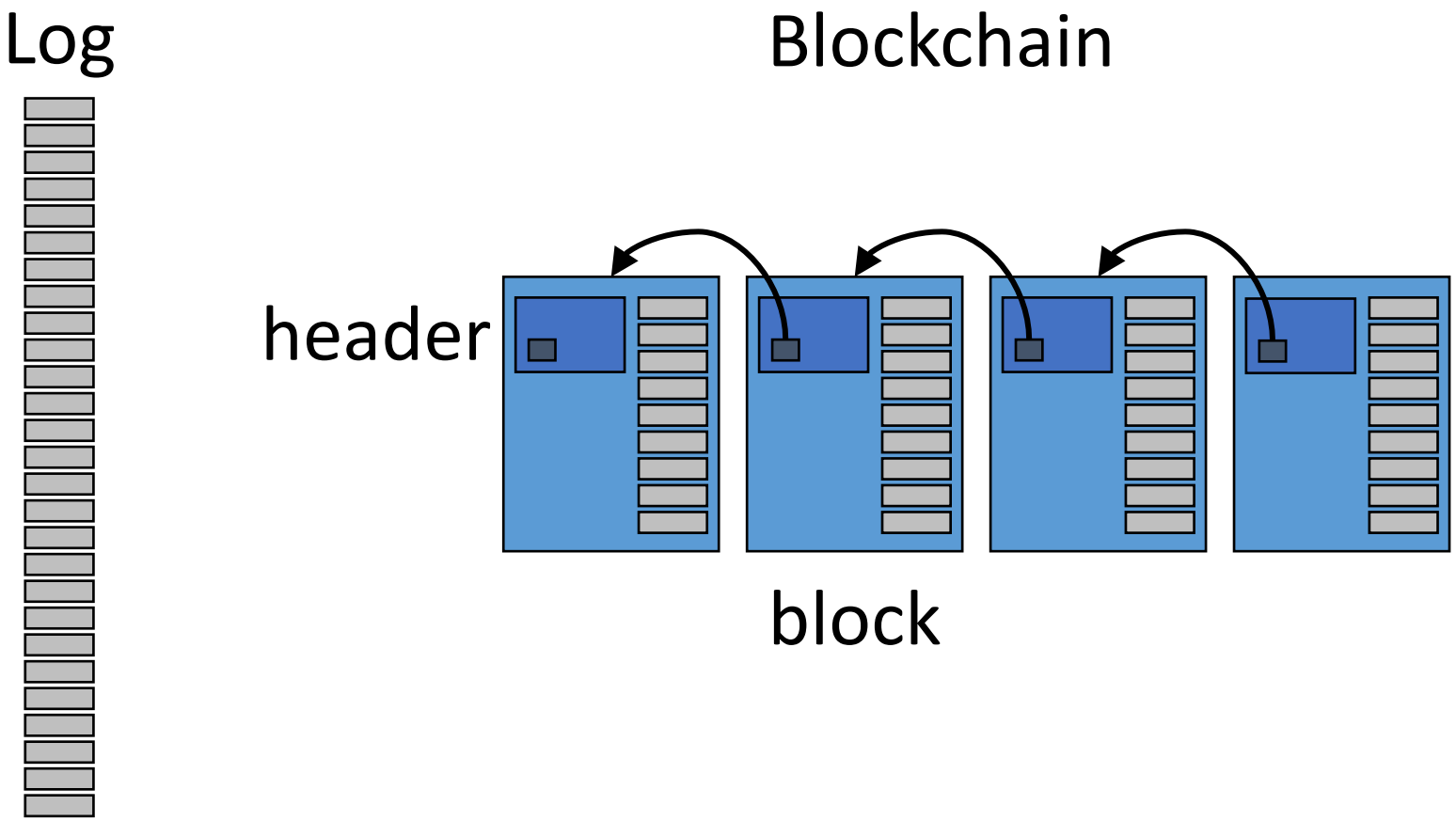
A Replicated State Machine



Nakamoto's Blockchain

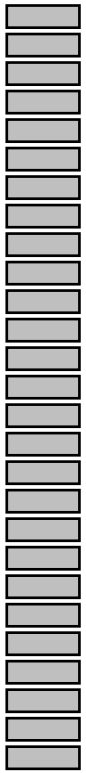


Nakamoto's Blockchain

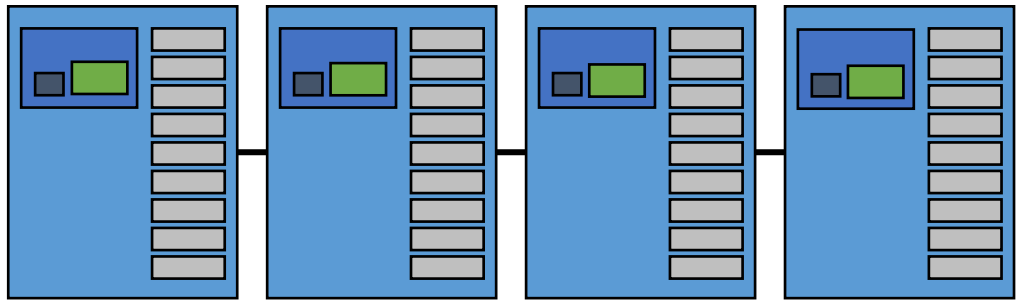


Nakamoto's Blockchain

Log



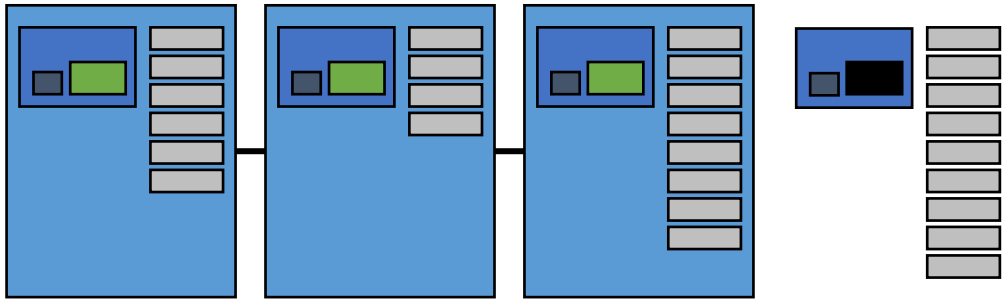
Blockchain



hash() < target*

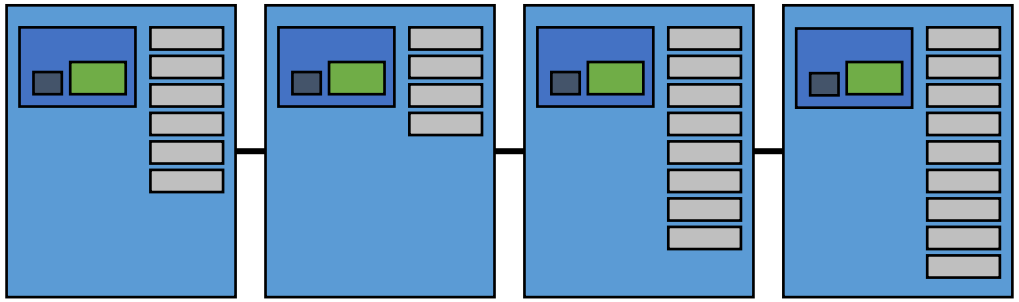
* *target*: a deterministic function of previous blocks

Nakamoto’s Blockchain



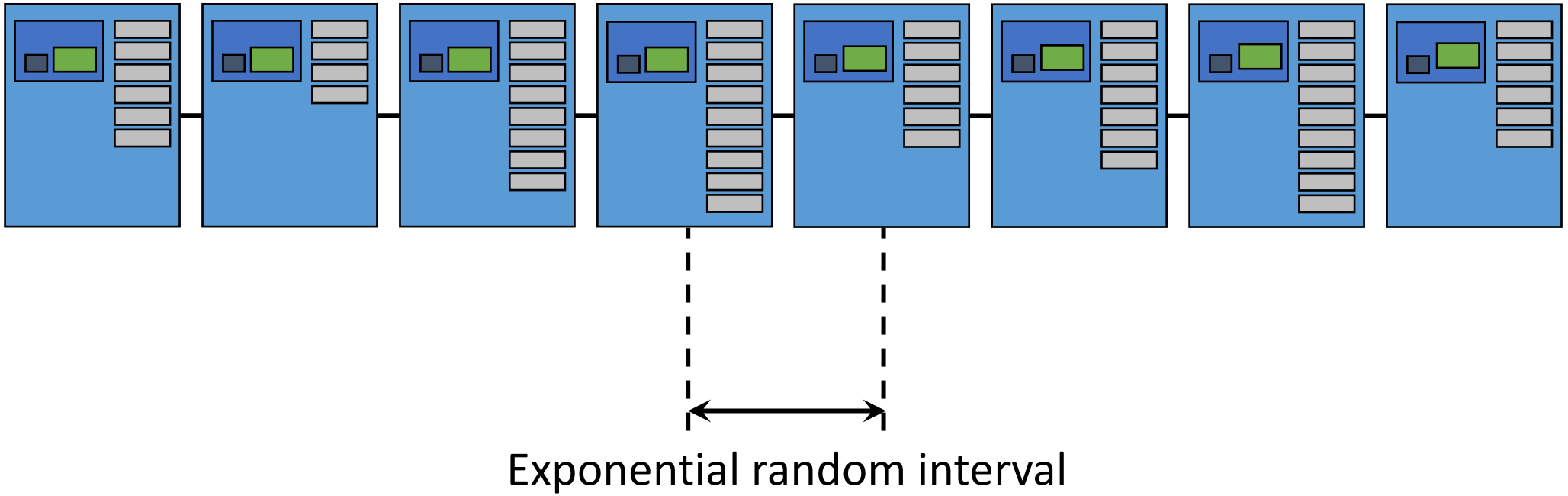
Slide courtesy of Ittay Eyal

Nakamoto's Blockchain



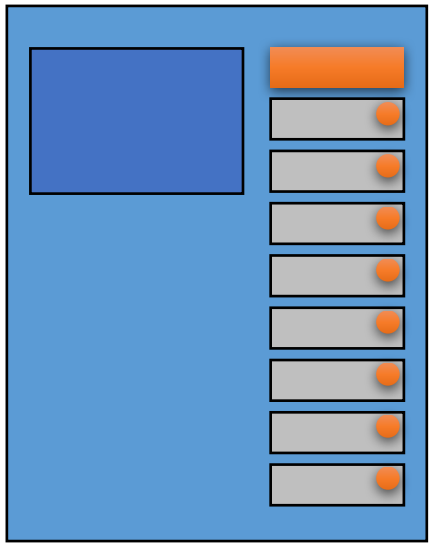
Slide courtesy of Ittay Eyal

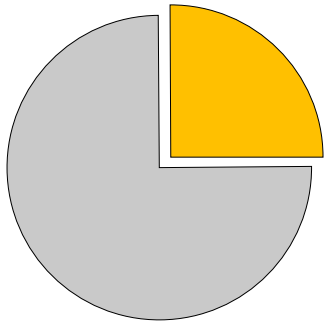
Nakamoto's Blockchain

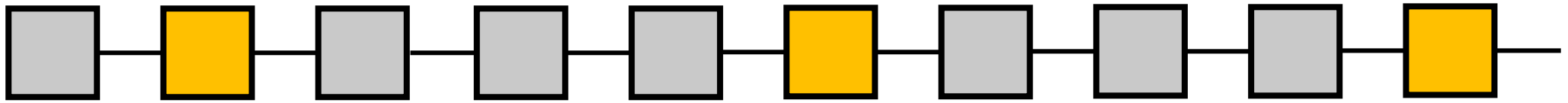


Incentive for Mining

- **Internal** Prize:
  - **Minting**
  - **Fees**

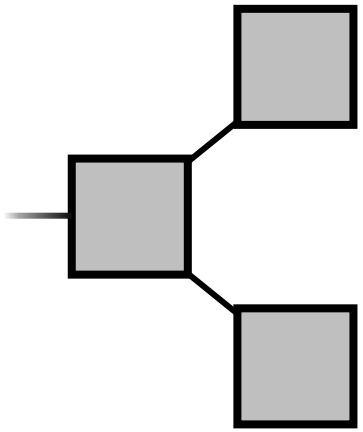






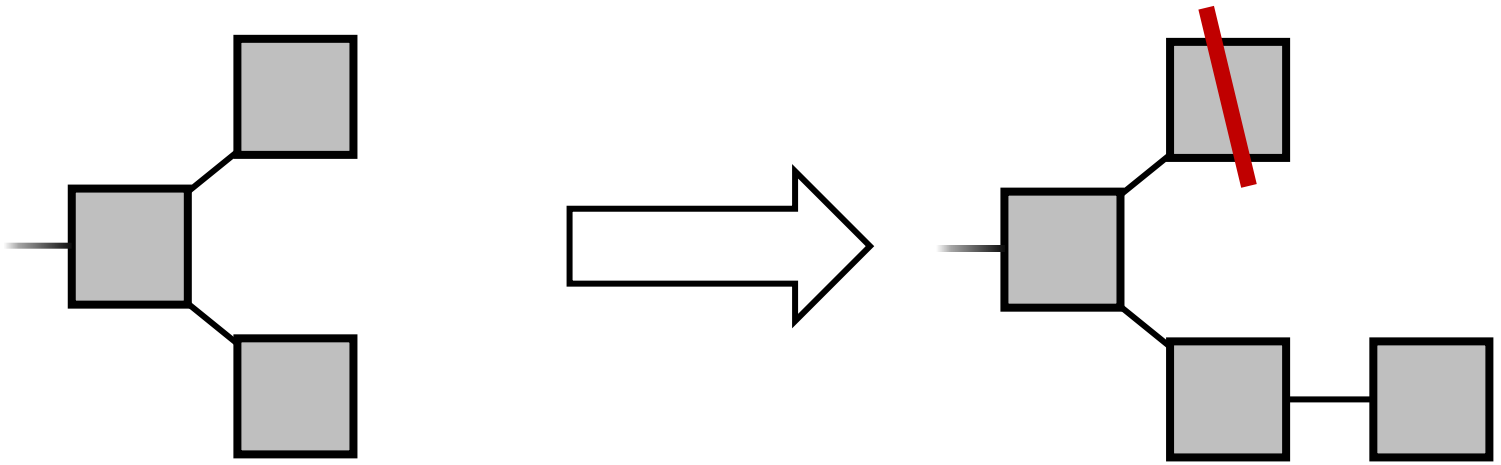
Wins proportional to computation power

Forks



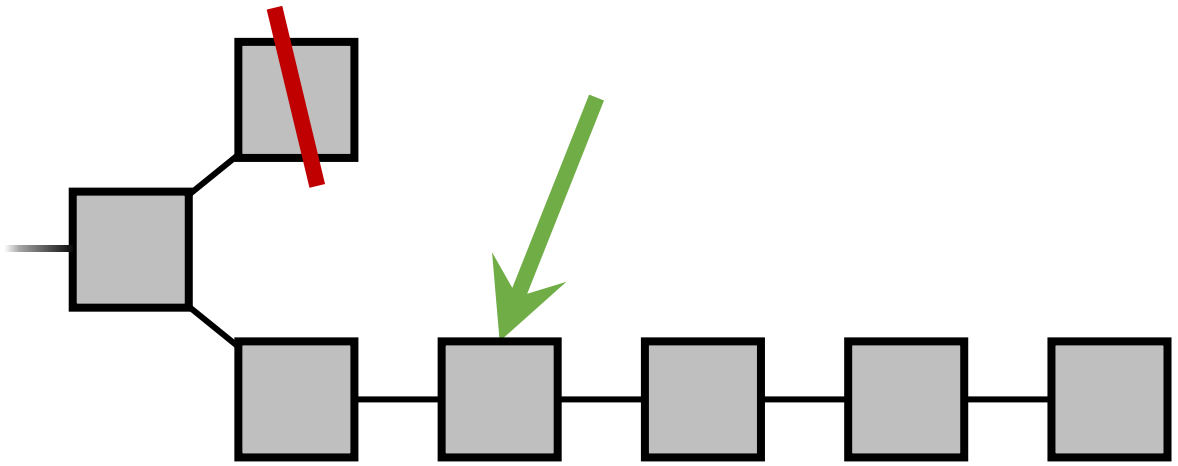
- Natural in a distributed system

Fork Resolution



- **Longest** chain wins
- Transactions are reverted
- Double-spending a threat

Fork Resolution



A transaction is **confirmed** when
it is **buried** deep enough

Key Challenges

1. No stealing: Only Alice can move her money

   Cryptographic signatures
2. No double-spending: Alice cannot duplicate her money

   Global ledger
3. Minting: Fair money creation

   Mint for proof of work

Decentralized

Next...

- Follow lecture 1
- Read Bitcoin: A Peer-to-Peer Electronic Cash System

- We meet on Wednesday