Topic 3.4: Bitcoin vs. Ethereum

Two Design Philosophies

Joerg Osterrieder

Digital Finance

2025

Learning Objectives

After completing this topic, you will be able to:

1. **Compare** the fundamental design philosophies of Bitcoin and Ethereum
2. **Explain** the technical differences: consensus, scripting, and supply models
3. **Define** what smart contracts are and why they matter
4. **Analyze** the security vs. flexibility tradeoff in blockchain design
5. **Evaluate** which platform suits different use cases

Key Insight

Bitcoin and Ethereum are not competitors – they are different tools designed for different jobs.

Prerequisites: Blockchain Fundamentals Recap

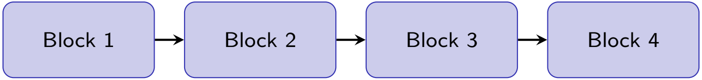
From Topic 3.2, recall:

What is a Blockchain?

- Distributed ledger of transactions
- Blocks linked by cryptographic hashes
- No central authority
- Immutable record (hard to change)

Key Components

- **Nodes:** Computers running the network
- **Consensus:** How nodes agree on truth
- **Transactions:** Records of value transfer
- **Blocks:** Groups of transactions



Prerequisites: Consensus Mechanisms

How do blockchains agree on the "truth"?

Proof of Work (PoW)

- Miners solve computational puzzles
- First to solve adds the block
- Requires significant energy
- Security through computational cost
- Example: Bitcoin

Proof of Stake (PoS)

- Validators lock up cryptocurrency
- Selected based on stake amount
- Energy efficient
- Security through economic incentives
- Example: Ethereum (since 2022)

The Blockchain Trilemma

Every blockchain must balance three properties:

Decentralization – Security – Scalability

You cannot maximize all three simultaneously.

Two Visions of Blockchain

Bitcoin

"Digital Gold"

- Store of value
- Fixed supply (21M)
- Minimal, robust
- Conservative changes
- "Don't break what works"

Goal: Sound money that
no one can inflate or censor

Ethereum

"World Computer"

- Platform for applications
- Programmable money
- Feature-rich, flexible
- Active development
- "Move fast, iterate"

Goal: Decentralized platform
for any application

Historical Context

Year	Event
2009	Bitcoin launches (Satoshi Nakamoto)
2012	Colored coins on Bitcoin (limited tokens)
2013	Ethereum whitepaper (Vitalik Buterin)
2015	Ethereum launches
2022	Ethereum transitions to Proof of Stake

Vitalik Buterin's insight (2013):
Bitcoin's scripting language was too limited. Think of it like the difference between a simple calculator (limited operations) and a full computer (can run any program). He proposed a blockchain with a **Turing-complete** programming language – one that could run any program, not just simple transactions.

The Key Difference

Bitcoin: "Is this a valid payment?" (simple yes/no)
Ethereum: "Run this arbitrary code and tell me the result" (general computation)

Key Term: Turing-Complete

Definition: Turing-Complete

A system is **Turing-complete** if it can compute anything that is theoretically computable – like a general-purpose computer. Think of it as the difference between a pocket calculator and a full computer.

In everyday terms: Bitcoin's simple calculator vs. Ethereum's full computer

Bitcoin Script

- NOT Turing-complete
- Limited operations
- No loops (by design)
- Can only check conditions
- Example: "Is signature valid?"

Ethereum EVM

- IS Turing-complete
- Full programming language
- Loops and conditionals
- Can run any program
- Example: DeFi protocols

(See dedicated EVM frame ahead)

Turing-completeness is named after Alan Turing, who defined what computation means mathematically.

Technical Comparison: Head to Head

Feature	**Bitcoin**	**Ethereum**
Launch	2009	2015
Consensus	Proof of Work	Proof of Stake (since 2022)
Block time	∼10 minutes	∼12 seconds
Supply cap	21 million BTC	No hard cap
Issuance	Halving every 4 years	Dynamic (can be deflationary)
Scripting	Limited (Bitcoin Script)	Turing-complete (EVM)
Primary use	Value transfer, store of value	Smart contracts, DeFi, NFTs
TPS	∼7	∼15-30
(For context: Visa handles ∼24,000 TPS)		

Key insight: These aren't competing for the same use case.
Bitcoin optimizes for *security and immutability*.
Ethereum optimizes for *programmability and flexibility*.

Bitcoin: The Sound Money Thesis

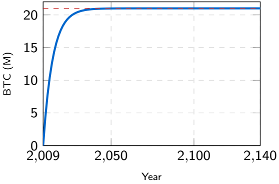
Core Properties

- **Fixed supply:** 21 million, ever
- **Predictable issuance:** Halving every 210,000 blocks
- **Decentralized:** No one controls it
- **Censorship resistant:** Anyone can transact
- **Immutable:** Rules don't change

The Narrative

- "Digital gold" – scarce, durable
- Hedge against inflation
- Separation of money and state
- Base layer for financial system

Bitcoin Supply Curve



Asymptotically approaches 21M

Bitcoin maximalists: "We already have one neutral, global money. Why do we need more?"

Bitcoin's Halving Mechanism

Definition: Halving

The **halving** is a pre-programmed event where Bitcoin's block reward is cut in half every 210,000 blocks (approximately every 4 years).

Halving	Year	Block Reward
Genesis	2009	50 BTC
1st	2012	25 BTC
2nd	2016	12.5 BTC
3rd	2020	6.25 BTC
4th	2024	3.125 BTC

Why it matters:

- Creates predictable, decreasing inflation
- Ensures 21M cap is never exceeded
- Historically associated with price increases (supply shock)

Ethereum: The World Computer Thesis

Core Properties

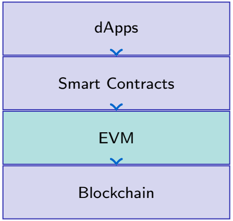
- **Smart contracts:** Self-executing code
- **EVM*:** Runs code on the blockchain
- **Programmable:** Any logic possible
- **Composable*:** Contracts call contracts
- **Tokens:** Create new assets easily

The Narrative

- Platform for dApps*
- "DeFi" – finance without banks
- NFTs, DAOs, and more
- Base layer for Web3

*EVM = Ethereum Virtual Machine (the "computer" running smart contracts). Composable = can be combined like LEGO bricks. dApps = decentralized applications.

Ethereum Stack



Apps built on contracts on EVM
(EVM explained in detail next)

Ethereum enables "programmable money" – money with built-in rules

Key Term: The Ethereum Virtual Machine (EVM)

You won't need to program – this is just to understand what happens behind the scenes.

Definition: EVM

The **Ethereum Virtual Machine** is a runtime environment that executes smart contract code. Every Ethereum node runs the same EVM, ensuring consistent execution worldwide.

How it Works

1. Developer writes contract in Solidity*
2. Code compiles to bytecode**
3. Bytecode deployed to blockchain
4. Every node can execute the code
5. Results are deterministic

Key Properties

- **Deterministic:** Same input = same output
- **Isolated:** Sandboxed*** execution
- **Metered:** Gas limits computation
- **Global:** Runs on all nodes

*Solidity = programming language for Ethereum smart contracts. **Bytecode = machine-readable instructions. ***Sandboxed = runs in an isolated environment for safety.

The EVM is like a global computer where everyone sees and verifies the same computation.

What Are Smart Contracts?

Traditional Contract

1. Legal document
2. Human interpretation
3. Court enforcement
4. (Maybe) execution

Requires: lawyers, courts, trust

Smart Contract

1. Code on blockchain
2. Automatic execution
3. No intermediaries
4. Guaranteed outcome

Requires: only the code

Definition: Smart Contract

A smart contract is code stored on a blockchain that automatically executes when predetermined conditions are met.
Key property: Once deployed, the code cannot be changed. “Code is law.”

Smart Contract Example: Escrow

Traditional Escrow

1. Buyer sends money to escrow agent
2. Seller ships goods
3. Buyer confirms receipt
4. Escrow agent releases funds to seller

Problems:

- Trust the escrow agent
- Agent takes a fee
- Disputes are slow
- Counterparty risk

Smart Contract Escrow

1. Buyer sends ETH to contract
2. Contract holds funds
3. Buyer calls `confirmReceipt()`
4. Contract automatically sends to seller

Advantages:

- Trust the code (auditable)
- Minimal fees (just gas)
- Instant execution
- No counterparty risk

Key insight: Smart contracts replace trusted intermediaries with verified code.

Key Term: Gas

Definition: Gas

Gas is the unit that measures computational work in Ethereum. Users pay gas fees (in ETH) to compensate validators for processing transactions.

Why Gas Exists:

1. **Prevents infinite loops:** Turing-complete code could run forever
2. **Allocates resources:** Scarce block space needs rationing
3. **Compensates validators:** Payment for processing
4. **Spam prevention:** Makes attacks expensive

Gas Calculation:

$$\boxed{\textbf{Transaction Fee} = \text{Gas Used} \times \text{Gas Price}}$$

Example: Simple ETH transfer uses ∼21,000 gas. At 50 gwei/gas = 0.00105 ETH fee.

Bitcoin doesn't need gas because its scripting is intentionally limited and cannot loop forever.

What “Programmable Money” Enables

DeFi

- Lending/borrowing
- Decentralized exchanges
- Yield farming*
- Derivatives**

NFTs/Tokens

- Digital ownership
- Tokenized assets***
- Loyalty programs
- Gaming items

DAOs

- Decentralized governance
- Treasury management
- Collective ownership
- Voting systems

*Yield farming = earning rewards by lending or providing liquidity to DeFi protocols. **Derivatives = financial contracts whose value depends on an underlying asset. ***Tokenized assets = real-world assets (property, stocks) represented as blockchain tokens.

All built on Ethereum's programmable foundation

The Power of Composability

Smart contracts can call other smart contracts. This means DeFi protocols can be combined like Lego blocks – creating entirely new financial products.

Key Terms: DeFi, NFT, DAO

DeFi

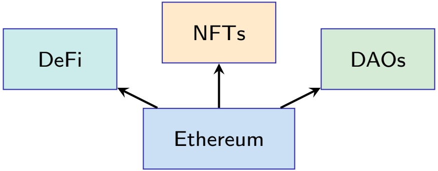
Decentralized Finance
Financial services (lending, trading, insurance) built on smart contracts without traditional intermediaries like banks.

NFT

Non-Fungible Token
A unique digital asset on the blockchain representing ownership of a specific item (art, collectibles, real estate).

DAO

Decentralized Autonomous Organization
An organization governed by smart contract rules and token-holder voting, with no traditional management structure.



The Tradeoff: Flexibility vs. Security

Bitcoin's Approach

- Limited scripting = fewer bugs
- Simple = easier to secure
- Ossification as a feature
- "If it ain't broke..."

Security record:

- Never been hacked
- No major protocol bugs
- Predictable behavior
- 15+ years of uptime

Ethereum's Approach

- Full programmability = more power
- Complex = more attack surface
- Continuous improvement
- "Move fast, fix things"

Security record:

- Many contract hacks
- The DAO hack (2016): $60M
- Ongoing exploits in DeFi
- "Code is law" cuts both ways

The Fundamental Tradeoff

More programmability = more capability = more things that can go wrong

Case Study: The DAO Hack (2016)

What Happened

The DAO was a smart contract that raised $150M in ETH. A bug in the code allowed an attacker to drain $60M.

The Dilemma:

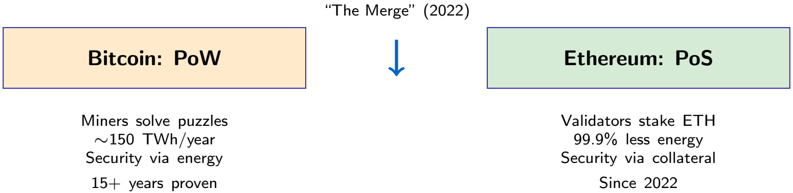
- The code executed exactly as written – no protocol bug
- “Code is law” – should the theft stand?
- But $60M was stolen due to a coding error

Ethereum’s Response:

- Community voted to “hard fork” – reverse the hack
- Created two chains: Ethereum (ETH) and Ethereum Classic (ETC)
- ETH: Reversed the hack — ETC: Kept the theft

Lesson: “Code is law” conflicts with “humans make mistakes.” Ethereum chose human intervention; Bitcoin’s philosophy would not.

Consensus: PoW vs. PoS



“The Merge” (2022)

Bitcoin: PoW

Miners solve puzzles
∼150 TWh/year
Security via energy

15+ years proven

Ethereum: PoS

Validators stake ETH
99.9% less energy
Security via collateral

Since 2022

Why Bitcoin keeps PoW:

- Proven security model
- True external cost to attack
- Conservative philosophy

Why Ethereum moved to PoS:

- Environmental concerns
- Enables future scaling
- Progressive philosophy

Market Positioning

Chain	Programmability	Market Cap	Niche
Bitcoin	Low	#1	Store of value
Ethereum	High	#2	Smart contract platform
Solana	High	#5	High-speed DeFi
Cardano	High	#9	Academic approach

Different chains, different niches:

- Bitcoin: Store of value, "digital gold", settlement layer
- Ethereum: Smart contract platform, DeFi hub
- Other L1s: Compete on speed, cost, specific use cases

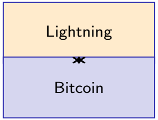
Key insight: It's not "which chain wins" – it's "which chain for which use case"

Layer 2 Scaling Solutions

Problem: Both Bitcoin (∼7 TPS) and Ethereum (∼15-30 TPS) are slow compared to Visa (∼24,000 TPS).

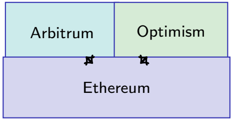
Bitcoin: Lightning Network

- Payment channels
- Off-chain transactions
- Instant, low fees
- Settles to main chain
- Focus: payments

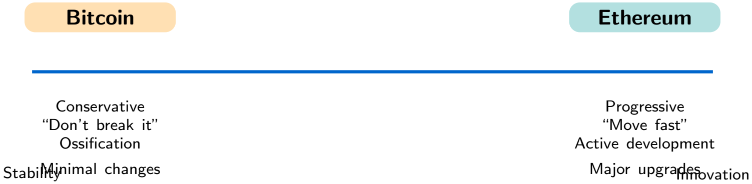


Ethereum: Rollups & Sidechains

- Optimism, Arbitrum (rollups)
- Polygon (sidechain)
- Batch transactions
- Submit proofs to L1
- Focus: all applications



Development Philosophy Comparison



Bitcoin

Ethereum

Conservative
“Don’t break it”
Ossification

Minimal changes

Stability

Progressive
“Move fast”
Active development

Major upgrades

Innovation

Bitcoin: Rare, carefully tested upgrades

- Only a handful of major changes since 2009
- Years of community debate before each one
- Philosophy: stability and predictability above all

Ethereum: Frequent upgrades adding new features

- New fee system (2021)
- Switched from PoW to PoS (2022)
- Regular hard forks with new capabilities
- Philosophy: improve and adapt continuously

Case Study: Bitcoin as Treasury Asset

MicroStrategy's Bitcoin Strategy

- CEO Michael Saylor began buying Bitcoin in 2020
- Rationale: Protect treasury from dollar inflation
- Holdings: 200,000+ BTC (as of 2024)
- Treats Bitcoin as "digital gold" for corporate reserves

Why Bitcoin (not Ethereum)?

1. Fixed supply – predictable scarcity
2. No protocol changes – long-term stability
3. Simpler to value – pure store of value thesis
4. Regulatory clarity – treated as commodity in US

Implication

Bitcoin's conservative design makes it attractive for entities seeking predictable, long-term value storage.

Case Study: Ethereum as DeFi Infrastructure

Uniswap: Decentralized Exchange

- Launched 2018 on Ethereum
- No order book – uses automated market makers (AMM)
- Over $1 trillion in cumulative trading volume
- Governance by UNI token holders

Why Ethereum (not Bitcoin)?

1. Requires smart contracts – not possible on Bitcoin
2. Needs token creation – ERC-20 standard
3. Composability – integrates with other DeFi protocols
4. EVM ecosystem – developers, tools, infrastructure

Implication

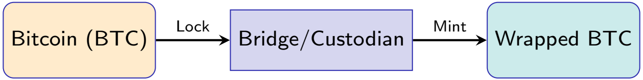
Ethereum's programmability enables financial applications impossible on Bitcoin, but with added complexity and risk.

Case Study: Cross-Chain Bridges

Problem: Users want Bitcoin's store of value + Ethereum's DeFi

Solution: Wrapped Bitcoin (WBTC)

- ERC-20 token on Ethereum backed 1:1 by Bitcoin
- Allows BTC holders to participate in DeFi
- Over $5 billion in circulation



Risks:

- Custodial risk – trust the bridge operator
- Bridge hacks – over $2B stolen from bridges (2021-2023)
- Regulatory uncertainty

Regulatory Landscape: Bitcoin vs. Ethereum

Bitcoin: Generally Clearer

- US: Treated as commodity (CFTC)
- No pre-mine, no foundation
- Decentralized from day one
- Spot ETFs approved (2024)

Lower regulatory risk

Ethereum: More Complex

- Had ICO (initial coin offering)
- Ethereum Foundation exists
- Proof of Stake = "staking rewards"
- SEC scrutiny on classification

Higher regulatory uncertainty

Key Question

Is ETH a security or a commodity? The answer affects exchanges, staking services, and DeFi protocols. Bitcoin's simpler design avoids this ambiguity.

Discussion Questions

1. **Store of value vs. programmability:** Can Ethereum also be “sound money”? Can Bitcoin add smart contracts?
2. **The DAO hack:** Ethereum’s response (hard fork to reverse the hack) violated “code is law.” Was this the right call?
3. **Energy debate:** Bitcoin’s PoW energy use is criticized, but it’s also what makes it secure. Ethereum moved to PoS – did it sacrifice anything?
4. **Maximalism:** Many believe only one blockchain will “win.” Others see room for many. What’s your view?
5. **Regulation:** How might different design philosophies affect regulatory treatment? Is ETH a security?

Application: Choosing a Platform

Scenario: You're advising a startup on blockchain strategy.

Use Case	Best Fit	Why
Corporate treasury reserve	Bitcoin	Fixed supply, stability
Decentralized lending app	Ethereum	Smart contracts
Cross-border remittances	Either/L2	Speed vs. security tradeoff
Tokenized real estate	Ethereum	Token standards
Long-term savings (10+ years)	Bitcoin	Conservative, proven
NFT marketplace	Ethereum	ERC-721 standard

Key Takeaway

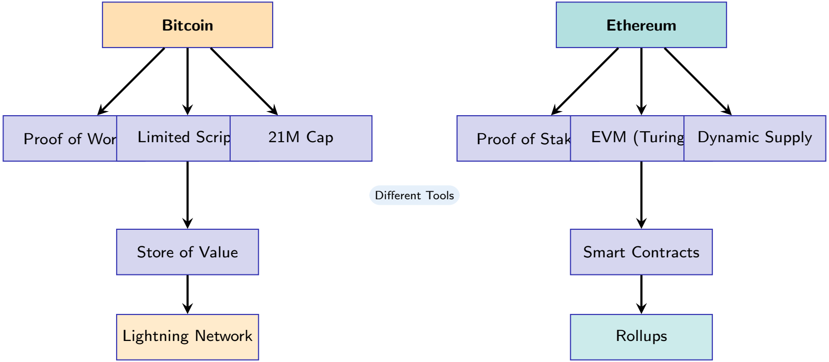
The right platform depends on your specific needs. Understanding both philosophies helps you make informed decisions.

Executive Summary

1. **Different Goals:** Bitcoin = sound money (store of value); Ethereum = world computer (programmable platform)
2. **Key Technical Difference:** Bitcoin Script is limited by design; Ethereum's EVM is Turing-complete
3. **The Tradeoff:** More programmability means more capability but also more attack surface
4. **Not Competitors:** They serve different use cases and can coexist
5. **Both Face Scaling Challenges:** Layer 2 solutions (Lightning, Rollups) address throughput limits

Bottom Line: Understanding both platforms' design philosophies is essential for navigating digital finance.

Concept Map: Architecture Comparison



Key Terms & Definitions (1/2)

Turing-Complete A system that can compute anything computable; has the power of a general-purpose computer. Ethereum's EVM is Turing-complete; Bitcoin Script is not.

Smart Contract Code stored on a blockchain that automatically executes when conditions are met. Immutable once deployed ("code is law").

EVM Ethereum Virtual Machine – the runtime environment that executes smart contract code on every Ethereum node.

Gas The unit measuring computational work in Ethereum. Users pay gas fees to compensate validators and prevent infinite loops.

Halving Bitcoin's pre-programmed event where block rewards are cut in half every 210,000 blocks, ensuring the 21M supply cap.

Key Terms & Definitions (2/2)

DeFi Decentralized Finance – financial services (lending, trading, insurance) built on smart contracts without traditional intermediaries.

NFT Non-Fungible Token – a unique digital asset representing ownership of a specific item on the blockchain.

DAO Decentralized Autonomous Organization – an organization governed by smart contract rules and token-holder voting.

Layer 2 Solutions built on top of a blockchain (Layer 1) to increase transaction throughput while inheriting the base layer's security.

The Merge Ethereum's September 2022 transition from Proof of Work to Proof of Stake, reducing energy consumption by 99.9%.

Common Misconceptions

Myth	Reality
“Bitcoin and Ethereum are direct competitors”	They solve different problems: Bitcoin = sound money; Ethereum = programmable platform
“Ethereum is just a faster Bitcoin”	Ethereum’s speed is secondary to its programmability. Bitcoin is intentionally slower for security.
“Smart contracts can do anything”	They can only access on-chain data. Real-world data requires “oracles” (trusted data feeds).
“Proof of Stake is always better than Proof of Work”	Each has tradeoffs. PoW provides external security cost; PoS is more energy efficient but newer.

Industry Perspective: Who Uses What?

Bitcoin Adopters

- MicroStrategy (Treasury)
- Tesla (Holdings)
- El Salvador (Legal Tender)
- BlackRock (ETF)
- Fidelity (Custody)

Ethereum Builders

- Uniswap (DEX)
- Aave (Lending)
- OpenSea (NFTs)
- MakerDAO (Stablecoin)
- ENS (Identity)

Multi-Chain

Coinbase, Binance
Grayscale, PayPal

Major Banks (exploring)

Pattern

Institutions seeking **store of value** gravitate to Bitcoin.
Builders creating **applications** build on Ethereum.

Self-Assessment Question 1

Q3: What is the main difference between Bitcoin Script and Ethereum's EVM?

A) Bitcoin Script is faster than the EVM

B) Bitcoin Script is limited and not Turing-complete, while the EVM is Turing-complete

C) The EVM can only handle simple transactions

D) Bitcoin Script uses more gas than the EVM

Self-Assessment Question 1

Q3: What is the main difference between Bitcoin Script and Ethereum's EVM?

A) Bitcoin Script is faster than the EVM

B) Bitcoin Script is limited and not Turing-complete, while the EVM is Turing-complete

C) The EVM can only handle simple transactions

D) Bitcoin Script uses more gas than the EVM

Answer: B

Bitcoin Script is intentionally limited and not Turing-complete – it can answer simple yes/no questions like “Is this a valid payment?” The Ethereum Virtual Machine is Turing-complete, meaning it can run any computable program. Think of it as Bitcoin's simple calculator vs. Ethereum's full computer. This reflects their design philosophies: Bitcoin prioritizes security through simplicity; Ethereum prioritizes programmability.

Self-Assessment Questions 2-3

Q11: How does Bitcoin's approach to changes differ from Ethereum's?

A) Bitcoin has no way to make protocol changes

B) Bitcoin follows "don't break what works"; Ethereum follows "move fast, iterate"

C) Ethereum never makes protocol changes

D) Both make changes at exactly the same rate

Answer: B – Bitcoin is conservative; Ethereum is progressive.

Q19: What is the key tradeoff between Bitcoin's limited scripting and Ethereum's Turing-completeness?

A) Bitcoin is faster than Ethereum

B) Bitcoin's simplicity provides fewer attack vectors; Ethereum's complexity enables more capabilities but more potential bugs

C) Ethereum cannot process simple payments

D) Bitcoin has no tradeoffs

Answer: B – More programmability = more capability = more attack surface.

What's Next: Day 4

Programmable Finance

Smart Contracts, DeFi, and Tokenization

- How smart contracts enable DeFi protocols
- Lending, borrowing, and trading without intermediaries
- Tokenization: bridging traditional and crypto finance
- Real-world asset tokenization
- Risks and opportunities in DeFi

Day 3: Foundations → Day 4: Applications → Day 5: Integration

Resources for Further Learning

Bitcoin

- Bitcoin Whitepaper: `bitcoin.org/bitcoin.pdf`
- "Mastering Bitcoin" by Andreas Antonopoulos
- Bitcoin Wiki: `en.bitcoin.it`

Ethereum

- Ethereum Whitepaper: `ethereum.org/whitepaper`
- "Mastering Ethereum" by Antonopoulos & Wood
- Ethereum Documentation: `ethereum.org/developers`

Comparison & Analysis

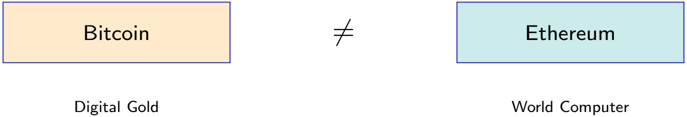
- CoinMetrics: On-chain data and research
- Messari: Crypto asset intelligence
- Glassnode: Bitcoin-focused analytics

Remember: Focus on understanding the "why" behind design decisions, not just the "what."

Questions?

Topic 3.4: Bitcoin vs. Ethereum

Two Design Philosophies



Bitcoin ≠ Ethereum

Digital Gold

World Computer

Joerg Osterrieder — Digital Finance — 2025