# Topic 3.5: Layer 2 Scaling Solutions [ADVANCED]

## Scaling Ethereum and Beyond

Joerg Osterrieder

Digital Finance

2025

# Learning Objectives

**By the end of this topic, you will be able to:**

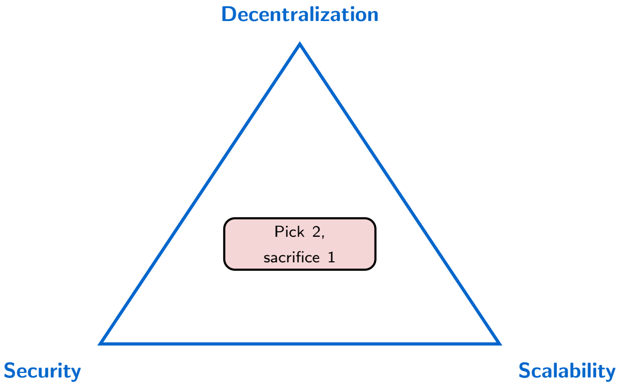
1. **Explain** why Layer 2 solutions are needed and how they relate to the blockchain trilemma
2. **Compare** different L2 approaches: payment channels, sidechains, and rollups
3. **Distinguish** between Optimistic Rollups and ZK-Rollups, including their tradeoffs
4. **Evaluate** the security assumptions and trust models of various L2 solutions
5. **Analyze** the role of bridges in cross-chain communication and their risks
6. **Apply** knowledge of L2 economics to real-world use case selection

## Core Question

How can we process thousands of transactions per second without sacrificing decentralization or security?

# Prerequisites: The Blockchain Trilemma (Recap from T3.2)

**Every blockchain must balance three properties:**



**The Problem:** Ethereum L1 processes ~15 Transactions Per Second (TPS). Visa processes ~24,000 TPS.
**The Question:** Can we scale without centralizing?

**Layer 2 solutions aim to break this trilemma by building on top of secure L1s**

# Prerequisites: The Gas Fee Problem

**Why are Ethereum transactions expensive?**

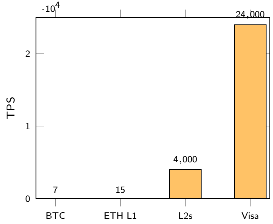
### Block Space is Scarce

- Each block has limited capacity
- Users bid for inclusion (gas fees)
- High demand = high fees
- 2021 peak: $50+ for simple transfers

### The User Experience Problem

- Small transactions become uneconomical
- $5 coffee + $20 fee = unusable
- Prices out retail users
- Benefits only whales

**Transaction Throughput Comparison**



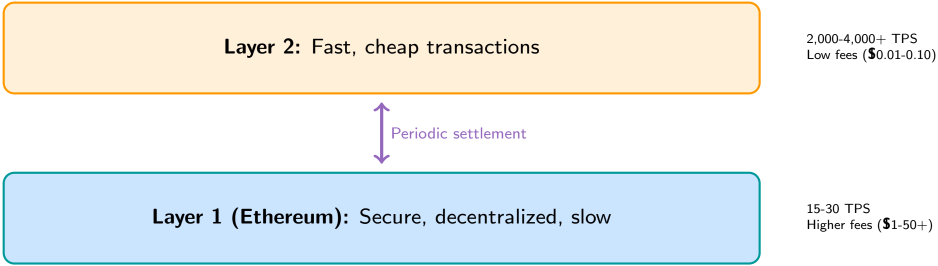
## Key Insight

Layer 2 solutions process transactions off-chain, then settle on L1 – getting L2 speed with L1 security.

# What Is Layer 2?

## Definition

**Layer 2 (L2)** refers to any off-chain network, system, or technology built on top of a blockchain (Layer 1) to extend its capabilities – primarily scaling and speed – while inheriting the security of the underlying L1.



**Layer 2:** Fast, cheap transactions

2,000-4,000+ TPS
Low fees ($0.01-0.10)

Periodic settlement

**Layer 1 (Ethereum):** Secure, decentralized, slow

15-30 TPS
Higher fees ($1-50+)

**Analogy:** L2 is like a bar tab – you make many small transactions throughout the night, but only settle once at the end.

# Why Layer 2 Matters: The Scalability Crisis

**The Problem (2020-2021)**

- DeFi and NFT boom overwhelmed Ethereum
- Gas fees spiked to $50-200 per transaction
- Small users priced out entirely
- “Ethereum is for whales only”

**The Stakes**

- Mass adoption impossible at 15 TPS
- Competitors (Solana, Avalanche) gaining ground
- Ethereum risked losing its lead
- Urgent need for scaling solutions

**Scale Requirements:**

Current: 15-30 TPS
For global payments: 100,000+ TPS

**Gap: 3,000x improvement needed**

**L2 Solution Benefits:**

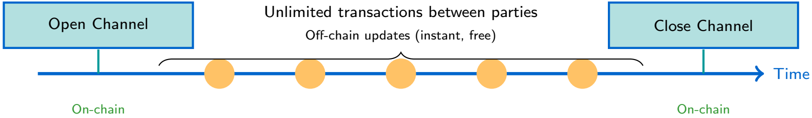
- 90%+ fee reduction
- 100x+ throughput increase
- Inherit L1 security
- No L1 changes required

**Vitalik Buterin: “Rollups are the only viable scaling solution for Ethereum in the medium term”**

# Payment Channels: The Original L2

## Definition

A **payment channel** is a two-party agreement to transact off-chain, with only the opening and closing transactions recorded on-chain.


Open Channel
Unlimited transactions between parties
Off-chain updates (instant, free)
Close Channel
Time
On-chain
On-chain

**Bar Tab Analogy:**

1. **Open tab:** Lock funds in a shared account (on-chain)
2. **Buy drinks:** Update balance off-chain (instant, no fees)
3. **Close tab:** Settle final balance on-chain

# Lightning Network: Bitcoin's Payment Channels

### How It Works

1. Alice and Bob open a channel (on-chain)
2. They can transact unlimited times (off-chain)
3. Either party can close the channel anytime
4. Final balances settle on Bitcoin

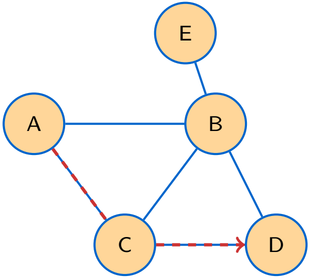
### Network of Channels

- Channels can be routed through intermediaries
- Alice → Carol → Bob
- No direct channel needed
- Atomic: all-or-nothing routing

**Stats (2024):**

- 15,000+ nodes
- 60,000+ channels
- $200M+ capacity

**Lightning Network Topology**



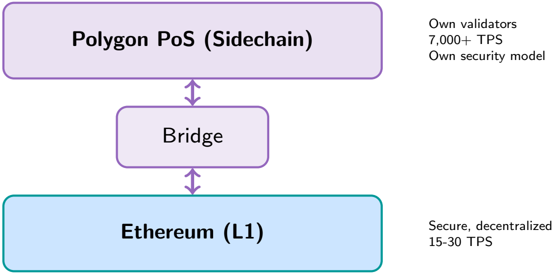
A pays D via C

**Limitation:** Only for payments, not smart contracts

# Sidechains: The Polygon Approach

## Definition

A **sidechain** is an independent blockchain that runs parallel to a main chain, connected via a two-way bridge. Sidechains have their own consensus and security.



**Key Difference from True L2:** Sidechains have *independent security* – they don't inherit Ethereum's security. If Polygon's validators collude, funds could be at risk.

# Sidechain Trust Model

### Advantages

- Very high throughput
- Low fees ($0.001-0.01)
- EVM compatible (see Topic 3.4)
- Easy to use
- Mature ecosystem

### Polygon PoS Stats:

- 100+ validators
- 7,000+ TPS capacity
- $1B+ TVL
- Used by: Uniswap, Aave, OpenSea

### Disadvantages

- Weaker security than L1
- Fewer validators
- Must trust validator set
- Bridge risk
- Not “true” L2

**Security Model:**

Ethereum: 500,000+ validators
Polygon PoS: ~100 validators

Smaller validator set = easier to attack

**Sidechains trade security for speed – acceptable for some use cases, not others**

# Plasma: Historical Context

### What Was Plasma? (2017)

- Proposed by Vitalik Buterin and Joseph Poon
- Child chains anchored to Ethereum
- Only submit state roots to L1
- Users can exit to L1 if problems arise

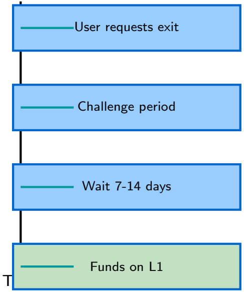
### The Promise

- "Millions of transactions per second"
- Inherit Ethereum security
- Low fees

### Why It Didn't Work

- Complex exit game (up to 2 weeks)
- Data availability problem
- Hard to support smart contracts (general computation)
- Mass exit vulnerability

**Plasma Exit Game**



User requests exit

Challenge period

Wait 7-14 days

Funds on L1

Too slow, too complex

**Plasma's limitations led directly to the development of rollups**

# Lessons from Plasma

**Why Plasma Failed – and What We Learned**

| Plasma Problem | Why It Matters | Rollup Solution |
|---|---|---|
| Data unavailability | Can't verify state | Post data to L1 |
| Complex exits | Bad UX | Simpler withdrawal process |
| Limited computation | Can't run smart contracts | EVM equivalence (see T3.4) |
| Mass exit attacks | Network congestion | Better exit mechanisms |

## The Key Insight

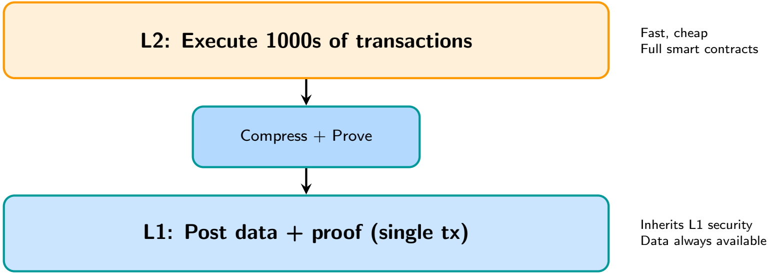
Plasma tried to minimize data posted to L1. Rollups take the opposite approach: post all transaction data to L1, but execute it off-chain. This is called **data availability**.

**Result:** Rollups became the dominant L2 paradigm by 2021-2022.

# Rollups: The Modern L2 Approach

## Definition

A **rollup** executes transactions off-chain, then posts compressed transaction data and a state root to L1. Users can always reconstruct the L2 state from L1 data alone.



**Shipping Container Analogy:** Instead of shipping items one by one, pack 1,000 items into one container and ship that. More efficient, same destination.

# The Data Availability Guarantee

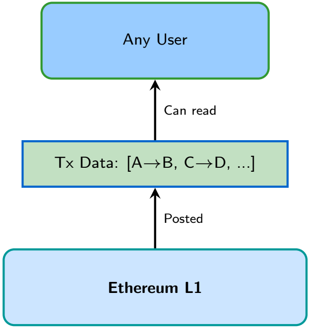
**Why posting data to L1 matters:**

### With Data Availability

- Anyone can reconstruct L2 state
- No need to trust the sequencer
- If L2 operators disappear, users can exit
- "Trustless" scaling

### Without Data Availability

- Must trust operators to provide data
- If operators withhold data, funds stuck
- "Trusted" scaling (weaker security)
- This was Plasma's problem



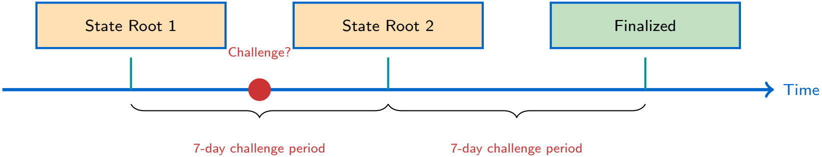
Always recoverable

**Key Property:**
Rollup security = L1 security
(assuming data is on L1)

# Optimistic Rollups: Assume Valid, Prove Fraud

## Definition

**Optimistic Rollups** assume transactions are valid by default ("optimistic") and only compute proofs if someone challenges a transaction during a dispute period.



**How it works:**

1. Sequencer posts state root to L1 (assumes valid)
2. Anyone can challenge with a **fraud proof** if they detect cheating
3. After 7 days with no successful challenge, state is finalized

*Analogy:* Like a warranty period where anyone can point out defects before the product is final.

# Fraud Proofs: Keeping Optimistic Rollups Honest

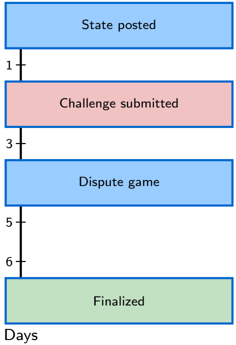
### The Fraud Proof Process

1. Sequencer posts: “State changed from A to B”
2. Challenger says: “That’s wrong!”
3. Interactive dispute on L1:
   - Narrow down to single instruction
   - L1 executes that instruction
   - Whoever was wrong loses bond

### Why 7 Days?

- Gives everyone time to detect fraud
- Allows time to submit challenge
- Allows time to complete the dispute game
- Works even during network congestion
- *Note: This delay exists to give anyone time to challenge potentially fraudulent transactions*

### Fraud Proof Timeline



**In practice, fraud is rare because cheaters lose their bond – it’s not profitable**

# Optimism and Arbitrum: The Leading Optimistic Rollups

**Optimism**

- Launched 2021
- EVM equivalent (see T3.4)
- Simple fraud proof design
- OP token for governance
- "Superchain" vision
- TVL: $800M+ (2024)

**Notable Users:**

- Uniswap
- Aave
- Synthetix

**Arbitrum**

- Launched 2021
- EVM equivalent (see T3.4)
- More complex fraud proofs
- ARB token for governance
- Arbitrum One + Nova
- TVL: $2.5B+ (2024)

**Notable Users:**

- GMX
- Radiant
- Treasure

**Both offer:** 90%+ fee reduction, 10-50x throughput increase, same Ethereum security (after 7 days)

# ZK-Rollups: Prove Validity Mathematically

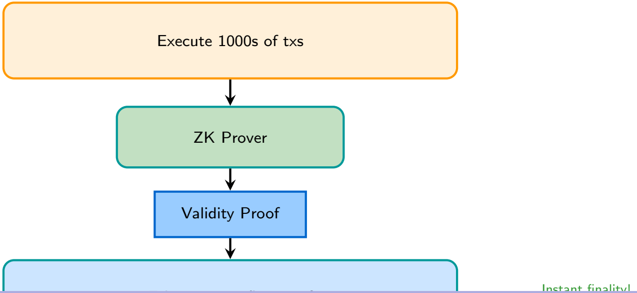
## Simple Hook

Imagine proving you know a secret without revealing the secret itself – that's the power of zero-knowledge proofs.

## Definition

**ZK-Rollups** use zero-knowledge proofs to cryptographically prove that all transactions in a batch are valid. No challenge period needed – validity is mathematically guaranteed.

*Analogy:* Like a math teacher verifying your homework is correct without checking every step.



# Validity Proofs: How ZK-Rollups Work

## Zero-Knowledge Proofs

- Prove a statement is true without revealing details
- Example: “I know the password” without showing it
- In ZK-rollups: “These 1000 txs are valid” in one small proof

## The Process

1. Sequencer executes transactions off-chain
2. ZK prover generates validity proof
3. Proof + state root posted to L1
4. L1 contract verifies proof (cheap!)
5. If proof valid, state is immediately finalized

*Analogy:* Like a math teacher checking your work upfront vs trusting you did it correctly.

## Proof Types:

- **SNARKs** (Succinct Non-interactive ARgument of Knowledge): smaller proofs, but need a “trusted setup”
- **STARKs** (Scalable Transparent ARgument of Knowledge): larger proofs, but don't require trusted setup
- *Technical names for two types of mathematical proof systems*

**Proof Compression:**

1000 transactions
↓
1 proof (∼300 bytes)

Verification: ∼500k gas
(cheaper than 1000 txs!)

## Analogy:

A teacher checking 1000 homework assignments vs. having a machine that instantly verifies “all 1000 are correct” in one step.

**ZK proofs are computationally expensive to generate but cheap to verify**

# zkSync and StarkNet: The Leading ZK-Rollups

**zkSync Era**

- By Matter Labs
- Launched 2023
- EVM compatible (zkEVM, see T3.4)
- SNARKs-based proofs (Succinct Non-interactive ARguments of Knowledge)
- ZK token airdrop 2024
- TVL: $150M+ (2024)

**Key Features:**

- Native account abstraction
- Hyperchain vision
- Growing ecosystem

**StarkNet**

- By StarkWare
- Launched 2022
- Custom language (Cairo)
- STARKs-based proofs (Scalable Transparent ARguments of Knowledge)
- STRK token 2024
- TVL: $200M+ (2024)

**Key Features:**

- No trusted setup
- StarkEx for app chains
- Strong research team

## The EVM Compatibility Challenge

Making ZK proofs for EVM execution is extremely difficult. zkSync achieved it with zkEVM; StarkNet uses a custom VM (requires learning Cairo).

# Optimistic vs. ZK Rollups: Head-to-Head

| **Aspect** | **Optimistic Rollups** | **ZK-Rollups** |
|---|---|---|
| Withdrawal time | 7 days* | Minutes |
| Proof type | Fraud proof (if challenged) | Validity proof (always) |
| Computation cost | Minimal | Heavy (prover) |
| EVM compatibility | High (mature) | Developing |
| Security model | Optimistic + challenge | Cryptographic |
| Maturity | More mature | Newer |
| Examples | Optimism, Arbitrum | zkSync, StarkNet |

*This delay exists to give anyone time to challenge potentially fraudulent transactions

**Choose Optimistic when:**

- Need full EVM compatibility
- 7-day withdrawal acceptable
- Deploying existing contracts

**Choose ZK when:**

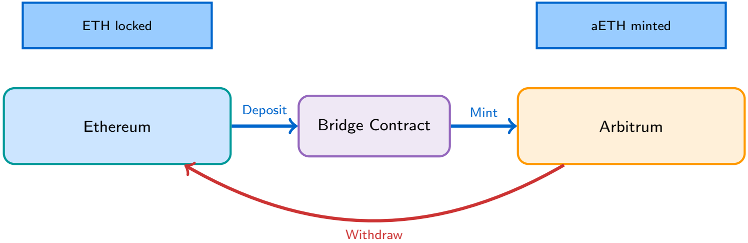
- Need fast withdrawals
- High-security applications
- Future-proofing matters

**Industry consensus (2024):** ZK-rollups are the long-term future, but optimistic rollups are more practical today due to EVM maturity. Both will likely coexist.

# Bridges: Connecting L1 and L2

## Definition

A **bridge** is a protocol that allows assets to move between different blockchains or layers. Bridges lock assets on one chain and mint equivalent assets on another.



**Bridge Types:**

- **Native/Canonical:** Official rollup bridges (most secure, slowest)
- **Third-party:** Hop, Across, Stargate (faster, more risk)
- **CEX bridges:** Via exchanges (convenient, custodial)

# Bridge Security: The Weak Link

**Bridges are prime targets for hackers:**

| Hack | Amount | Cause |
|---|---:|---|
| Ronin (2022) | $625M | Compromised validators |
| Wormhole (2022) | $320M | Smart contract bug |
| Nomad (2022) | $190M | Verification flaw |
| Harmony (2022) | $100M | Compromised keys |

### Why Bridges Are Vulnerable

- Large TVL (attractive target)
- Complex smart contracts
- Cross-chain communication hard
- Often centralized components

### Risk Mitigation

- Use canonical bridges when possible
- Limit amounts bridged
- Check bridge audits
- Diversify across bridges

**“Bridges are the Achilles heel of the multi-chain world” – many security experts**

# Hands-On: Exploring Layer 2 (NB15 Preview)

**In the upcoming notebook, you will:**

1. **Compare gas fees** – Execute same transaction on L1 vs. L2
2. **Explore rollup data** – See how batches are posted to Ethereum
3. **Track bridge activity** – Monitor deposits and withdrawals
4. **Analyze L2 economics** – Calculate savings and throughput

**Tools You'll Use:**

Arbiscan, Optimistic Etherscan, L2Beat
Python web3 library for data analysis

**Key Question to Explore:**
How much cheaper is a Uniswap swap on Arbitrum vs. Ethereum mainnet?

# NB15: What You'll Learn

**Practical Skills:**

- Reading L2 block explorers
- Understanding batch submissions
- Comparing fee structures
- Evaluating bridge options

**Analysis Tasks:**

- Calculate real-world savings
- Track L2 adoption metrics
- Compare rollup performance
- Assess security tradeoffs

**Expected Findings:**

1. L2 fees are 10-100x cheaper
2. Batch sizes vary by demand
3. Different L2s serve different niches
4. Bridge liquidity affects speed

**Time estimate:** 30-45 min
**Prerequisites:** T3.2, T3.4

**No wallet or funds needed – purely analytical exercise using public data**

# Discussion: Layer 2 Economics

**Who benefits from Layer 2?**

**Winners**

- Retail users (affordable fees)
- DeFi protocols (more users)
- L2 operators (sequencer fees)
- Ethereum (more value settled)

**Losers?**

- L1 validators? (less direct fees)
- Alternative L1s? (less migration)
- Complexity (user confusion)

**Discussion Questions:**

1. Should L2 fees go to L1 validators to align incentives?
2. Will L2s eventually compete with Ethereum itself?
3. How do token airdrops (OP, ARB, ZK) affect adoption?

# Discussion: L2 Decentralization Concerns

**Are Layer 2s actually decentralized?**

| L2 | Sequencer | Status (2024) |
|---|---|---|
| Arbitrum | Single (Offchain Labs) | Decentralization planned |
| Optimism | Single (Optimism Foundation) | Decentralization planned |
| zkSync | Single (Matter Labs) | Decentralization planned |
| StarkNet | Single (StarkWare) | Decentralization planned |

**The Sequencer Problem:**

- Currently centralized sequencers order transactions
- Could censor or front-run users
- "Escape hatch" exists but requires L1 transaction
- Decentralized sequencers are technically challenging

## Question

Is it acceptable for L2s to have centralized sequencers if users can always exit to L1?

# Discussion: The Future of Scaling

**Where is the industry heading?**

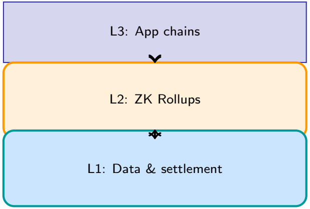
### Current Trends

- L2s becoming dominant
- ZK technology maturing
- Cross-L2 bridges emerging
- App-specific rollups

### Open Questions

- Will ZK replace Optimistic?
- How will liquidity fragment?
- What about L3s?
- Composability across L2s?

### Potential Futures



Ethereum as "settlement layer"
L2s for general compute
L3s for specific apps

# Executive Summary: Key Takeaways

1. **Layer 2 solves the scalability problem**
   Process thousands of transactions off-chain, settle on L1 for security.

2. **Rollups are the dominant L2 approach**
   Post all data to L1, enabling trustless verification and exits.

3. **Optimistic = assume valid, prove fraud**
   7-day withdrawal, but mature and EVM-compatible.

4. **ZK = prove validity cryptographically**
   Instant finality, but computationally expensive and newer.

5. **Bridges are critical infrastructure but high-risk**
   $1B+ lost to bridge hacks – use with caution.

6. **L2s unlock mass adoption**
   90%+ fee reduction makes Ethereum usable for everyone.

# Key Terms & Definitions (1/2)

Layer 2 (L2) A secondary framework built on top of an existing blockchain (L1) to improve scalability while inheriting L1 security.

Rollup An L2 that executes transactions off-chain but posts transaction data to L1, enabling anyone to reconstruct the state.

Optimistic Rollup A rollup that assumes transactions are valid by default and uses fraud proofs to catch invalid state transitions.

ZK-Rollup A rollup that uses zero-knowledge proofs to cryptographically prove transaction validity, enabling instant finality.

Data Availability The guarantee that transaction data is published and accessible, allowing anyone to verify the L2 state.

# Key Terms & Definitions (2/2)

Fraud Proof A cryptographic proof submitted during a challenge period showing that a state transition was invalid.

Validity Proof A cryptographic proof (SNARK = Succinct Non-interactive ARgument of Knowledge / STARK = Scalable Transparent ARgument of Knowledge) that mathematically proves all transactions in a batch are valid.

Sequencer The entity responsible for ordering and batching L2 transactions before posting to L1.

Bridge A protocol enabling assets to move between different blockchains or layers by locking on one chain and minting on another.

Payment Channel An L2 mechanism allowing two parties to transact off-chain unlimited times, only settling on-chain when closing.

# Common Misconceptions

| Myth | Reality |
|---|---|
| “L2s are less secure than L1” | Rollups **inherit L1 security** because all data is on L1. Users can always exit to L1 if the L2 fails. |
| “All L2s are the same” | Different L2 types have **vastly different trust models**. Sidechains have independent security; rollups inherit L1 security. |
| “7-day withdrawal means 7 days to use funds” | The 7-day challenge period is only for **L2→L1 withdrawals**. On L2, transactions are instant. This delay exists to give anyone time to challenge potentially fraudulent transactions. |
| “ZK-rollups are always better” | ZK-rollups have **tradeoffs**: higher proving costs, developing EVM compatibility, and newer technology. |

# Self-Assessment Questions (1/2)

**Question 1:** What is the main difference between Optimistic and ZK-Rollups?

A. Optimistic rollups are faster than ZK-rollups

B. Optimistic rollups use fraud proofs; ZK-rollups use validity proofs

C. ZK-rollups have a 7-day withdrawal period

D. Optimistic rollups don't post data to L1

# Self-Assessment Questions (1/2)

**Question 1:** What is the main difference between Optimistic and ZK-Rollups?

A. Optimistic rollups are faster than ZK-rollups

B. Optimistic rollups use fraud proofs; ZK-rollups use validity proofs

C. ZK-rollups have a 7-day withdrawal period

D. Optimistic rollups don't post data to L1

**Answer: B**
*Explanation:* Optimistic rollups assume validity and only compute proofs if challenged (fraud proofs). ZK-rollups prove validity cryptographically before posting (validity proofs). This is why ZK-rollups have instant finality while optimistic rollups need a 7-day challenge period (to give anyone time to challenge potentially fraudulent transactions).

# Self-Assessment Questions (2/2)

**Question 2:** Why are bridges considered high-risk infrastructure?

A. Bridges are slower than direct L1 transactions

B. Bridges hold large amounts of locked assets and have complex smart contracts

C. Bridges don't support all tokens

D. Bridges are not decentralized

# Self-Assessment Questions (2/2)

**Question 2:** Why are bridges considered high-risk infrastructure?

A. Bridges are slower than direct L1 transactions
B. Bridges hold large amounts of locked assets and have complex smart contracts
C. Bridges don't support all tokens
D. Bridges are not decentralized

**Answer: B**
*Explanation:* Bridges are attractive targets because they hold large TVL, have complex cross-chain logic, and often rely on centralized components. Over $1B has been lost to bridge hacks (Ronin, Wormhole, Nomad).

**Question 3:** What makes rollup security different from sidechain security?

# Self-Assessment Questions (2/2)

**Question 2:** Why are bridges considered high-risk infrastructure?

A. Bridges are slower than direct L1 transactions
B. Bridges hold large amounts of locked assets and have complex smart contracts
C. Bridges don't support all tokens
D. Bridges are not decentralized

**Answer: B**
*Explanation:* Bridges are attractive targets because they hold large TVL, have complex cross-chain logic, and often rely on centralized components. Over $1B has been lost to bridge hacks (Ronin, Wormhole, Nomad).

**Question 3:** What makes rollup security different from sidechain security?
**Answer:** Rollups post all transaction data to L1, inheriting L1 security. Sidechains have independent validators and their own security – if sidechain validators collude, funds could be at risk.

# What's Next: Topic 4.1 – Smart Contracts

**From scaling to programming: How to build on blockchains**

**Topics we'll cover:**

- What is a smart contract?
- Solidity programming basics
- EVM execution model
- Common patterns and pitfalls
- Security considerations

**Key insight preview:**

**Smart Contracts:**

Code that executes exactly as written, with no human interpretation.

"Code is law" – for better and worse.

**Connection to L2:** Most L2s are EVM-compatible, meaning smart contracts written for Ethereum can deploy to L2s with minimal changes – same code, lower fees.

# Resources for Further Learning

**Essential Reading:**

- Vitalik Buterin: “An Incomplete Guide to Rollups” (2021)
- L2Beat: `https://l2beat.com` – L2 risk analysis and metrics
- Ethereum.org: Layer 2 documentation

**Data & Analytics:**

- **L2Beat:** Comprehensive L2 comparison and risk assessment
- **Arbiscan:** Arbitrum block explorer
- **Optimistic Etherscan:** Optimism block explorer
- **Dune Analytics:** L2 adoption dashboards

**Deep Dives:**

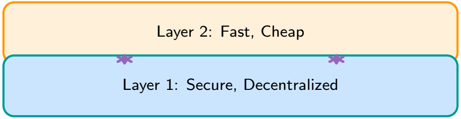
- Matter Labs blog: ZK-rollup technology explained
- Paradigm research: Optimistic rollup design
- StarkWare blog: STARKs vs. SNARKs

**NB15 provides hands-on exploration of L2 data and metrics**

# Questions?

Topic 3.5: Layer 2 Scaling Solutions [ADVANCED]

Scaling Ethereum and Beyond



Layer 2: Fast, Cheap

Layer 1: Secure, Decentralized

**Next:** Topic 4.1 – Smart Contracts

# Reference: L2 Solutions Comparison

| Solution | Type | Security | Finality | TPS | EVM |
|---|---|---|---|---|---|
| Arbitrum One | Optimistic | L1 inherited | 7 days | 4,000+ | Yes |
| Optimism | Optimistic | L1 inherited | 7 days | 2,000+ | Yes |
| zkSync Era | ZK | L1 inherited | Minutes | 2,000+ | Yes |
| StarkNet | ZK | L1 inherited | Minutes | 1,000+ | No (Cairo) |
| Polygon PoS | Sidechain | Independent | Seconds | 7,000+ | Yes |
| Lightning | Channel | L1 inherited | Instant | 1M+ | No |

**2024 TVL Rankings:**

1. Arbitrum: $2.5B+
2. Optimism: $800M+
3. Polygon: $1B+ (sidechain)
4. zkSync: $150M+
5. StarkNet: $200M+

Source: L2Beat, DeFiLlama (data as of 2024)