



ARBITRUM



Ups and Downs: Onboarding a Million Users to Layer-2

Matt Pearing



THE STATE OF ARBITRUM

OVERVIEW



\$2.5B+

TOTAL VALUE
LOCKED



51%+

ROLLUP MARKET
SHARE



400+

DAPP
INTEGRATIONS



1.4M+

UNIQUE
ADDRESSES

BLUE CHIPS



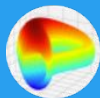
UNISWAP



SUSHISWAP



AAVE



CURVE



YEARN

NATIVE



DOPEX



VESTA
FINANCE



SPERAX



GMX



TRACER

INFRASTRUCTURE



CHAINLINK



SAFE



INFURA



THE GRAPH



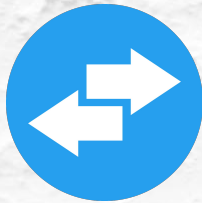
ALCHEMY



ETHERSCAN

A FEW AREAS OF FOCUS

OVERVIEW



**BRIDGING UX
FOR EVERYONE**



**CONTINUOUS
COMPATIBILITY**



**SCALING,
GENERALLY**

ZERO DEGREES OF SEPARATION

BRIDGING



ZERO DEGREES OF SEPARATION

BRIDGING



THE STATE OF BRIDGING

Bridging assets continues to be
challenging (and risky)

ZERO DEGREES OF SEPARATION

BRIDGING



THE STATE OF BRIDGING

Bridging assets continues to be
challenging (and risky)

UX for users is tricky (or impossible)
depending on the asset

ZERO DEGREES OF SEPARATION

BRIDGING



THE STATE OF BRIDGING

Bridging assets continues to be
challenging (and risky)

UX for users is tricky (or impossible)
depending on the asset

Not all token implementations are
created equal
I.e. standard vs. custom bridging

ZERO DEGREES OF SEPARATION

BRIDGING

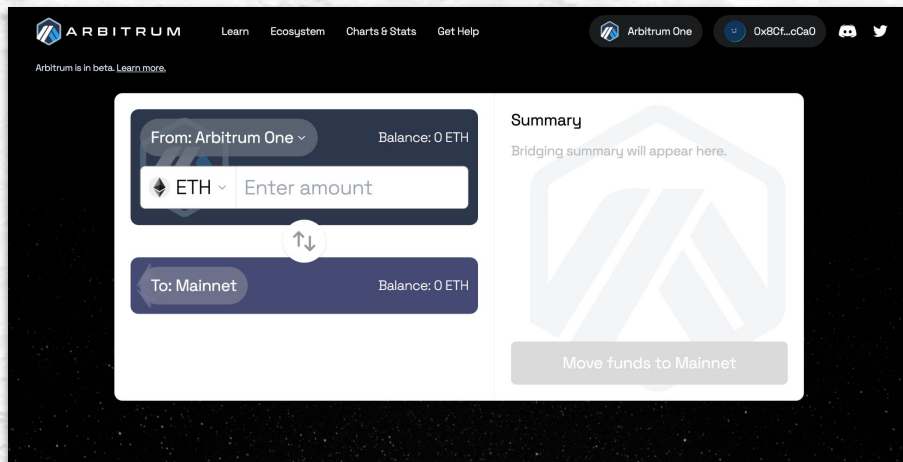


THE STATE OF BRIDGING

Bridging assets continues to be
challenging (and risky)

UX for users is tricky (or impossible)
depending on the asset

Not all token implementations are
created equal
I.e. standard vs. custom bridging



SOURCE: <https://bridge.arbitrum.io>

ZERO DEGREES OF SEPARATION

BRIDGING



WHAT'S NEXT

ZERO DEGREES OF SEPARATION

BRIDGING



WHAT'S NEXT

Fast withdrawal confirmations* (enabled
by Data Availability Committees)

ZERO DEGREES OF SEPARATION

BRIDGING



WHAT'S NEXT

Fast withdrawal confirmations* (enabled by Data Availability Committees)

Deep wallet integration with fast bridging

ZERO DEGREES OF SEPARATION

BRIDGING



WHAT'S NEXT

Fast withdrawal confirmations* (enabled by Data Availability Committees)

Deep wallet integration with fast bridging

UX Iteration: Discovery of fast bridging, and more performant UIs

ZERO DEGREES OF SEPARATION

BRIDGING



WHAT'S NEXT

Fast withdrawal confirmations* (enabled by Data Availability Committees)

Deep wallet integration with fast bridging

UX Iteration: Discovery of fast bridging, and more performant UIs

Bridge aggregation as standard functionality

SEAMLESS COMPATIBILITY

TOOLING



THE STATE OF TOOLING

SEAMLESS COMPATIBILITY

TOOLING



THE STATE OF TOOLING

On Arbitrum, drop-in compatibility is standard, all EVM programs work out of the box

SEAMLESS COMPATIBILITY

TOOLING



THE STATE OF TOOLING

On Arbitrum, drop-in compatibility is standard, all EVM programs work out of the box

Core infra across L1 is generally available on L2

SEAMLESS COMPATIBILITY

TOOLING



THE STATE OF TOOLING

On Arbitrum, drop-in compatibility is standard, all EVM programs work out of the box

Core infra across L1 is generally available on L2

SDKs and frameworks are beginning to multiply

SEAMLESS COMPATIBILITY

TOOLING

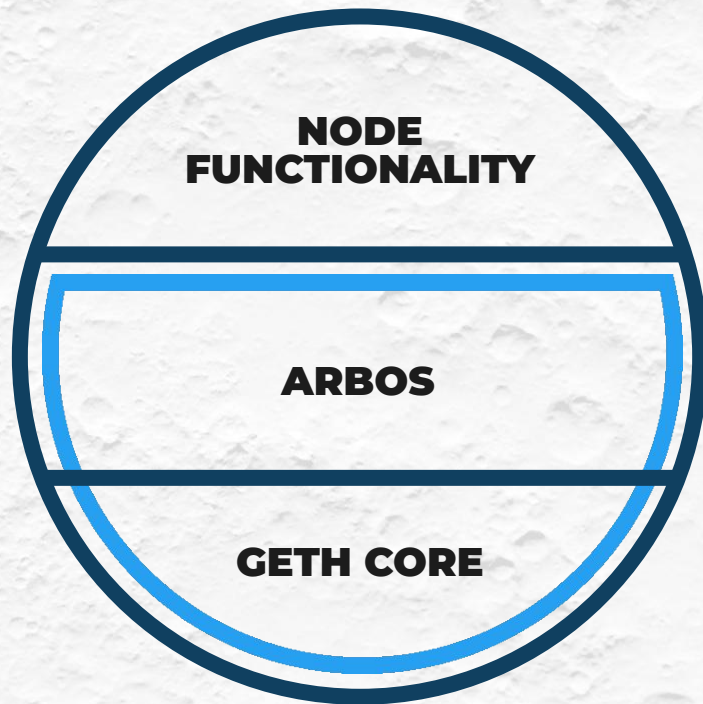


THE STATE OF TOOLING

On Arbitrum, drop-in compatibility is standard, all EVM programs work out of the box

Core infra across L1 is generally available on L2

SDKs and frameworks are beginning to multiply



SEAMLESS COMPATIBILITY

TOOLING



WHAT'S NEXT

SEAMLESS COMPATIBILITY

TOOLING



WHAT'S NEXT

Broader language support for smart
contract execution

SEAMLESS COMPATIBILITY

TOOLING



WHAT'S NEXT

Broader language support for smart contract execution

Continuous improvement on gas pricing and estimation

SEAMLESS COMPATIBILITY

TOOLING



WHAT'S NEXT

Broader language support for smart contract execution

Continuous improvement on gas pricing and estimation

Widely agreed upon standard dev frameworks

SEAMLESS COMPATIBILITY

TOOLING



WHAT'S NEXT

Broader language support for smart contract execution

Continuous improvement on gas pricing and estimation

Widely agreed upon standard dev frameworks

Long-term: more web2 -> web3 rails and rants

PERFORMANCE = ADOPTION

SCALING



THE STATE OF SCALE

PERFORMANCE = ADOPTION

SCALING



THE STATE OF SCALE

Teams are making strides in
performance and throughput every
week

PERFORMANCE = ADOPTION

SCALING



THE STATE OF SCALE

Teams are making strides in
performance and throughput every
week

Arbitrum Nitro shipped just last
month, 7x'ing our capacity

PERFORMANCE = ADOPTION

SCALING



THE STATE OF SCALE

Teams are making strides in performance and throughput every week

Arbitrum Nitro shipped just last month, 7x'ing our capacity

Efficiency is crucial across rollup systems from sequencing, through execution, to validation

PERFORMANCE = ADOPTION

SCALING












THE STATE OF SCALE

Teams are making strides in performance and throughput every week

Arbitrum Nitro shipped just last month, 7x'ing our capacity

Efficiency is crucial across rollup systems from sequencing, through execution, to validation

Name	Send ETH	Swap tokens
Metis Network 	< \$0.01	\$0.05 ▾
 Loopring	\$0.03	\$0.32 ▾
 Arbitrum One	\$0.03	\$0.10 ▾
 ZKSync	\$0.04	\$0.09 ▾
 Optimism	\$0.11	\$0.17 ▾
 Boba Network	\$0.18	\$0.31 ▾
 Polygon Hermez	\$0.25	- ▾
 Aztec Network	\$0.28	- ▾
 Ethereum	\$0.70	\$3.49 ▾

SOURCE: <https://l2fees.info>

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

Node architecture must continue to evolve

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

Node architecture must continue to evolve

Client Optimization
(Erigon, etc.)

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

Node architecture must continue to evolve

Layer 2s and Layer 1 mainly dealing with the same set of problems / core bottleneck

Client Optimization
(Erigon, etc.)

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

Node architecture must continue to evolve

Client Optimization
(Erigon, etc.)

Layer 2s and Layer 1 mainly dealing with the
same set of problems / core bottleneck

Rentable Storage

MANAGING GROWTH

SCALING



DEALING WITH STATE BLOAT

Node architecture must continue to evolve

Client Optimization
(Erigon, etc.)

Layer 2s and Layer 1 mainly dealing with the
same set of problems / core bottleneck

Rentable Storage

State Expiry &
Regenesiis

MANAGING GROWTH

SCALING



GROWING STATE USAGE

MANAGING GROWTH

SCALING



GROWING STATE USAGE

Emerging use cases,
with lower barriers to entry

(i.e. Games with less pay-to-win)

MANAGING GROWTH

SCALING



GROWING STATE USAGE

Emerging use cases,
with lower barriers to entry
(i.e. Games with less pay-to-win)

Continuous iteration of DeFi onboarding

MANAGING GROWTH

SCALING



GROWING STATE USAGE

Emerging use cases,
with lower barriers to entry
(i.e. Games with less pay-to-win)

Continuous iteration of DeFi onboarding

Insurance as a core primitive



ARBITRUM

