smartFHE: Privacy-Preserving Smart Contracts from Fully Homomorphic Encryption

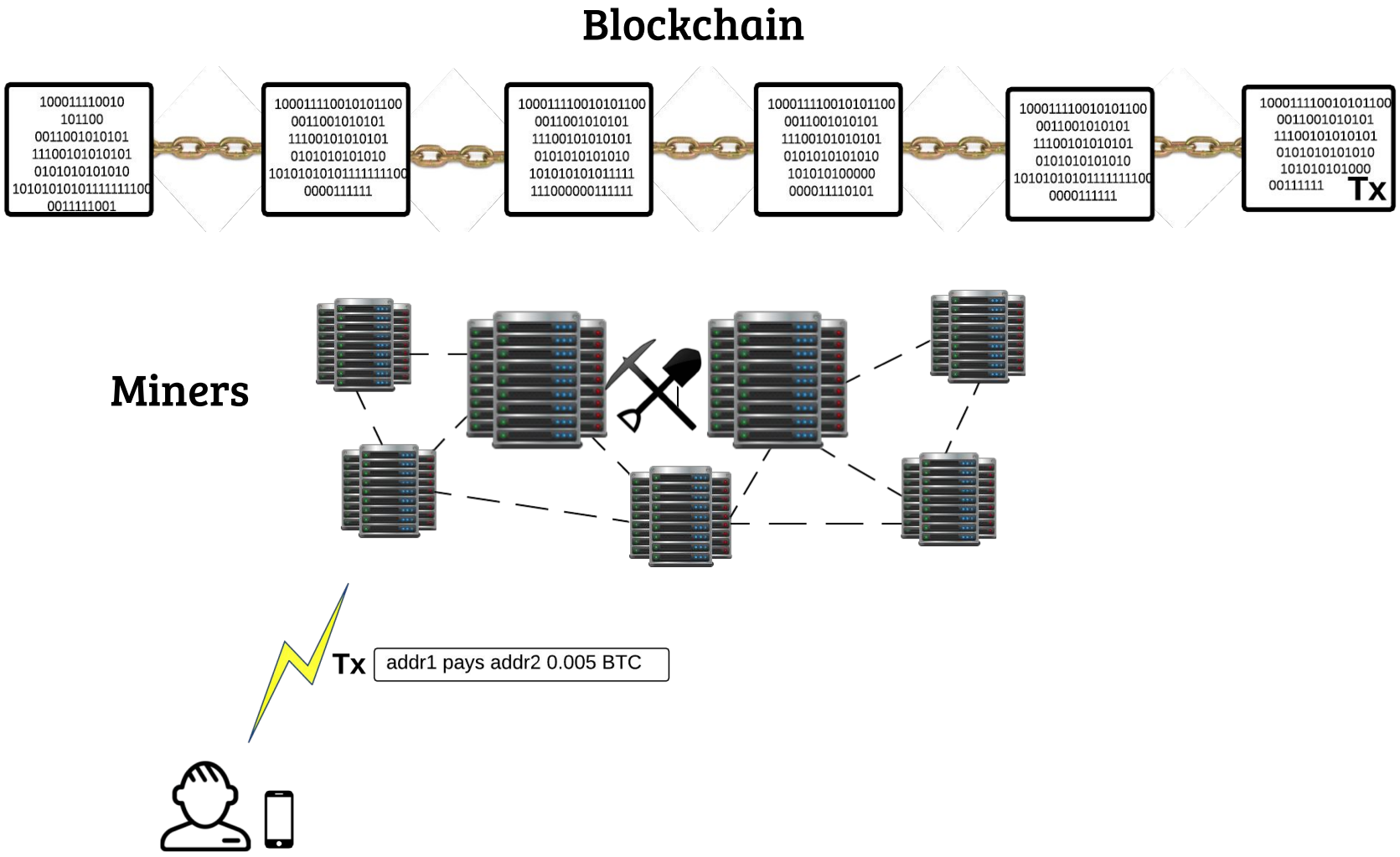
Ravital Solomon[1], Rick Weber[1], **Ghada Almashaqbeh[2]**

[1]Sunscreen, [2]University of Connecticut

EuroS&P 2023

Big Dreams …

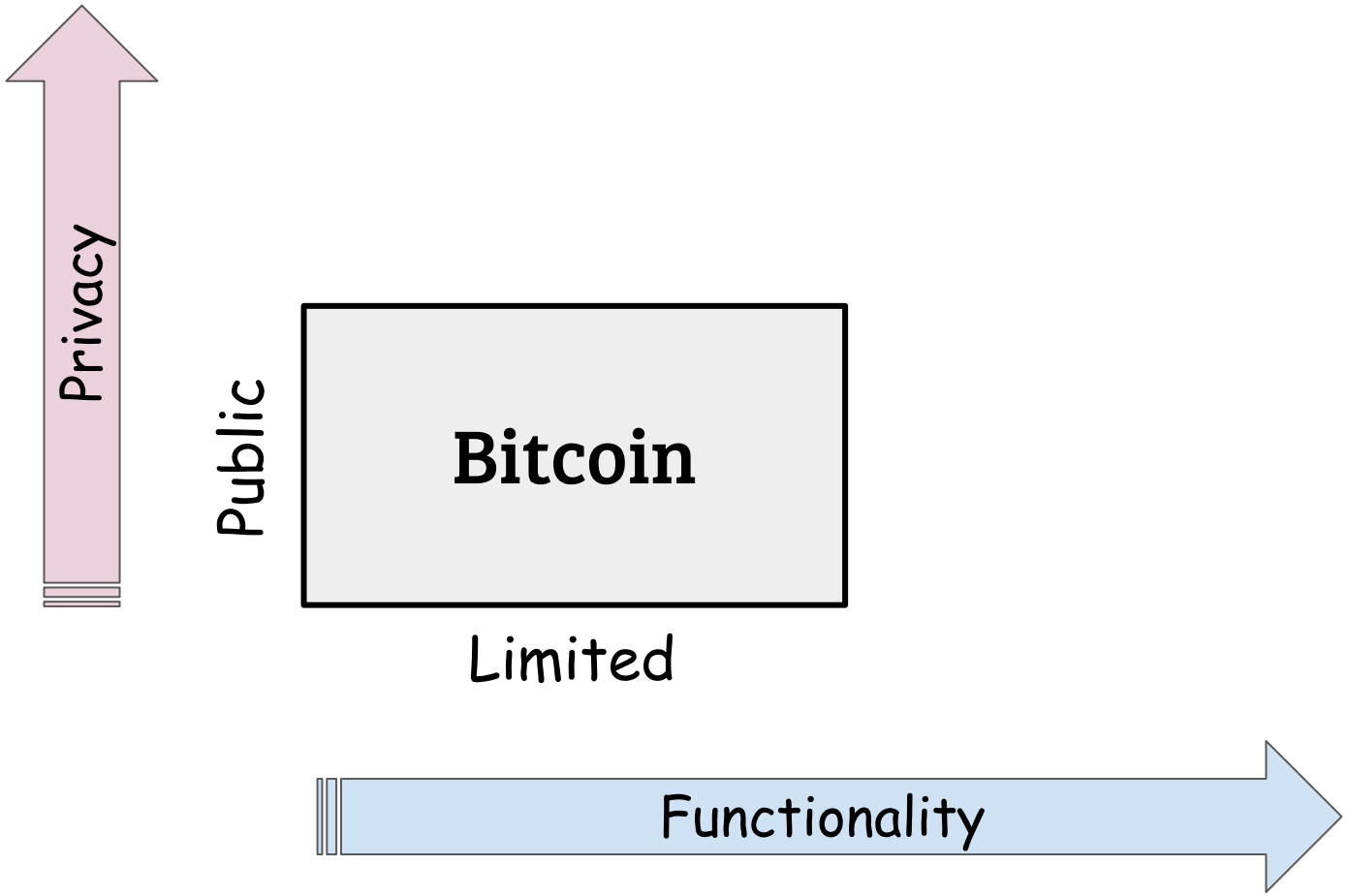
Bitcoin 2009



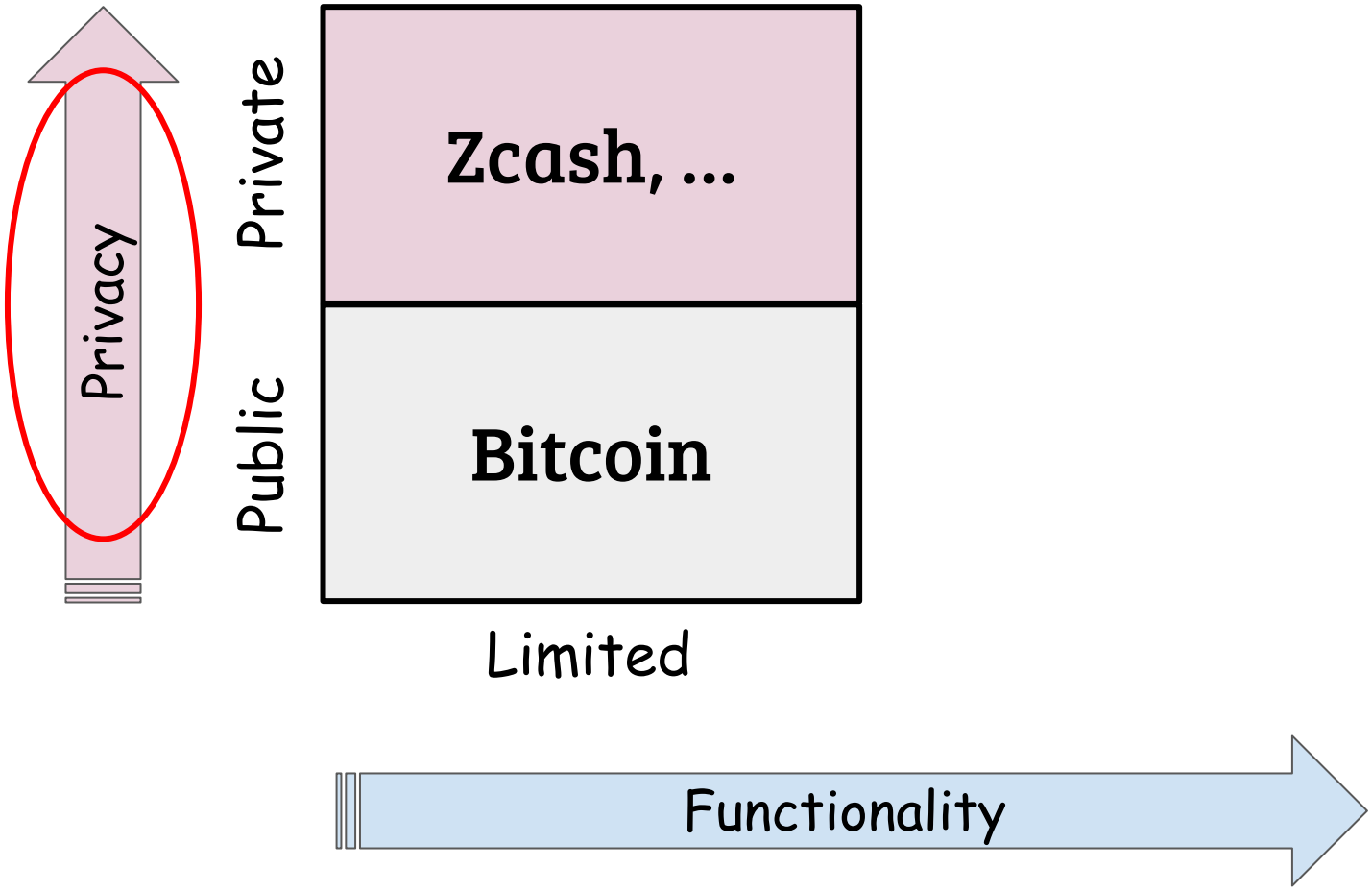
Limited functionality

No privacy

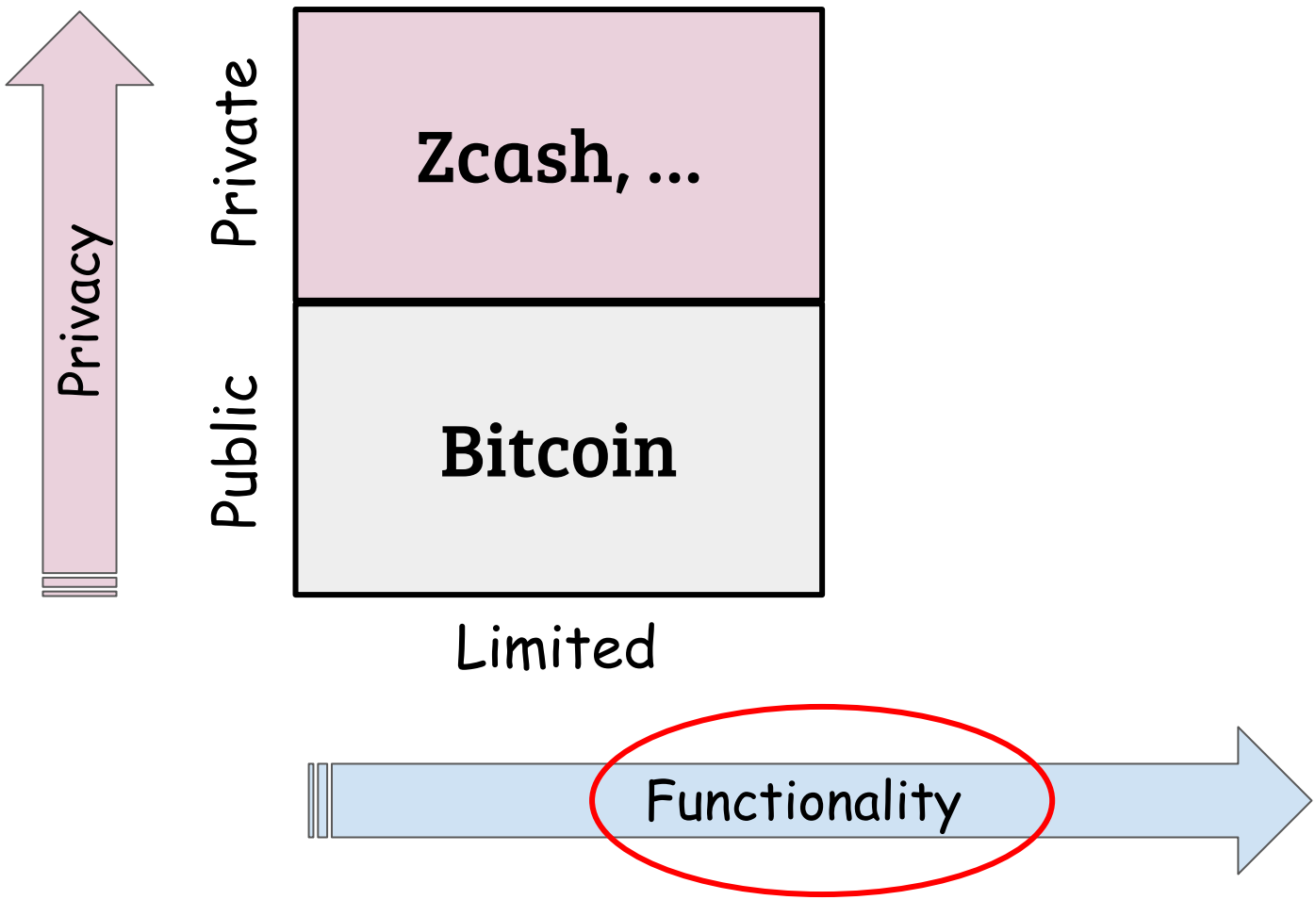
Solutions Went Different Directions



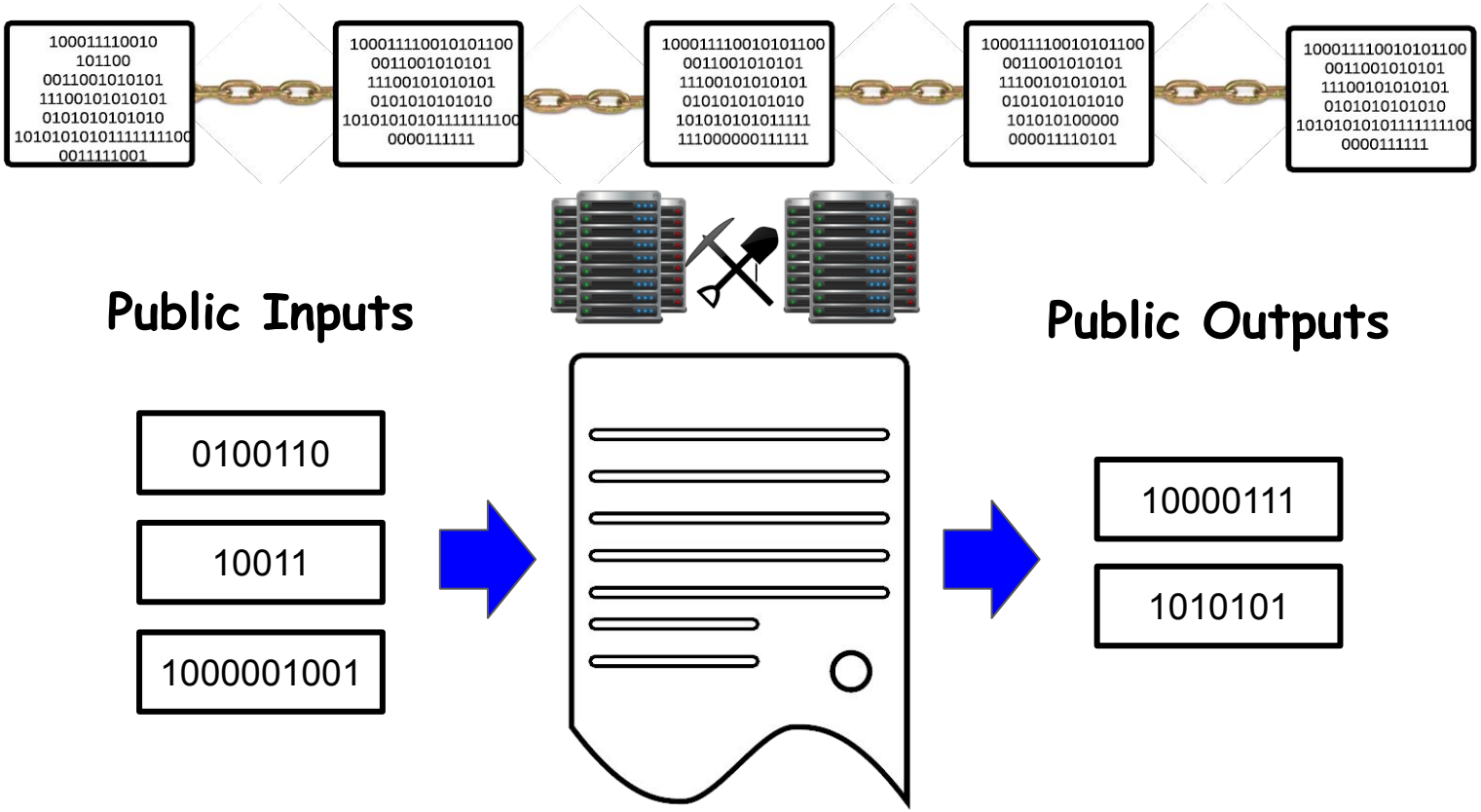
Solutions Went Different Directions



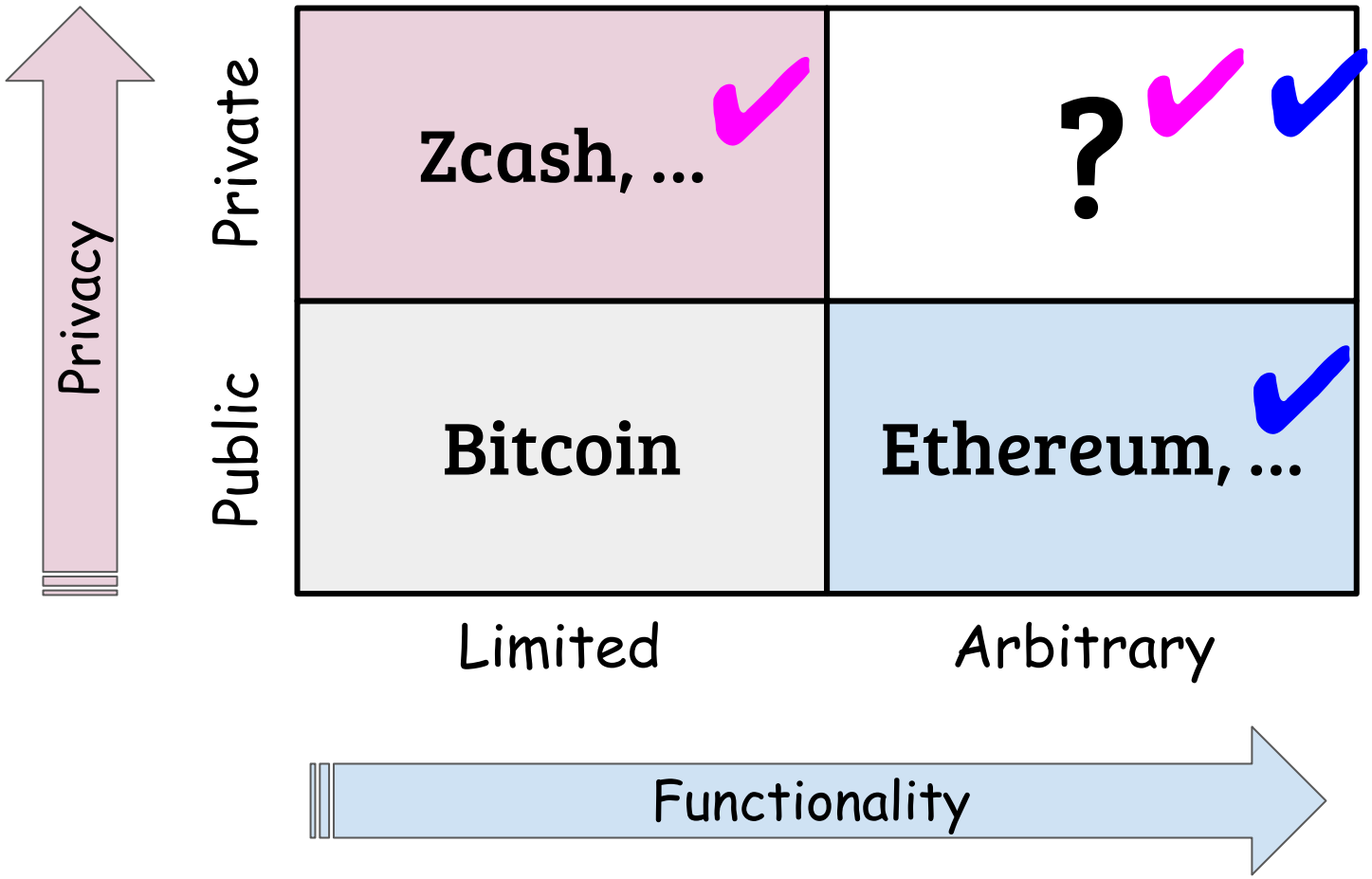
Solutions Went Different Directions



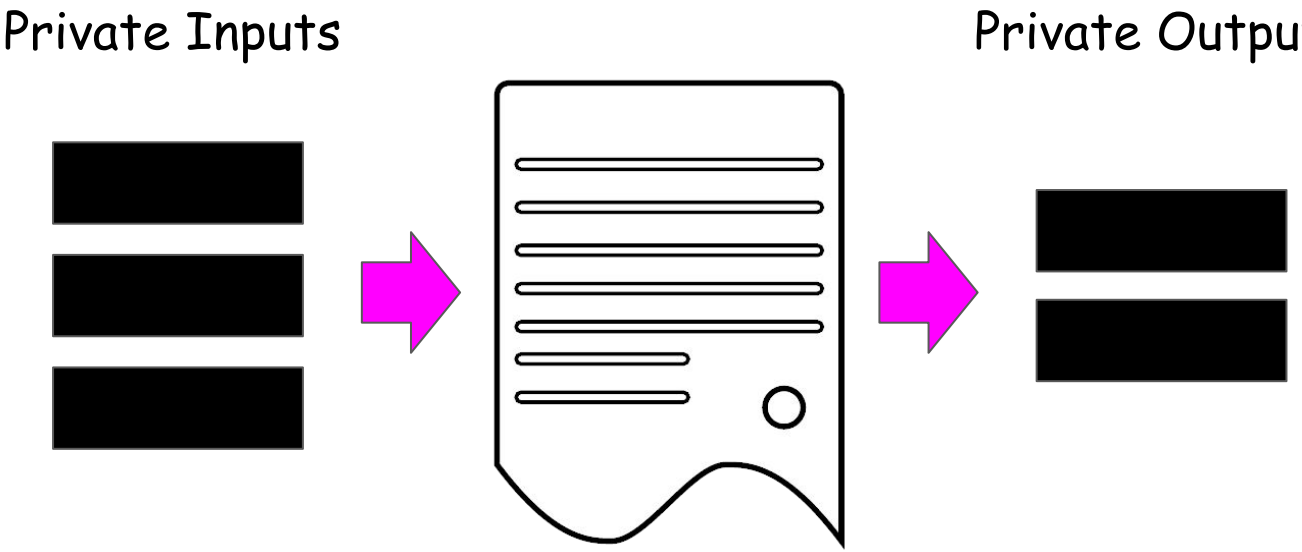
Ethereum was Born in 2015



Bigger Dreams ...



Privacy-preserving Smart Contracts?



More Initiatives

Hawk

Zether

Kachina

Zexe

Ekiden

Zkay

Arbitrum

More Initiatives, But …

Hawk

Zether

Kachina

Zexe

Ekiden

Zkay

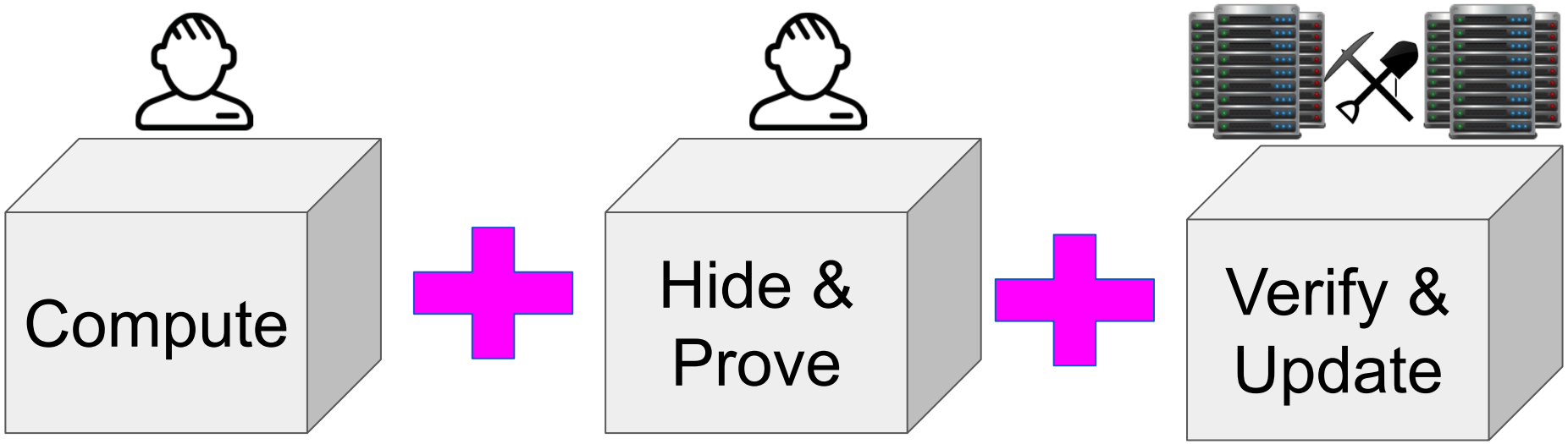
Arbitrum

Limited Functionality!

Overload users!

ZKP-based Approach (Not Us)

Off-chain Private Computing

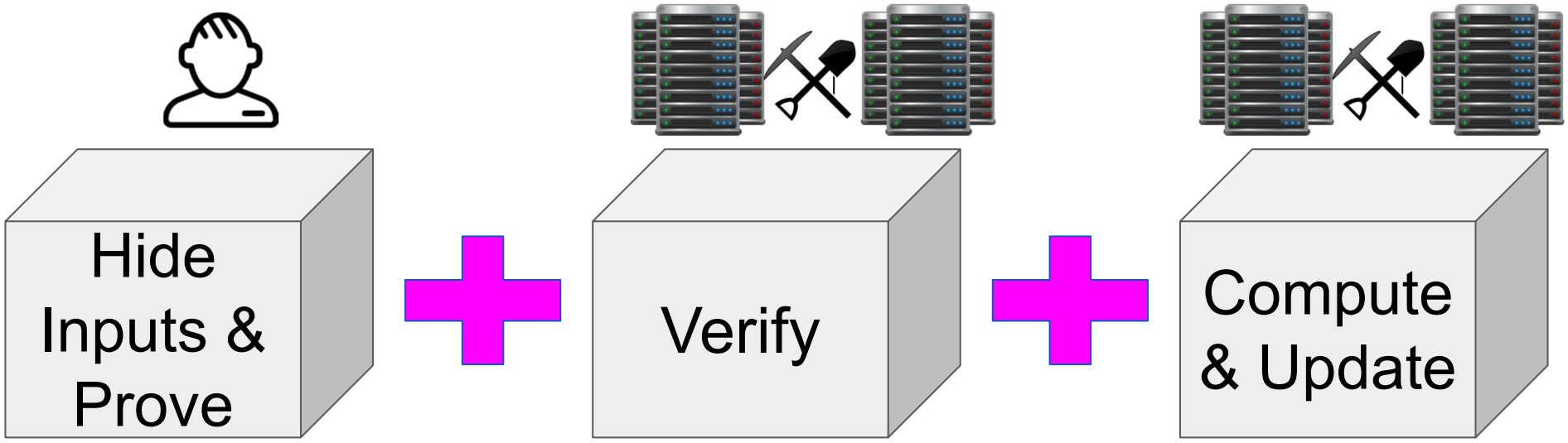


Zero Knowledge Proofs (ZKPs)

Not suited for lightweight users

Our Goal

On-chain Private Computing



Fully Homomorphic Encryption (FHE) +
Zero Knowledge Proof (ZKPs)

Contributions

A formal notion for privacy-preserving smart contracts (PPSCs) capturing arbitrary computation with I/O privacy.

Contributions

A formal notion for privacy-preserving smart contracts (PPSCs) capturing arbitrary computation with I/O privacy.

smartFHE framework: the first scheme to use FHE in the blockchain model!

Contributions

A formal notion for privacy-preserving smart contracts (PPSCs) capturing arbitrary computation with I/O privacy.

smartFHE framework: the first scheme to use FHE in the blockchain model!

smartFHE instantiation.

Contributions

A formal notion for privacy-preserving smart contracts (PPSCs) capturing arbitrary computation with I/O privacy.

smartFHE framework: the first scheme to use FHE in the blockchain model!

smartFHE instantiation.

Formal security proofs and implementation/benchmarks
- The first library for short-discrete log proofs

smartFHE Framework

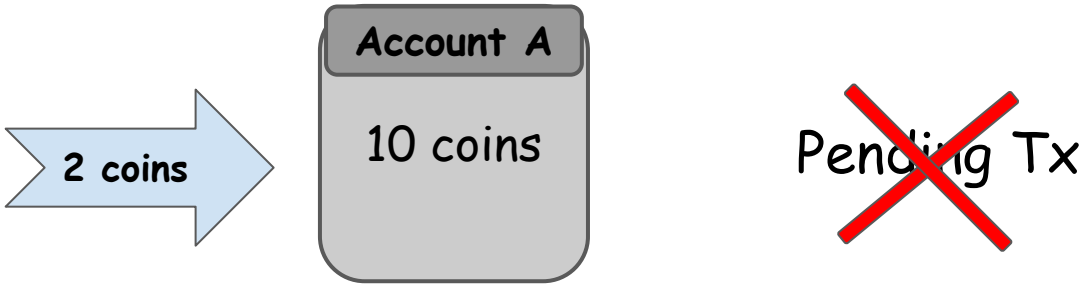
- Privacy extensions for a public smart contract-enabled blockchain, e.g., Ethereum
- Flexible and modular.
- Supports:
    - Private and public payments
    - Private and public smart contracts
- A user can have public and private accounts

smartFHE Framework

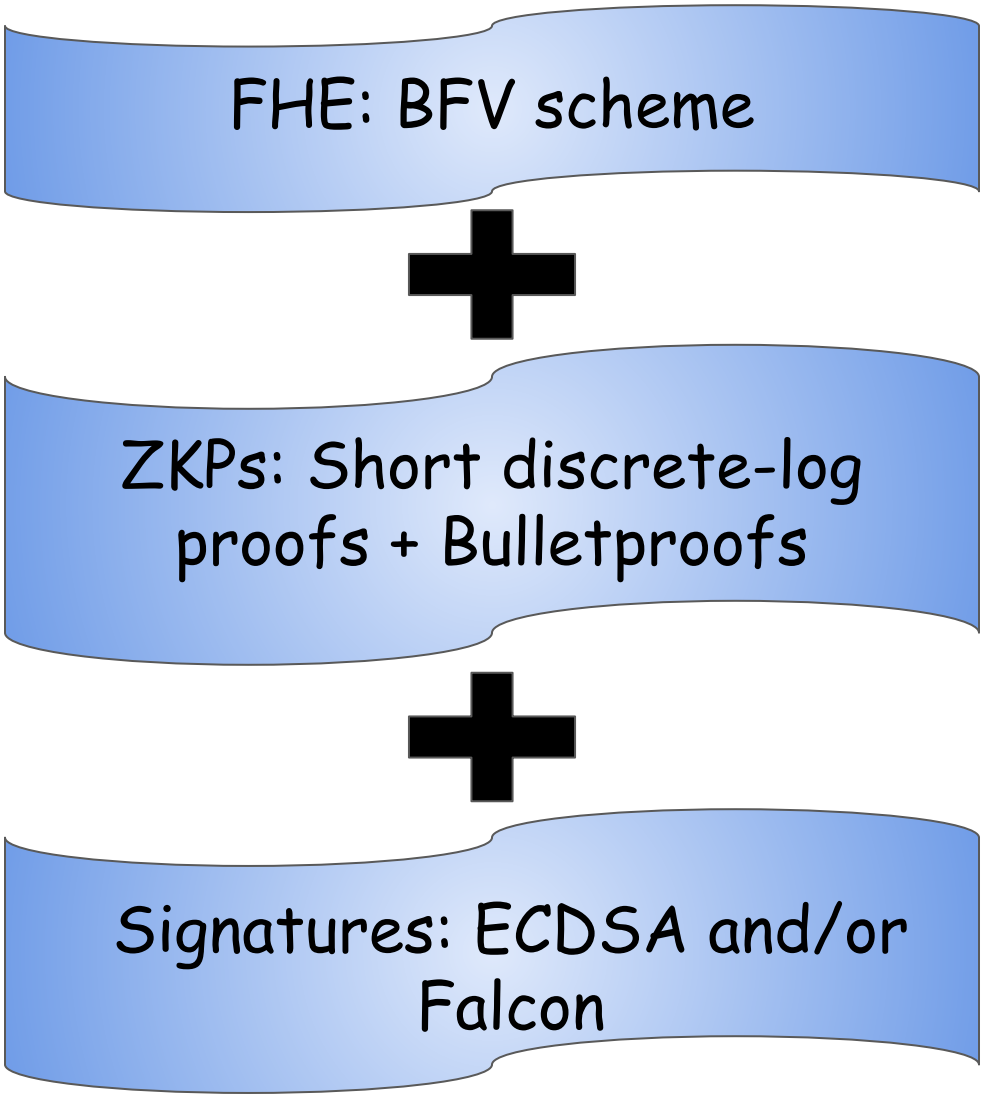
- **Network protocol operations:**
  - Private accounts: FHE keypairs and signature keypairs. Encrypted balance.
  - Private payments:
    - `Shield`
    - `PrivTransfer`
    - `Deshield`
  - Private smart contracts: contract-dependent, translated into FHE operations.

Several Challenges

- Working with FHE
- Combining FHE with ZKP
- Concurrency



smartFHE Instantiation



Implementation

- **Existing libraries:**
  - Microsoft SEAL for BFV
  - Dalek for Bulletproofs
  - OpenSSL for ECDSA
- **New library:**
  - First implementation of short-discrete log proofs with Apple Metal GPU-accelerated code.
- Benchmarks on Apple M2 Max with 64GB RAM

Results - Benchmarks

TABLE 1: Setup times (one time cost)

Performed by	**Operation**	$d = 1024$	$d = 2048$	$d = 4096$
User	KeyGen	0.216 ms	0.375 ms	36.5 ms
System	ZKP setup	0.8 s	2.06 s	5.7 s

TABLE 2: Private transaction costs for smartFHE's instantiation—user side.

	Operation	**Time (s)**	**Size (KB)**
$d = 1024$	$\mathsf{Shield}(\mathsf{tx}_{\mathsf{shield}})$	0.0002	0.101
	$\mathsf{Deshield}(\mathsf{tx}_{\mathsf{deshield}})$	1.89	2.47
	$\mathsf{PrivTransfer}(\mathsf{tx}_{\mathsf{privtransf}})$	3.57	20.03
$d = 2048$	$\mathsf{Shield}(\mathsf{tx}_{\mathsf{shield}})$	0.0002	0.101
	$\mathsf{Deshield}(\mathsf{tx}_{\mathsf{deshield}})$	3.58	2.53
	$\mathsf{PrivTransfer}(\mathsf{tx}_{\mathsf{privtransf}})$	10.7	64.76
$d = 4096$	$\mathsf{Shield}(\mathsf{tx}_{\mathsf{shield}})$	0.0002	0.101
	$\mathsf{Deshield}(\mathsf{tx}_{\mathsf{deshield}})$	11.17	2.66
	$\mathsf{PrivTransfer}(\mathsf{tx}_{\mathsf{privtransf}})$	23.89	180.1

Results - Benchmarks

TABLE 3: Private transaction costs for smartFHE's instantiation—miner side.

	Operation	**Time (s)**
$d = 1024$	VerifyShield	0.00017
	VerifyDeshield	0.92
	VerifyPrivTransfer	1.95
$d = 2048$	VerifyShield	0.00017
	VerifyDeshield	1.92
	VerifyPrivTransfer	6.37
$d = 4096$	VerifyShield	0.00017
	VerifyDeshield	6.42
	VerifyPrivTransfer	14.77

Results - Comparison

TABLE 4: Base private transaction costs for Veri-zexe.

no. of inputs × no. of outputs	User generation time (s)	Miner verification time (ms)	Size (KB)
2 × 2	27.82	13.21	4.82
3 × 3	54.9	13.14	4.88
4 × 4	59	13.15	4.95
8 × 8	121	13.15	5.2

smartHE allows a user to issue payments at a rate 1.16x - 7.79x faster than Veri-zexe

Results - Applications

TABLE 5: Private smart contract application costs.

Application	**Per user generation time (s)**	**Miner verification/computing time (s)**	**Size (KB) per user**
AMM ($d = 2048$)	6.4	3.58	33.53
AMM ($d = 4096$)	20.88	12.64	91.4
Mean/variance ($d = 4096$)	20.89	62.9	91.45
Chi-squared ($d = 4096$)	23.89	44.39	26.95

Conclusion and Future Work

- **This work**
  - A privacy-preserving smart contract framework (and instantiation) from FHE and ZKP
  - Formal treatment
  - Implementation/testing
- **Future work**
  - Look into instantiations using other FHE/ZKP schemes
  - Addressing anonymity
  - Handling storage cost

Thank you!

Questions?