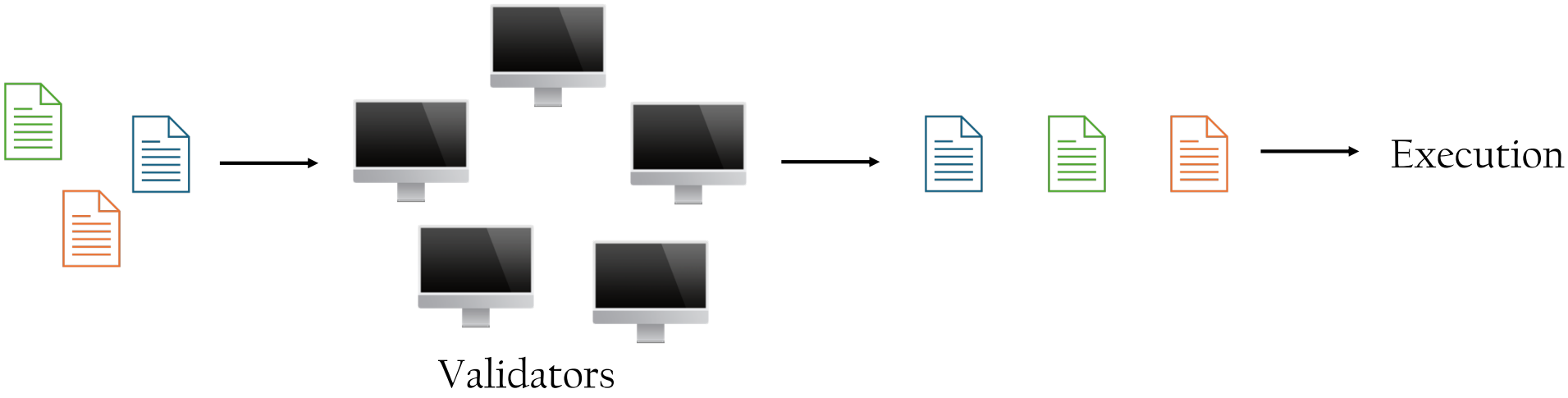
Data Availability Sampling with Repair

Dan Boneh, Joachim Neu, Valeria Nikolaenko, **Aditi Partap**

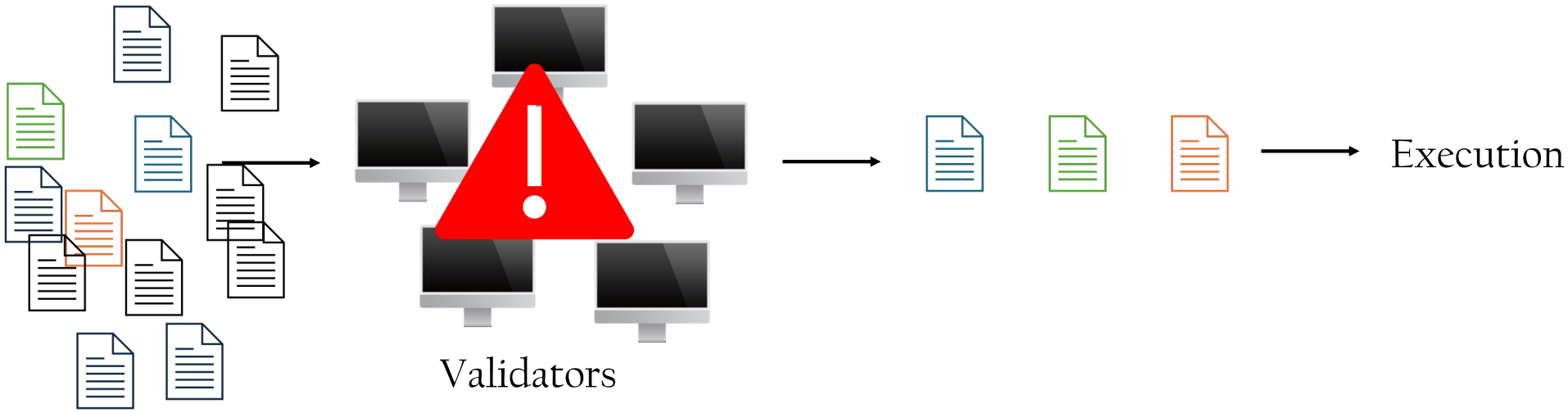

Stanford University

a16zcrypto

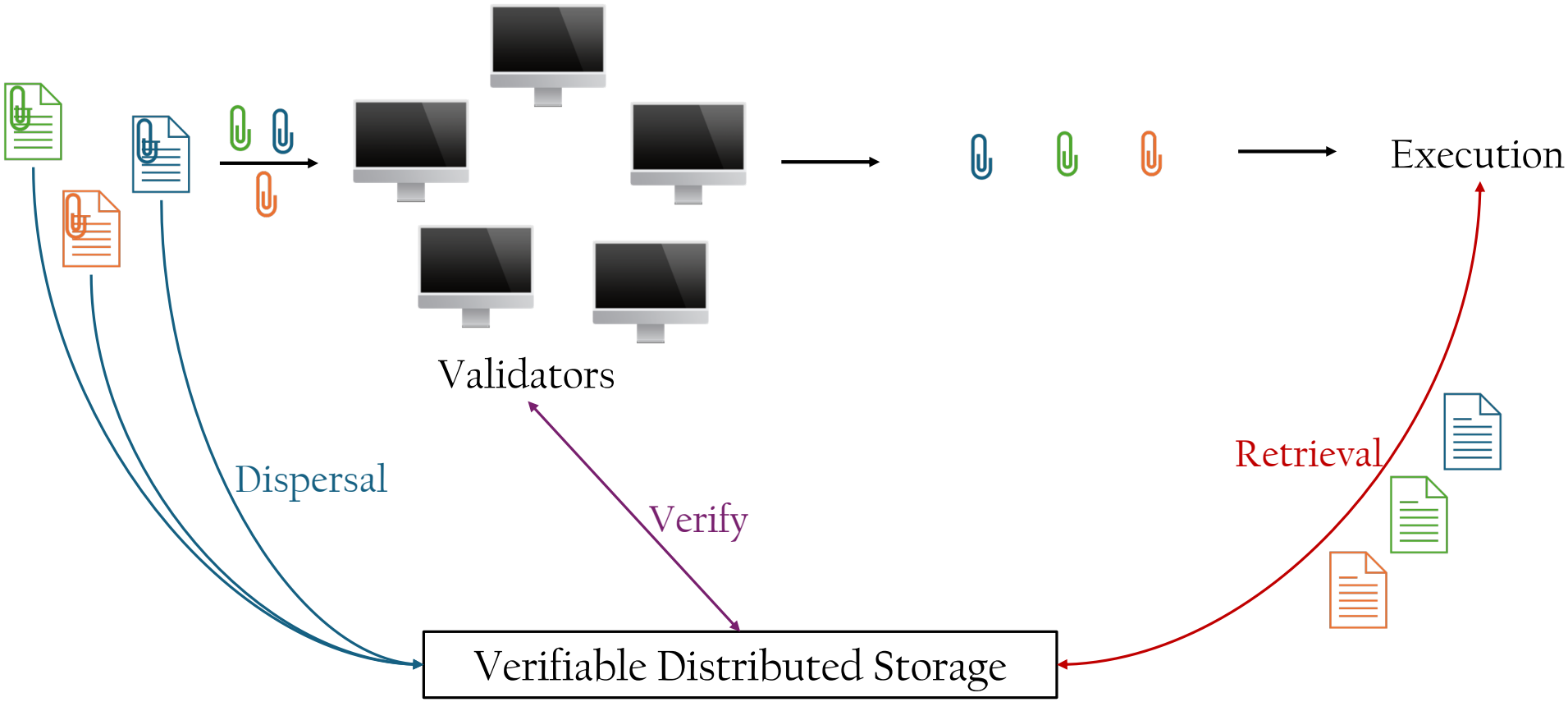
The need to scale Consensus



The need to scale Consensus

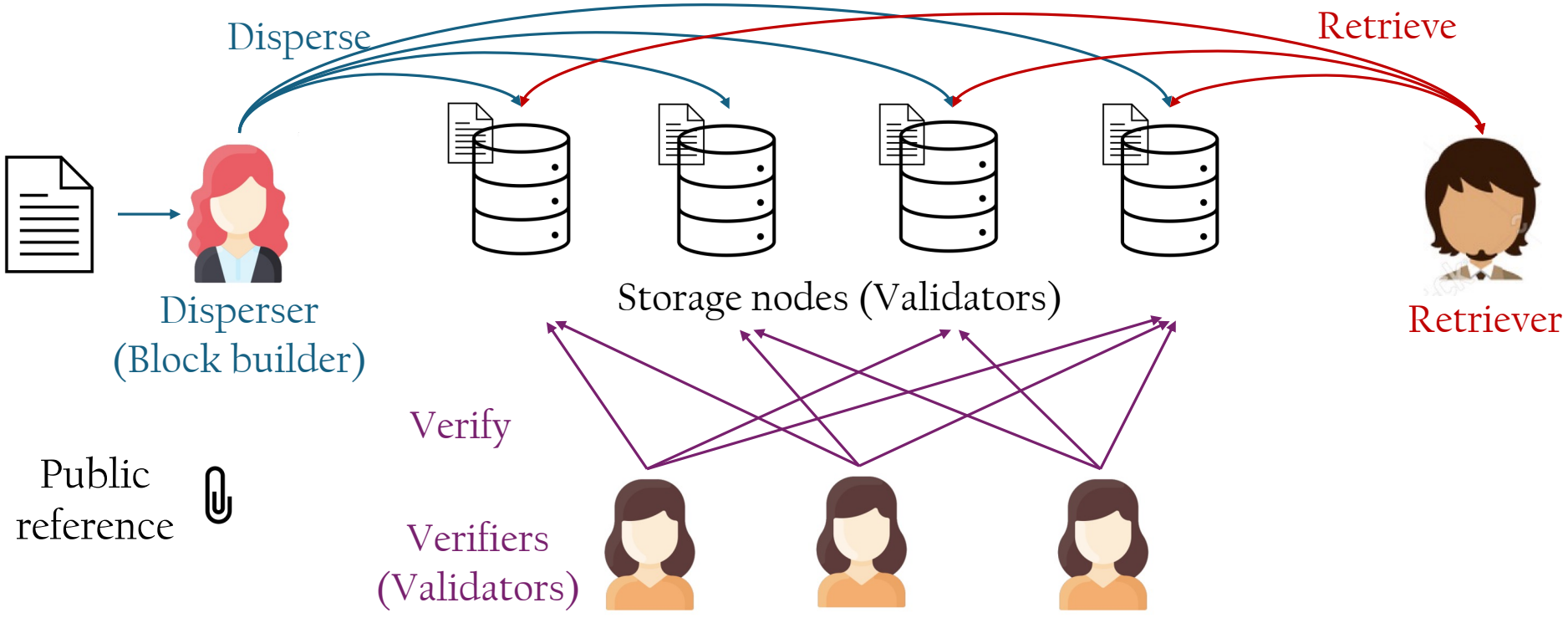


Solution: Decouple Ordering from Dissemination



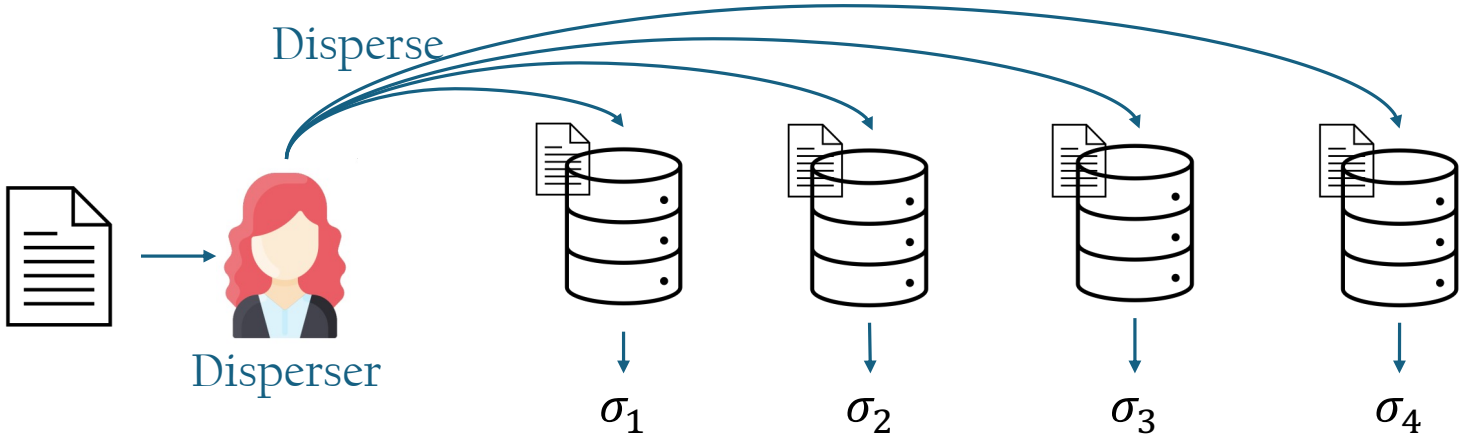
Verifiable Distributed Storage

Verifiably store data among storage nodes, so that each node communicates and stores only a small part



Example: Verifiable Information Dispersal [CT'05,...]

Verify is non-interactive:



Used in:

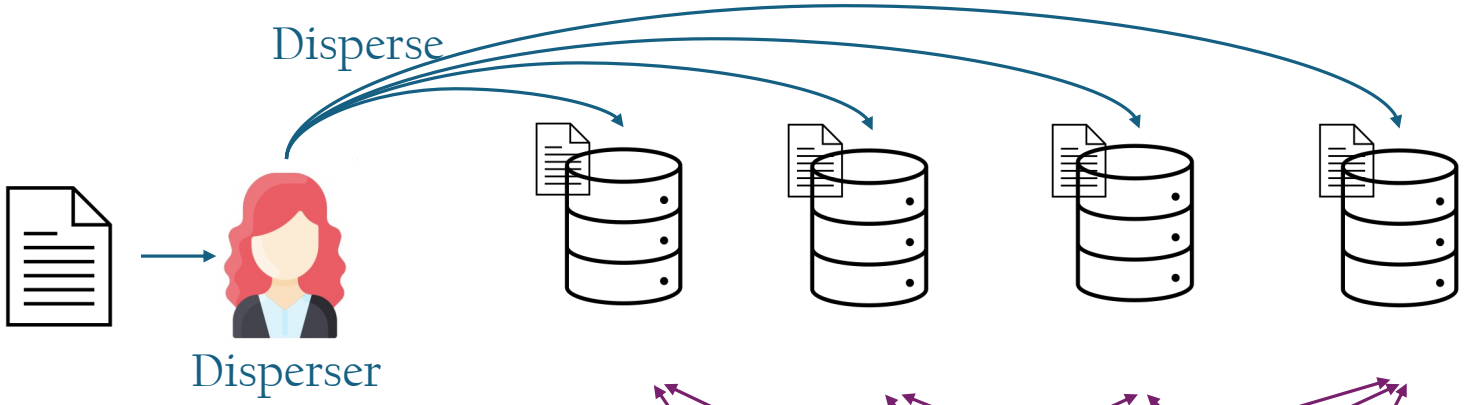
EigenDA

Espresso

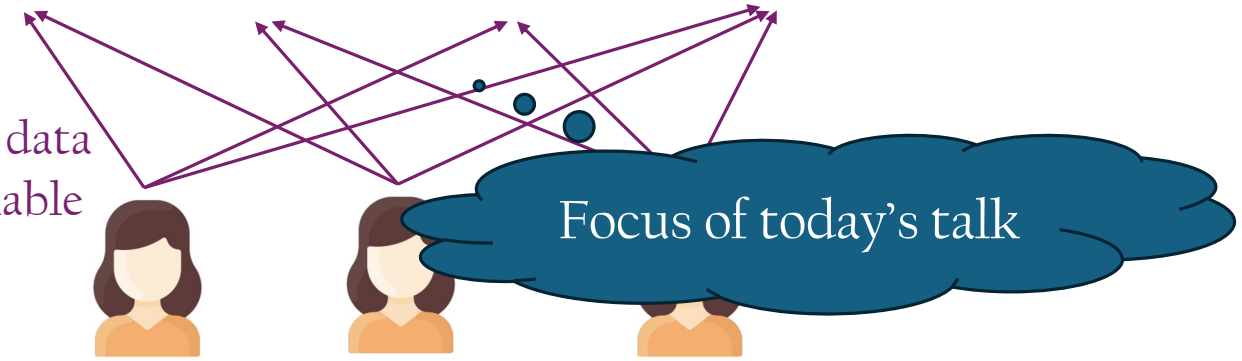
Verify: Output 1 if enough valid signatures

Example: Data Availability Sampling [HASW'23,...]

Verify is interactive:



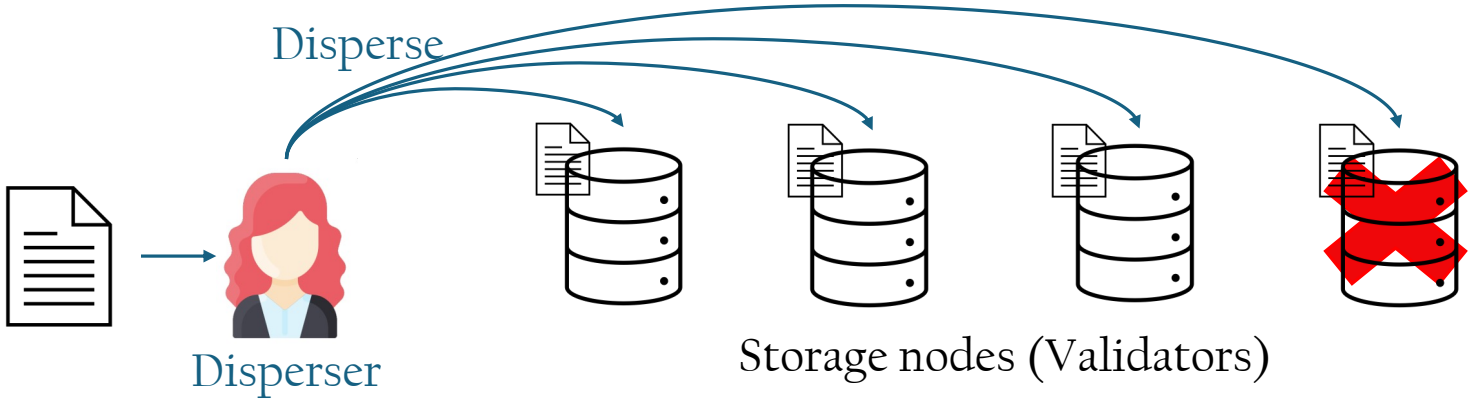
Verify:
'Sample' random chunks of data
Output 1 if all chunks available



Used in:

How to repair lost chunks?

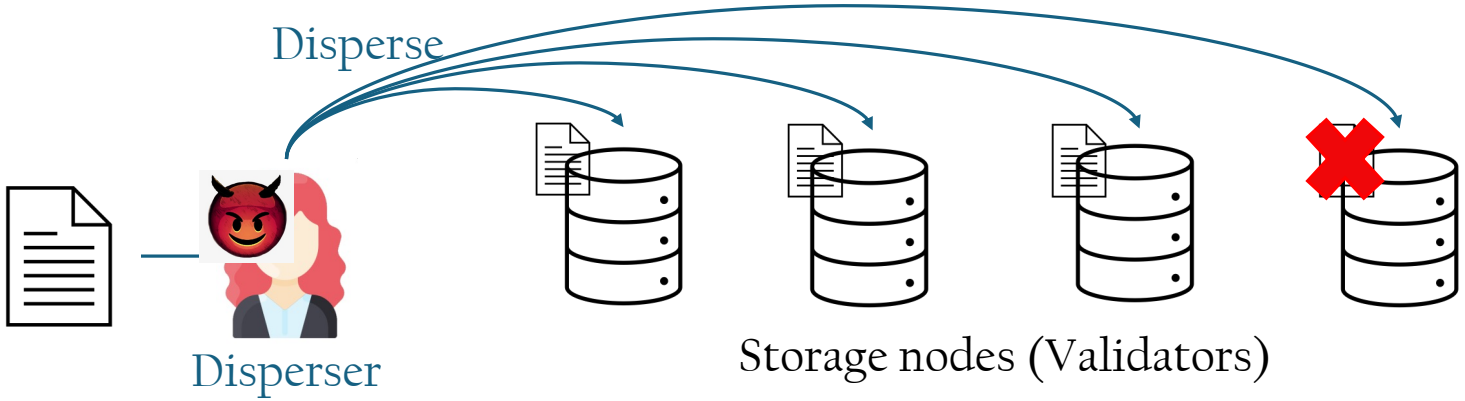
What if some node loses its chunk?



How to repair lost chunks?

What if some node loses its chunk?

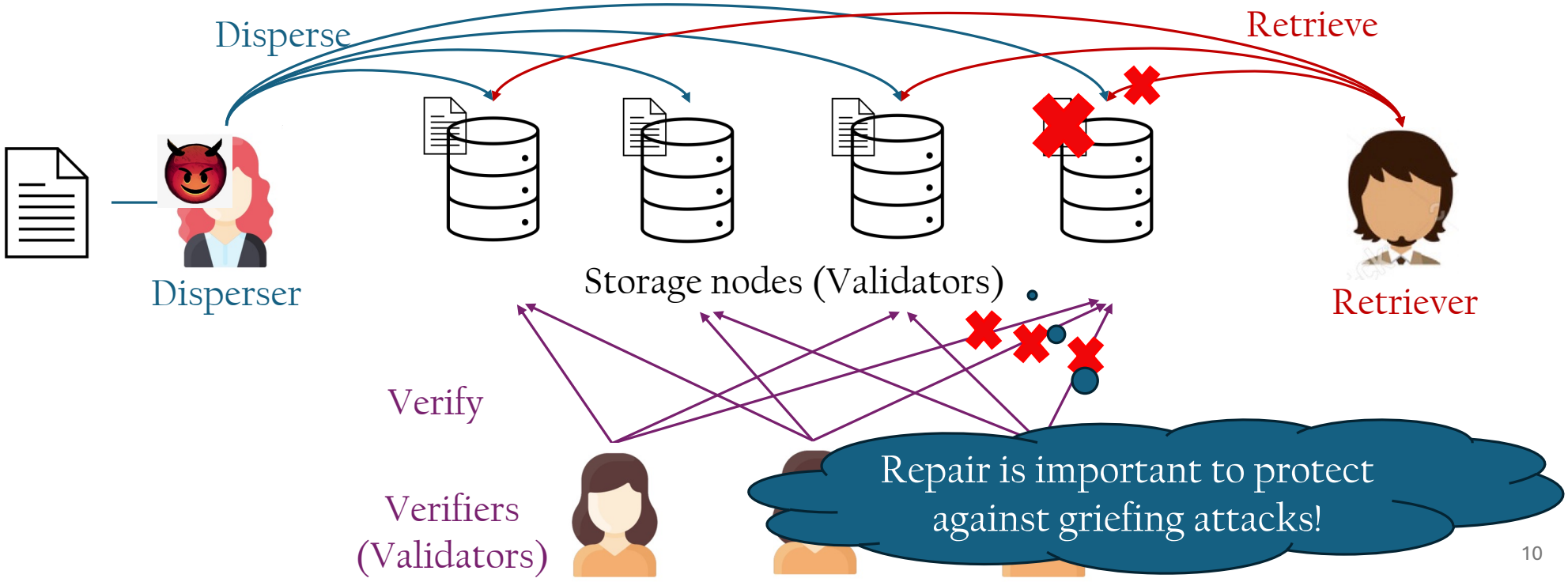
Or, what if the disperser never dispersed its chunk?



How to repair lost chunks?

What if some node loses its chunk?

Or, what if the disperser never dispersed its chunk?



This work

- New definitions for Data Availability Sampling (DAS), strengthening Hall-Anderson,Simkin,Wagner'23 to formalize:
  - Local Repair
  - Retrieval
- Enhance DAS framework from HASW'23 for building DAS with local repair, from

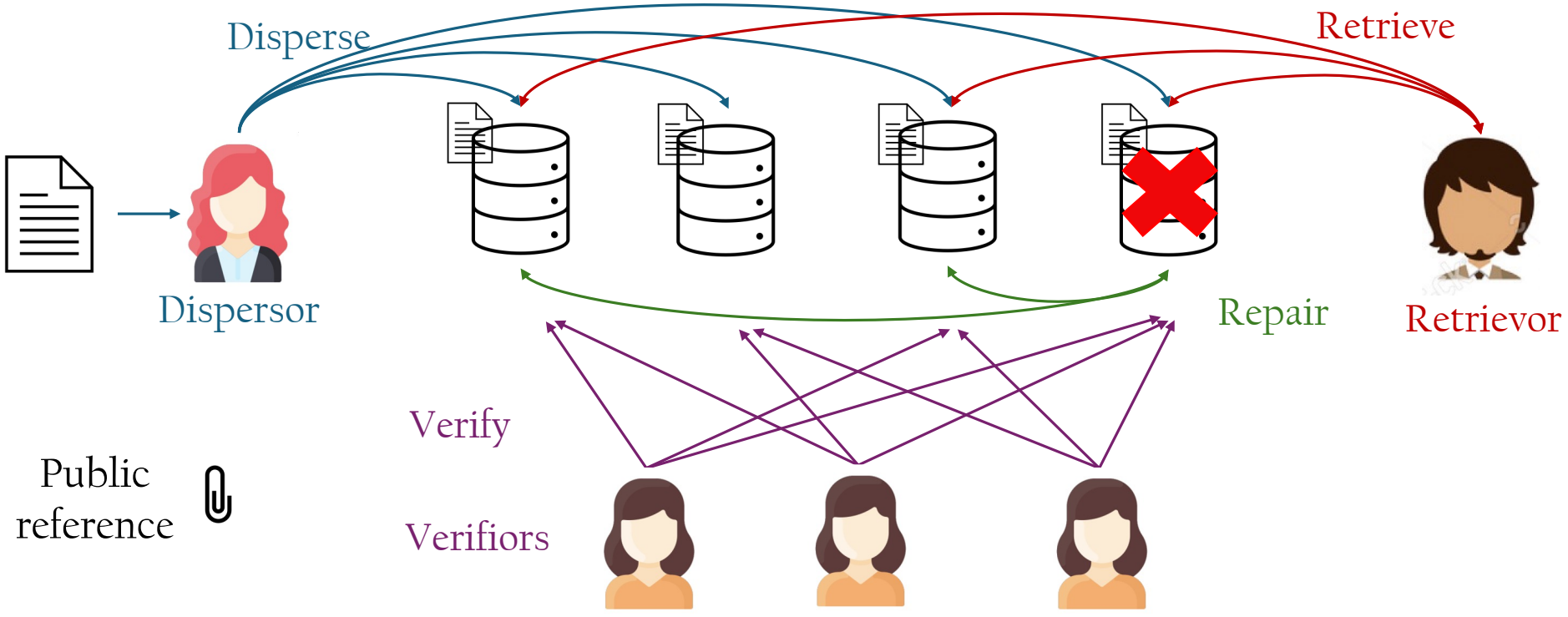
Codes with locality + Erasure code commitment with locality

- New Construction using the framework with

Multiplicity codes + New Polynomial commitment scheme

Defining Local Repair for Data Availability Sampling

Repair protocol:



Our Construction

	Ethereum Fulu DAS	Our Construction
Node storage	8.18 KB	0.23 KB [35x better]

Our Construction

	Ethereum Fulu DAS	Our Construction
Node storage	8.18 KB	0.23 KB [35x better]
Local Repair complexity: Total bandwidth	128 KB	21.3 KB [6x better]
Local Repair complexity: Number of subnets	64	91 [1.4x worse]

P2P Gossip network model: Each chunk is distributed in a different subnet

Our Construction

	Ethereum Fulu DAS	Our Construction
Node storage	8.18 KB	0.23 KB [35x better]
Local Repair complexity: Total bandwidth	128 KB	21.3 KB [6x better]
Local Repair complexity: Number of subnets	64	91 [1.4x worse]
Sampling bandwidth	16.37 KB*	9.14 KB [1.8x better]
Disperser work (Multi-threaded)	0.3s	0.42s [1.4x worse]

Fulu DAS does not guarantee retrieval!

This work

- New definitions for Data Availability Sampling (DAS), strengthening Hall-Anderson,Simkin,Wagner'23 to formalize:
  - Local Repair
  - Retrieval
- Enhance DAS framework from HASW'23 for building DAS with local repair, from

Codes with locality + Erasure code commitment with locality

- New Construction using the framework with

Multiplicity codes + New Polynomial commitment scheme

Framework for Data Availability Sampling

Without local repair [HASW23]:

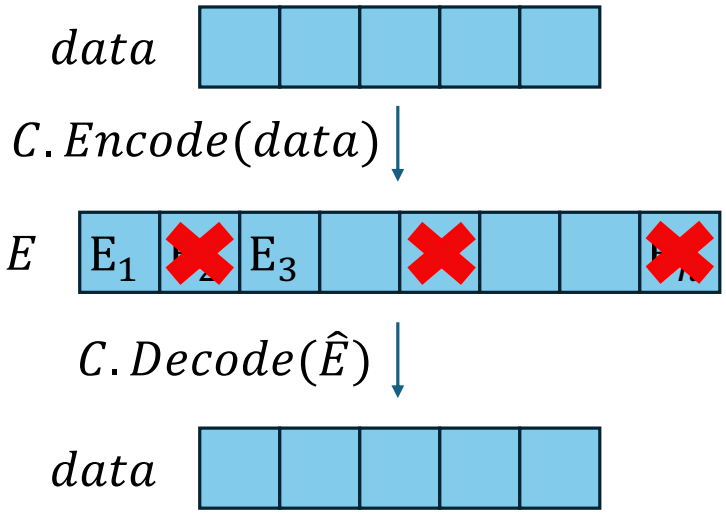
With local repair:

Building Block: Erasure Codes

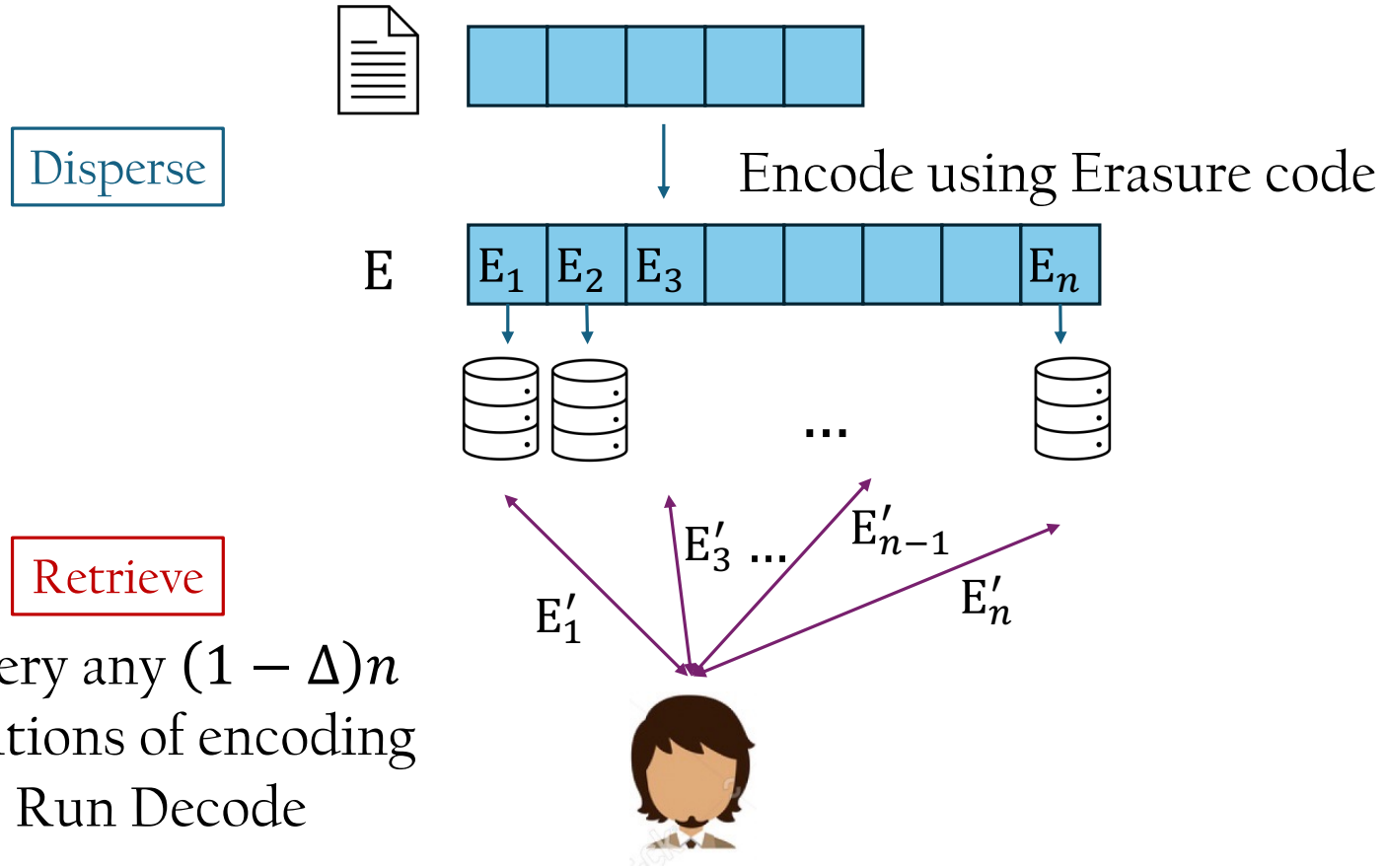
A $(k, n, \Delta) -$ Erasure Code C is a pair of algorithms:

- $Encode(data \in \Sigma^k) \quad \rightarrow \quad E \in \Gamma^n$
- $Decode(\hat{E} = \{E_i\}_{i \in S}) \quad \rightarrow \quad data \; OR \; \perp$

Wherein $Decode$ outputs the encoded data if $\hat{E}$ contains $> (1 - \Delta)n$ symbols of E



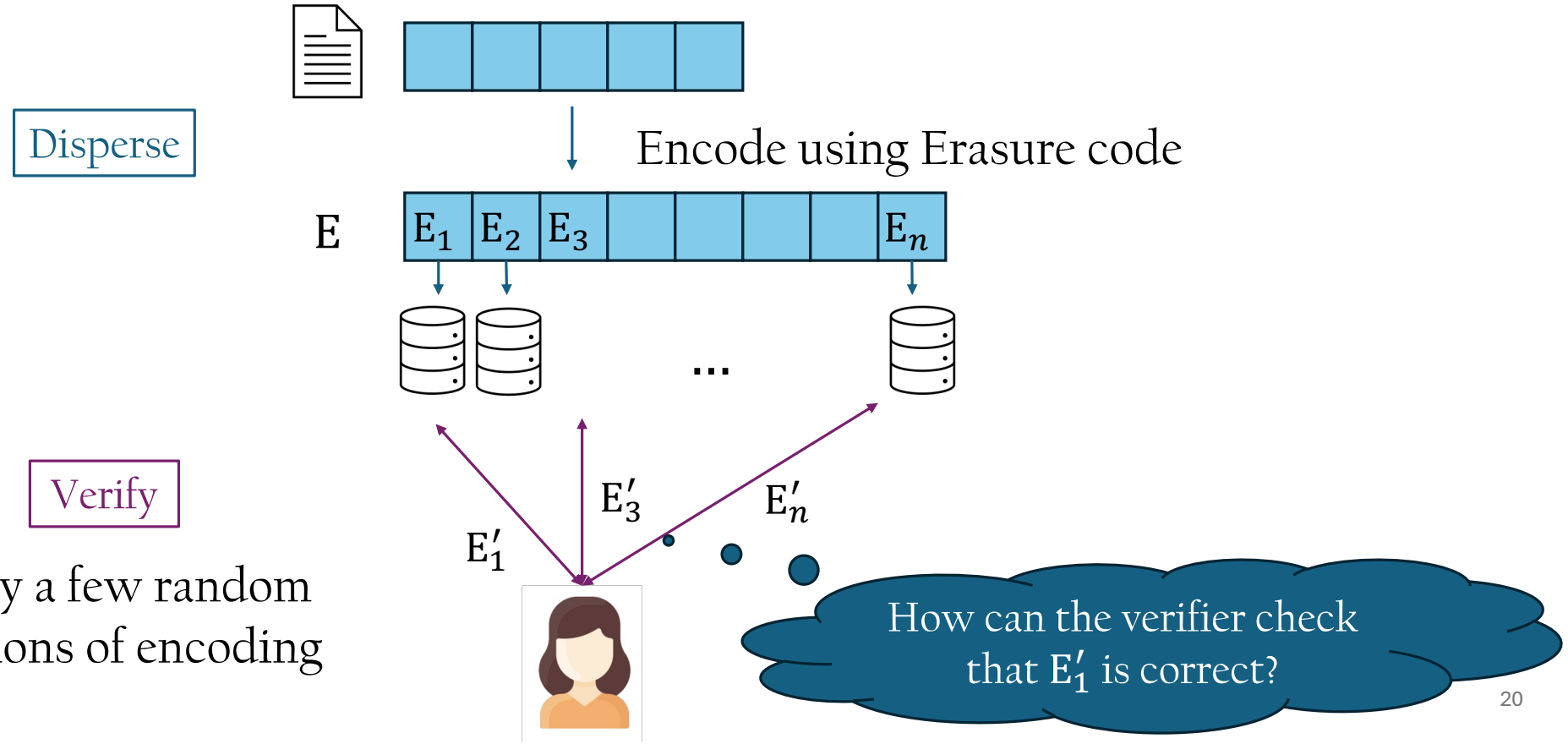
Framework for Data Availability Sampling [HASW23]



Retrieve

1. Query any $(1-\Delta)n$ positions of encoding
2. Run Decode

Framework for Data Availability Sampling [HASW23]



Query a few random positions of encoding

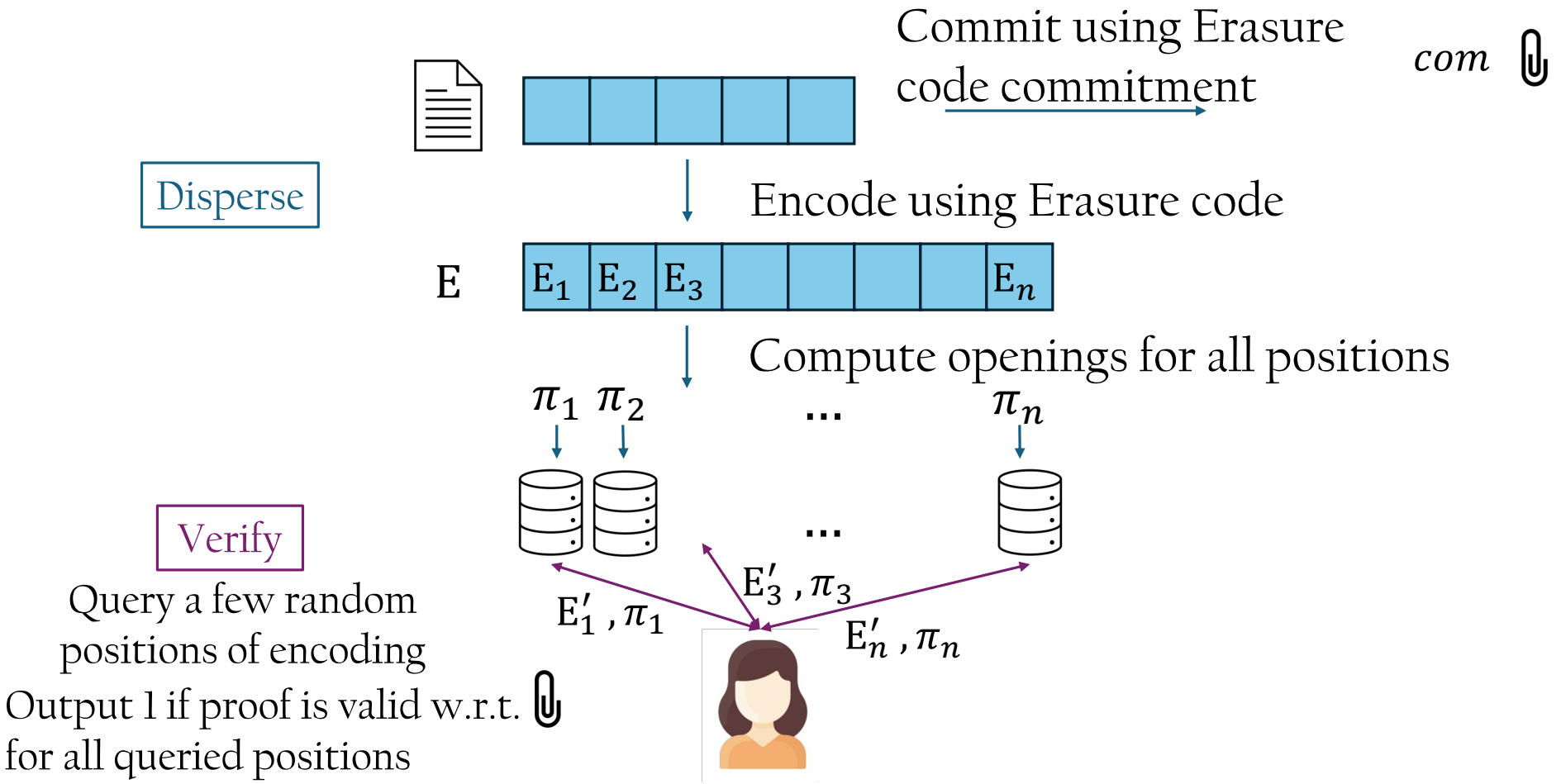
Building Block: Erasure Code Commitment [HASW23]

Can commit to $data$ w.r.t. code C s.t. we can open to any position of the encoding.

- Setup(1^λ) $\rightarrow$ ck
- Commit$(ck, data)$ $\rightarrow$ com
- Open$(ck, data, i)$ $\rightarrow$ π_i
- CVerify$(ck, com, i, \mathrm{E}_i, \pi_i)$ $\rightarrow$ $b \in \{0,1\}$

Properties: Correctness, Position Binding and more... [see paper]

Framework for Data Availability Sampling [HASW23]

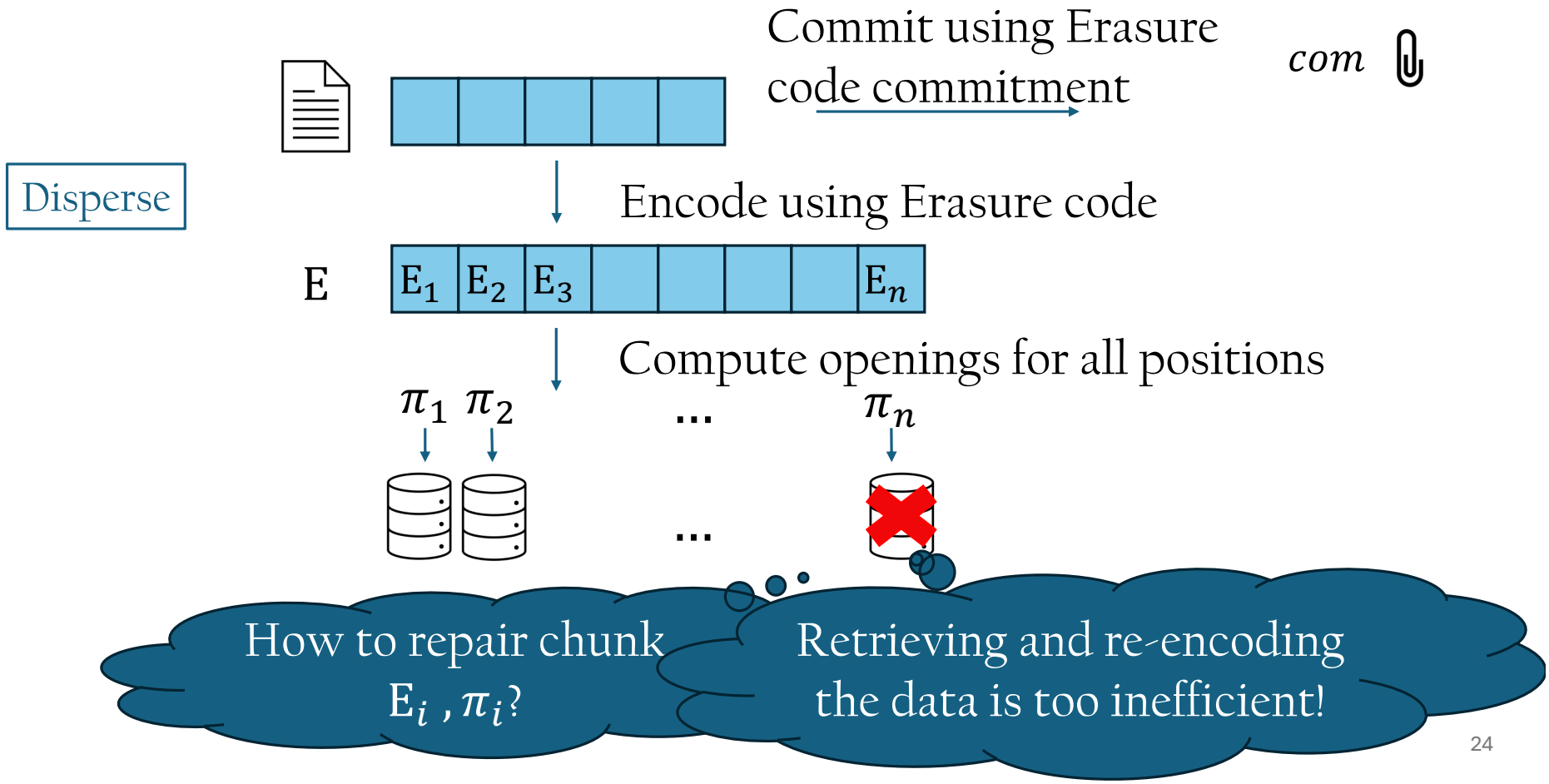


Example: Ethereum Fulu DAS

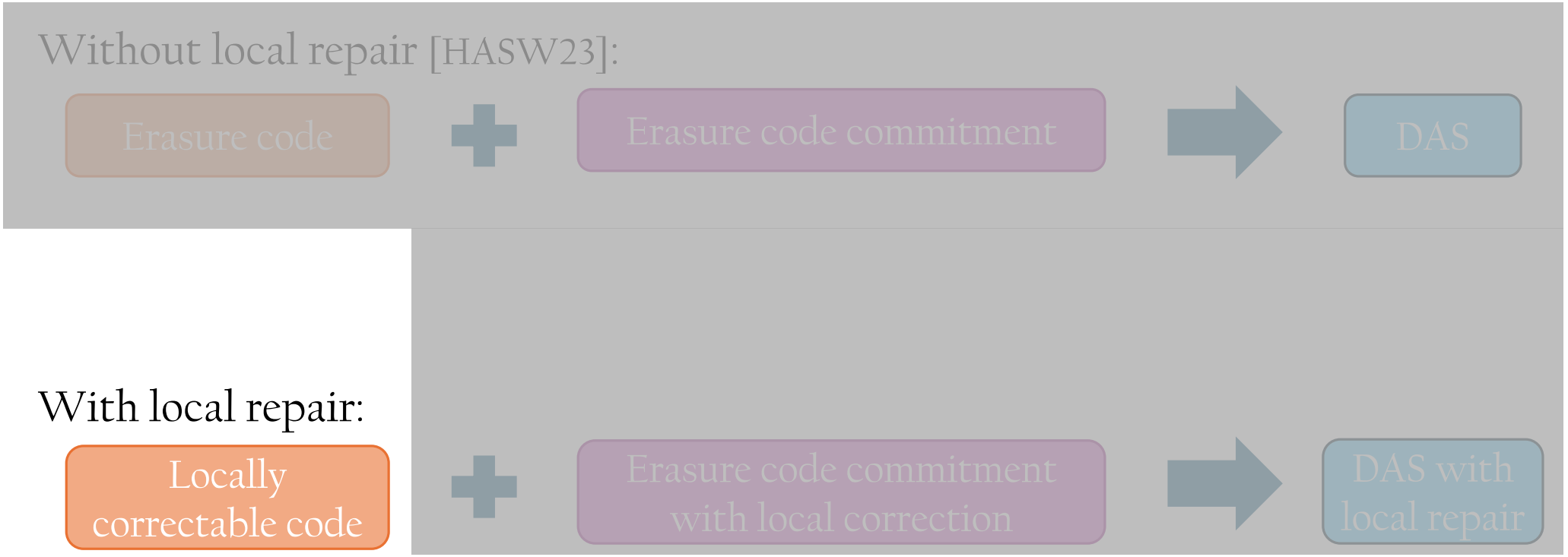
Without local repair [HASW23]:

With local repair:

How to repair lost chunks?



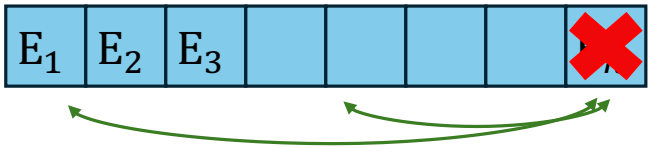
Framework for Data Availability Sampling



Codes with Locality

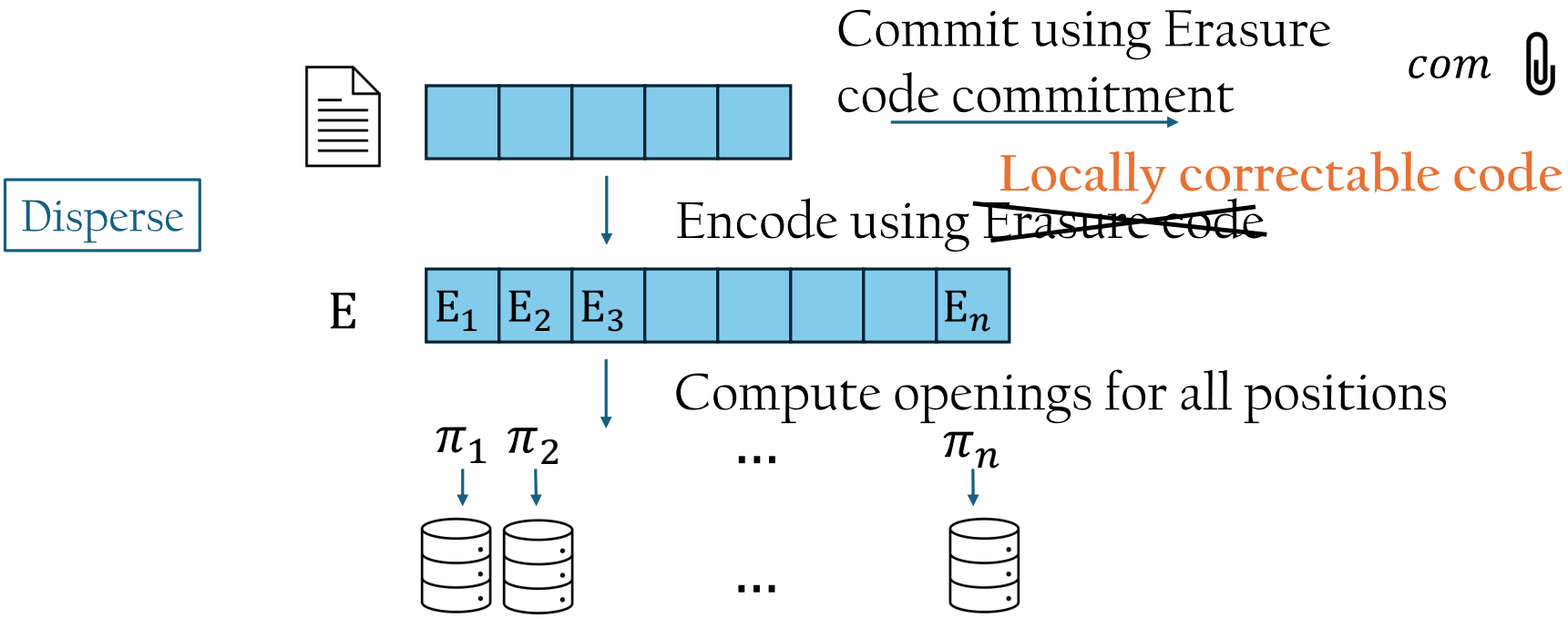
Locally Correctable codes: Can repair any position of the encoding by querying only $r \ll n$ random positions

Encoding E

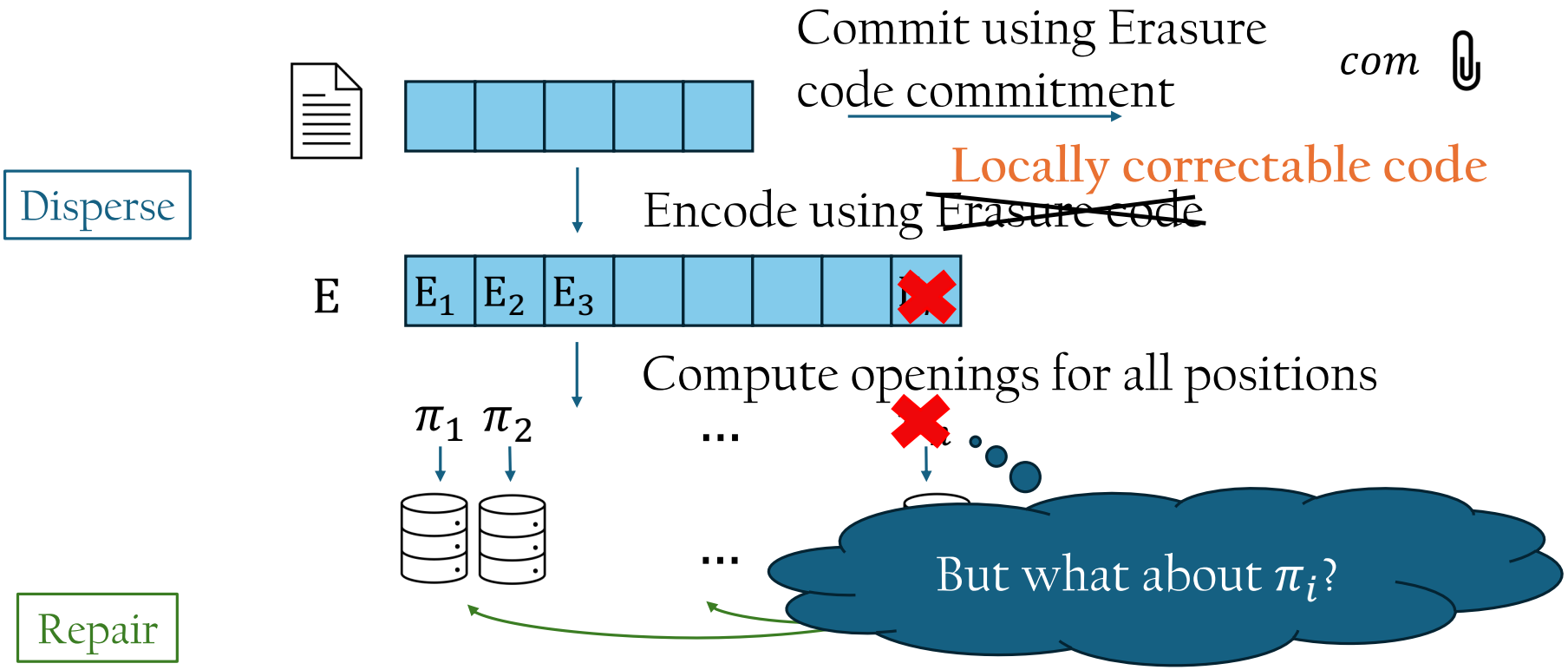


Can compute E_n from E_1 and E_5

New Framework for Data Availability Sampling

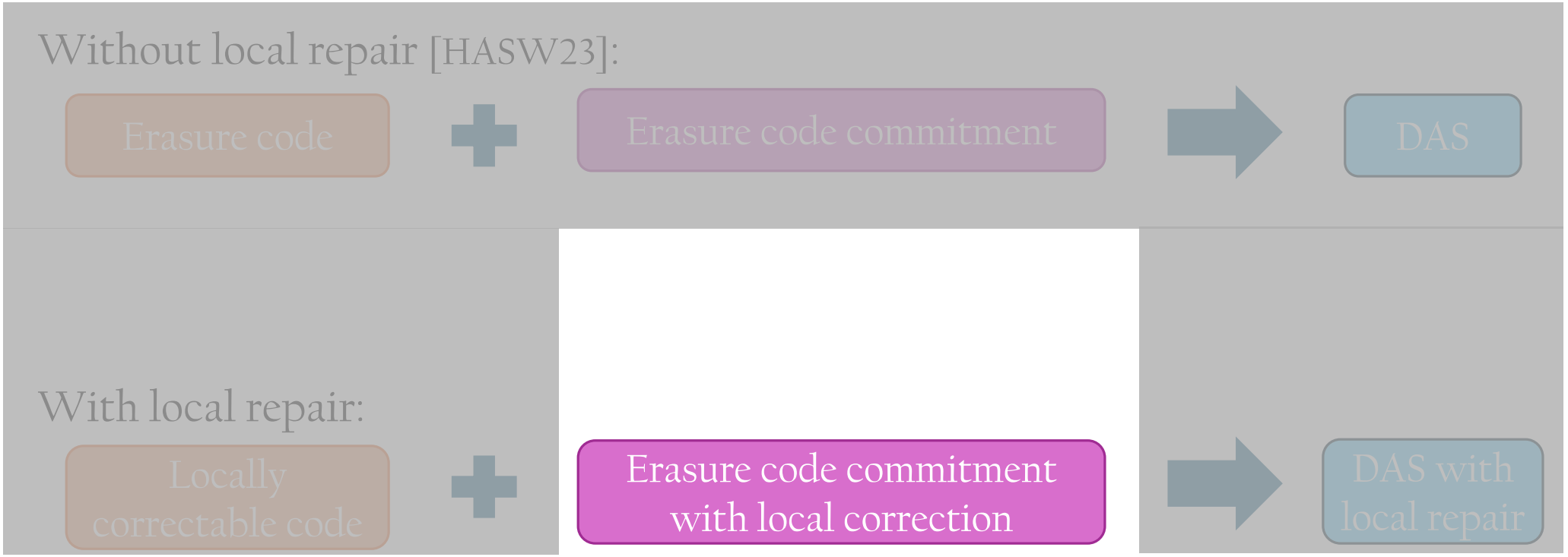


New Framework for Data Availability Sampling



Query r nodes to repair E_n e.g. Nodes 1,5

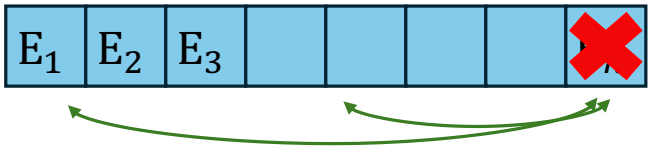
Framework for Data Availability Sampling



New: Erasure Code Commitment with Local correction

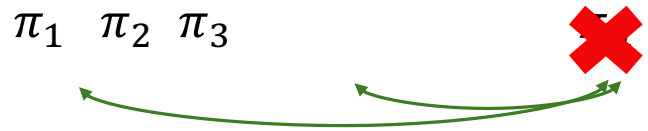
Can repair opening proof for any position of the encoding by querying opening proofs of only $r \ll n$ random positions

Encoding E



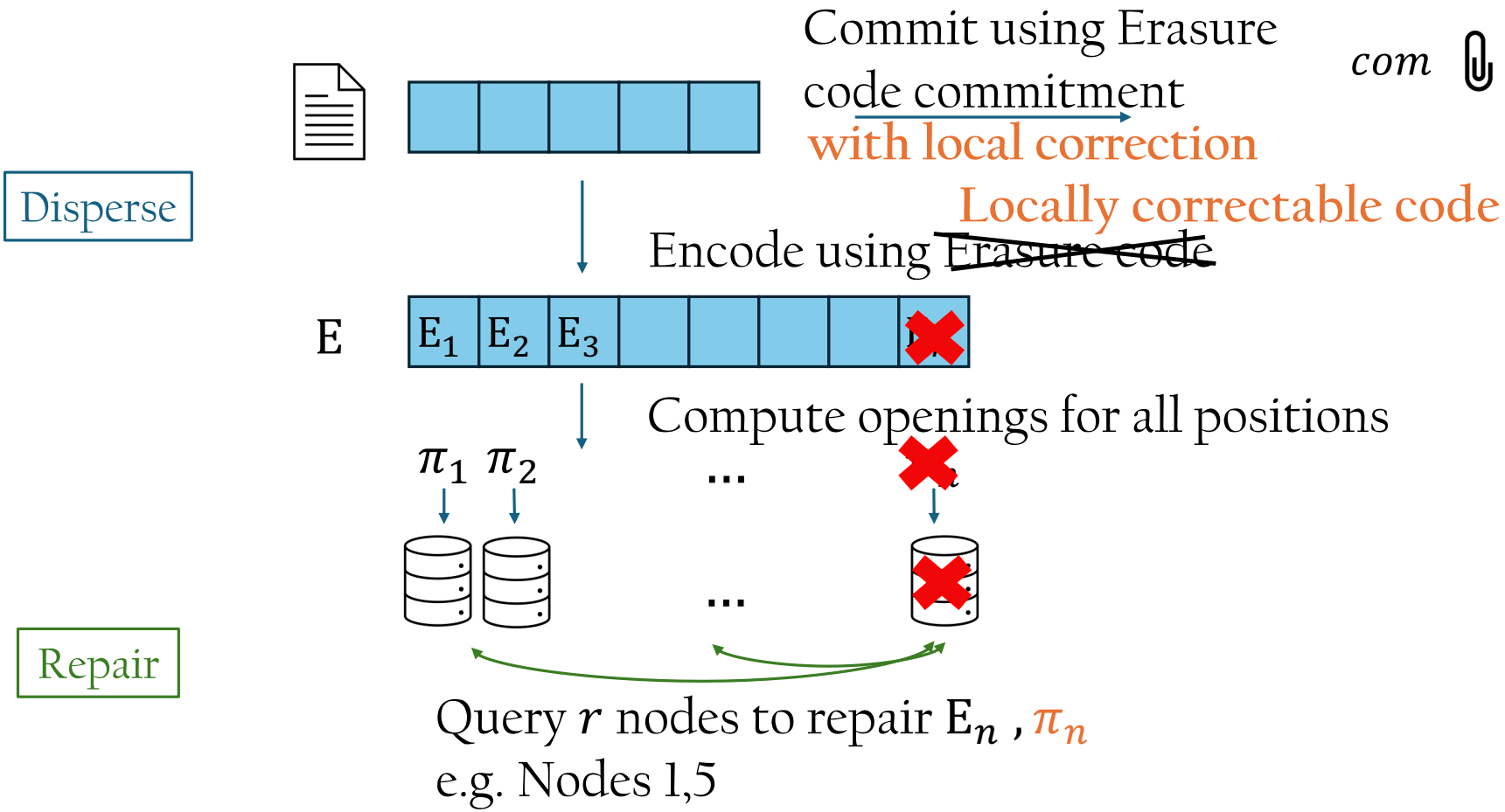
Can compute E_n from E_1 and E_5

Opening Proofs



Can compute π_n from π_1 and π_5

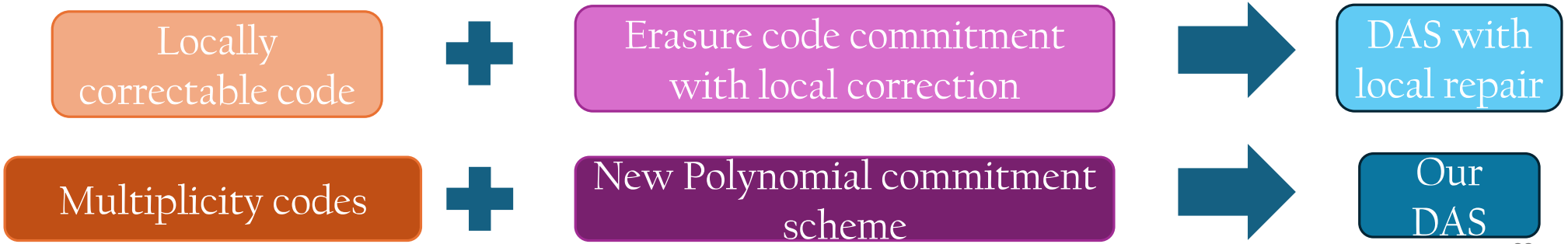
New Framework for Data Availability Sampling



Summary of DAS Framework with Local Repair

Without local repair [HASW23]:

With local repair:



Efficiency of Constructions using DAS Framework

	Locally correctable code		Erasure code commitment with local correction
• Node storage:	One symbol of encoding	+	One opening proof
• Disperser work:	Encoding	+	Opening proofs for all positions
• Repair complexity:	Local correction	+	Local correction for opening proofs
• Sampling bandwidth:	Distance		

Key Challenge:
Co-designing locally correctable codes with good distance and corresponding erasure code commitments...

This work

- New definitions for Data Availability Sampling (DAS), strengthening Hall-Anderson,Simkin,Wagner'23 to formalize:
  - Local Repair
  - Retrieval
- Enhance DAS framework from HASW'23 for building DAS with local repair, from

Codes with locality + Erasure code commitment with locality

- New Construction using the framework with

Multiplicity codes + New Polynomial commitment scheme

Multiplicity Codes [KSY14]

- Generalization of Reed-Solomon codes
- Interpret data as a multivariate polynomial of degree d:

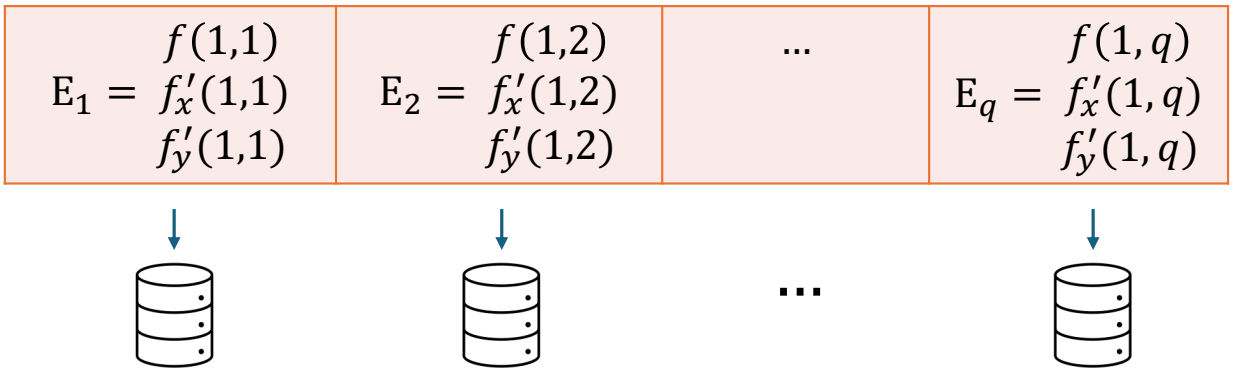
$$data = f(x, y)$$

- Encoding contains:
  - Evaluation of f at all values of $x, y \in S$
  - Evaluation of $f'_x = \frac{\partial f}{\partial x}$ at all values of $x, y \in S$
  - Evaluation of $f'_y = \frac{\partial f}{\partial y}$ at all values of $x, y \in S$

Multiplicity Codes [KSY14]

$$\mathrm{E}_1 = \begin{matrix} f(1,1) \\ f'_x(1,1) \\ f'_y(1,1) \end{matrix}$$

Multiplicity Codes [KSY14]

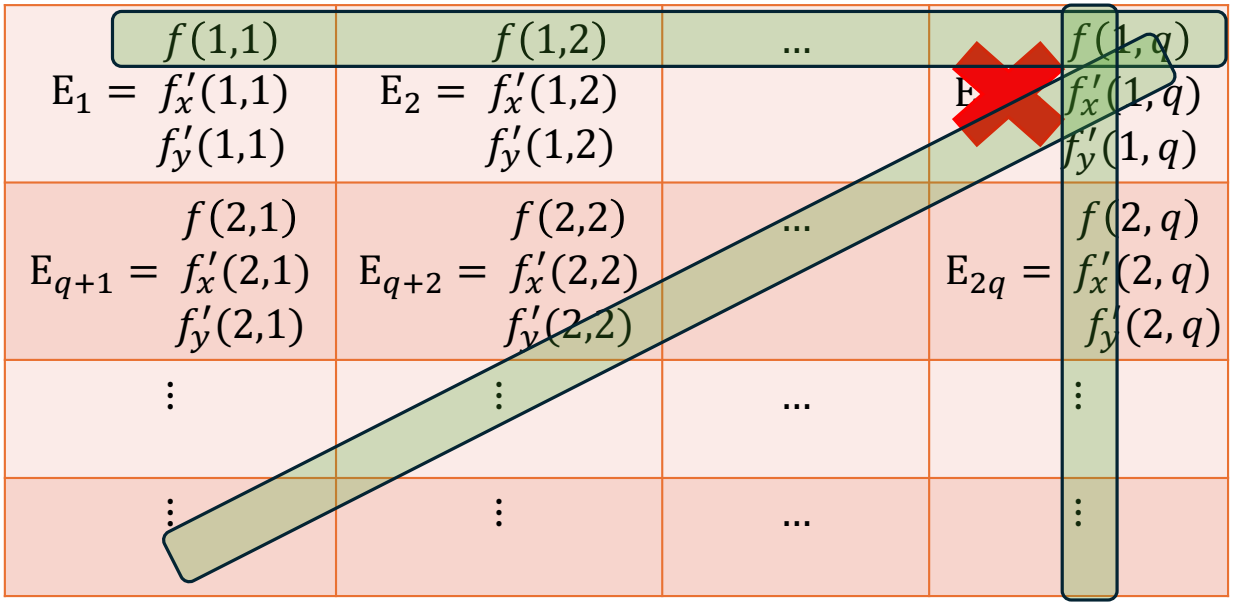


Multiplicity Codes [KSY14]

We give the first $O(k \log^2 k)$ algorithm for systematic encoding, that enables efficient retrieval! [See paper]

Derivatives provide redundancy
⇒ Good distance

Degree of f along any line is d ⇒
Good local correction



$f(1,y)$ has degree d :
Can repair by querying $d+1$ symbols on this line

This work

- New definitions for Data Availability Sampling (DAS), strengthening Hall-Anderson,Simkin,Wagner'23 to formalize:
  - Local Repair
  - Retrieval
- Enhance DAS framework from HASW'23 for building DAS with local repair, from

Codes with locality  Erasure code commitment with locality

- New Construction using the framework with

Multiplicity codes  New Polynomial commitment scheme

The need for a new commitment scheme

Need a commitment scheme with the following properties:

- Erasure code commitment for Multiplicity codes i.e.
  - Commitment for multivariate polynomials
  - Opening proofs for derivatives
- Local correction for opening proofs
- Efficient Batch opening for proofs at all positions

Crucial to instantiate framework with multiplicity codes

Crucial for local repair!

Crucial for reducing disperser work!

New Multivariate Polynomial Commitment

First scheme which supports:

Generalize PST Commitment

- Opening proofs for evaluation of multivariate polynomial **and its derivatives**

  Proof size: 3 $\mathbb{G}$ Verify time: 4 pairings for bivariates

- Efficient Batch opening for proofs at all positions
- Local correction for opening proofs

Generalize Khovratovich-Feist to multivariates

https://eprint.iacr.org/2025/1414

Conclusion

- Local Repair in Data Availability Sampling is an important problem
- New DAS with local repair using

Multiplicity Codes + New Multivariate Polynomial commitment scheme

Open Questions

Designing commitment schemes with local correction for other locally correctable codes:

- Expander codes
- Folded Reed-Solomon codes

Designing new erasure codes with local correction