# Redactable Distributed Ledger Technology with Hyperledger Fabric

**Privacy Symposium 2023**

Industry Session: Blockchain, Privacy and Use Cases

Venice, Italy, April 18, 2023

Rick Kuhn

US National Institute of Standards and Technology

kuhn@nist.gov

# Key Points – why listen to this talk?

- Blockchain has valuable properties, but conflicts with privacy and exception management – "immutable" - deletion impossible

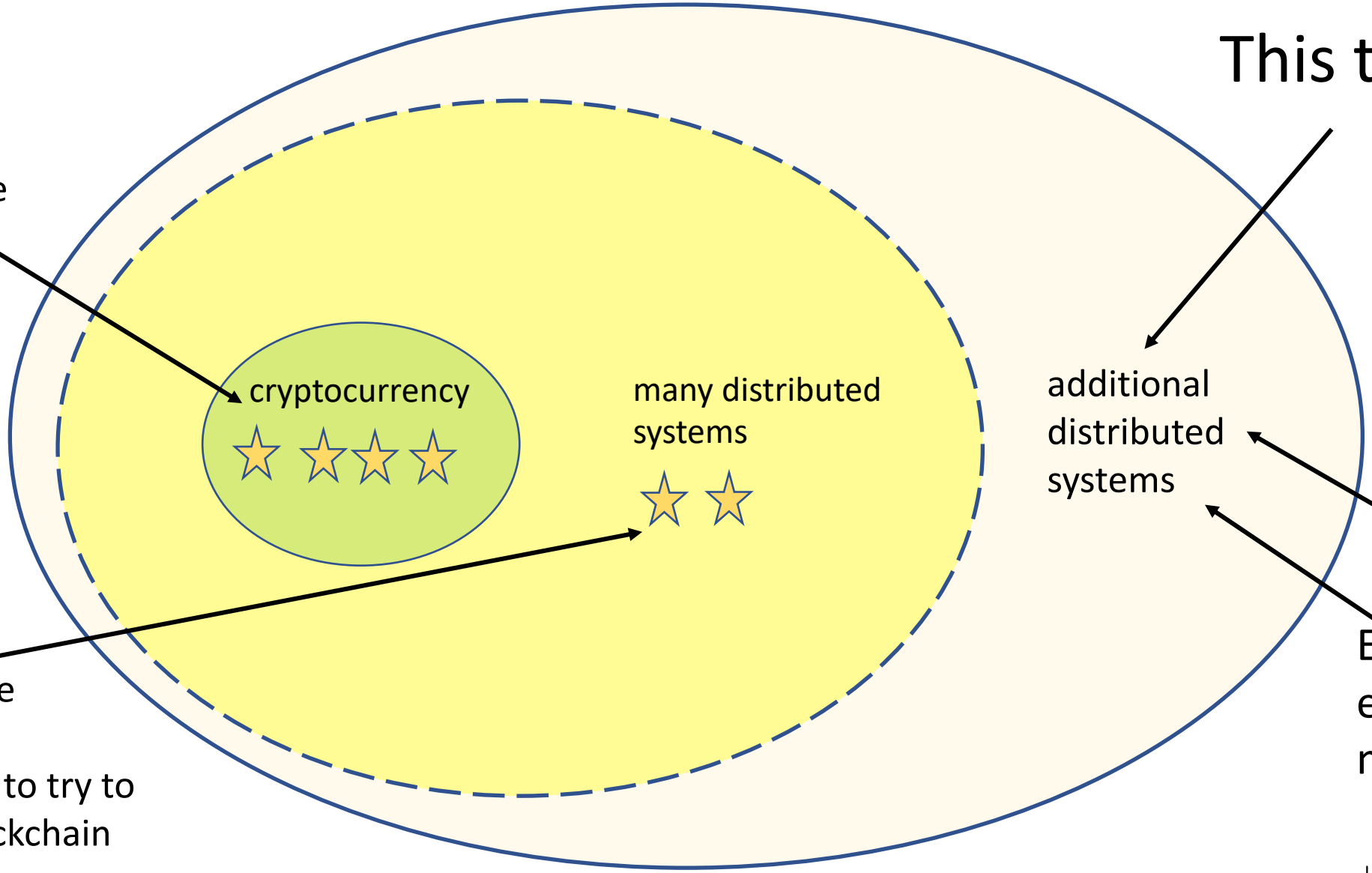
  ➡ Sometimes we don't need blockchain,
  just some blockchain features

- Data structure called *blockmatrix* provides distributed trust, integrity protection of blockchain, but allows controlled edits for privacy or corrections

- Drop-in compatibility for Hyperledger Fabric applications

  ➡ Released and available

# Market, range of applications for DLT




Much research to try to get around blockchain properties

# Why use redactable distributed ledger for error and exception management?

- Using blockchain in logistics is difficult
- "This Could Be the End of Enterprise Blockchain" (shutdown of TradeLens by IBM)

  - Motley Fool (popular investment news), Jan 10, 2023

- "Digital forwarding - Respondents say the biggest advantage is improved tracking and visibility; the biggest disadvantage is error/exception management."
  - survey of 750 global logistics executives, Supply Chain Management Review, Feb 13, 2023

# Why use redactable DLT for privacy?

- Permanence/immutability conflicts with ‘right to erasure’ privacy regulations
- Privacy rules such as European Union General Data Protection Regulation (GDPR) require that all information related to a particular person can be deleted at that person's request
  - *personal* data, defined as "any information concerning an identified or identifiable natural person" - data for which blockchains are designed
  - "Personal data which have undergone pseudonymisation, which could be attributed to a natural person by the use of additional information should be considered to be information on an identifiable natural person."
- US states adopting similar privacy rules, including California and Virginia
- Law enforcement also requires expungement of data in some cases

# What's been tried to solve blockchain/privacy conflict?

- Don't put personal data on blockchain
  - Pseudo-anonymized data are still considered personal
  - Even if not directly tied to a person – dynamic IP address can be considered personal if it can be indirectly tied
  - Financial transactions are obviously personal data
- Encrypt data and destroy key to delete
  - Data must be secure for decades (DES replaced in only 17 years)
  - Advancements in cryptography usually compromise old crypto – e.g., quantum computing puts current public key systems at risk

# Redactable/editable blockchain, DLT

- Chameleon hash function - most common approach to providing editability
  - Generate a collision for the hash, given trapdoor or key, changing data but hash not disturbed
  - Standard blockchain for integrity protection
  - Requires specialized chameleon hash function
- Our approach, data block matrix
  - Dual hash list for integrity protection
  - Use standard hash function (SHA 256)
- Either may be best, depending on application requirements
  - *Tradeoffs like any other engineering problem*

# Many blockchain applications don't need blockchain, just some blockchain features

## Can we try something else?

**Datablock matrix** – uses two hash values per block instead of a linked chain

- Java or Go example code available as open source
- Incorporated into Next Gen Access Control – practical demo
- NOT to replace blockchain, to provide alternative tools for distributed system design
- Hyperledger Fabric component available

# Datablock matrix data structure

- A data structure that provides integrity assurance using hash-linked records while also allowing the deletion of records
- Stores hashes of each row and column
- => each block within the matrix is protected by two hashes
- Suggested use for private/permissioned distributed ledger systems

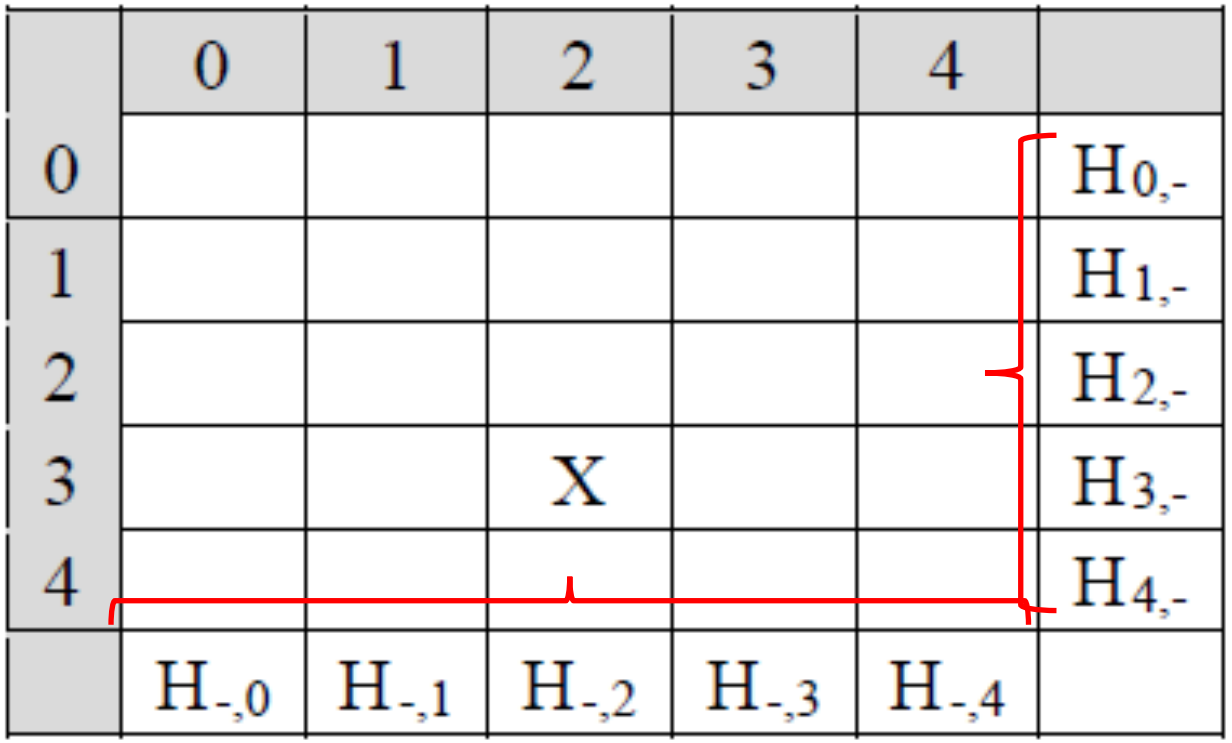



Figure 1. Block matrix

# How does this work?

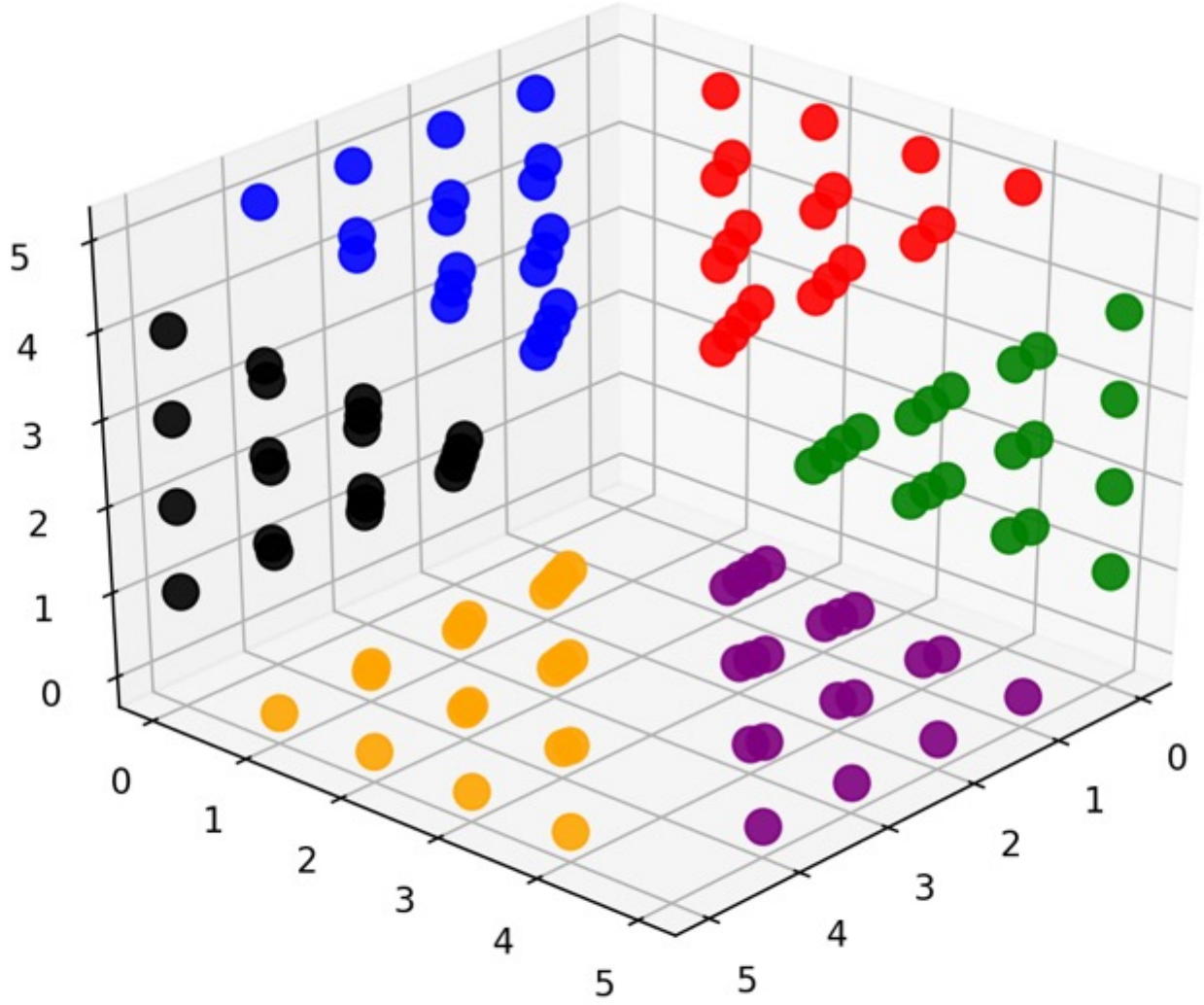
- Suppose we want to delete block 12

- disrupts the hash values of $H_{3,-}$ for row 3 and $H_{-,2}$ and column 2
- blocks of row 3 are included in the hashes for columns 0, 1, 3, and 4
- blocks of column 2 are included in the hashes for rows 0, 1, 2, and 4

| | 0 | 1 | 2 | 3 | 4 | |
|---|---|---|---|---|---|---|
| 0 | • | 1 | 3 | 7 | 13 | $H_{0,-}$ |
| 1 | 2 | • | 5 | 9 | 15 | $H_{1,-}$ |
| 2 | 4 | 6 | • | 11 | 17 | $H_{2,-}$ |
| 3 | 8 | 10 | 12 | • | 19 | $H_{3,-}$ |
| 4 | 14 | 16 | 18 | 20 | • | $H_{4,-}$ |
| | $H_{-,0}$ | $H_{-,1}$ | $H_{-,2}$ | $H_{-,3}$ | $H_{-,4}$ | etc. |

# Structure can be extended to multiple dimensions

- Block dispersal for 3 dimensions
- Location in sectors 0..5 according to *b* mod 6 for block *b*



# Why use this data structure?

Again, many blockchain applications don't need blockchain, just some features

## Enlarge the market for blockchain

- Solve the conflict between blockchain and privacy regulations
- Allow for exception management

## Replace network communication with local data

- You can obviously do this with conventional database functions, but
- New data structure adds integrity checks as in blockchain

## Easy-to-use component for distributed database design

# NIST blockchain decision flowchart

Do you need a shared, consistent data store?

NO → Distributed ledgers provide a historically consistent data store. If you don't need that, you don't need a distributed ledger

**CONSIDER:** Email / Spreadsheets

YES

Does more than one entity need to contribute data?

NO → Your data comes from a single entity. Distributed ledgers are typically used when data comes from multiple entities.

**CONSIDER:** Database **CAVEAT:** Auditing Use Cases

YES **AUDITING**

Data records, once written, are never updated or deleted?

NO

YES

Sensitive identifiers WILL NOT be written to the data store?

NO → You should not write sensitive information to a blockchain that requires medium to long term confidentiality, such as PII, even if it is encrypted

**CONSIDER:** Encrypted Database **OR blockmatrix**

YES

Are the entities with write access having a hard time deciding who should be in control of the data store?

NO → If there are no trust or control issues over who runs the data store, traditional database solutions should suffice

**CONSIDER:** Managed Database

YES

Do you want a tamperproof log of all writes to the data store?

NO → If you don't need to audit what happened and when it happened, you don't need a distributed ledger

**CONSIDER:** Database

YES

You may have a useful blockchain use case

Uses handled by blockmatrix that cannot be done in blockchain

Are the entities with write access having a hard time deciding who should be in control of the data store?

NO

YES

Do you want a tamperproof log of all writes to the data store?

NO

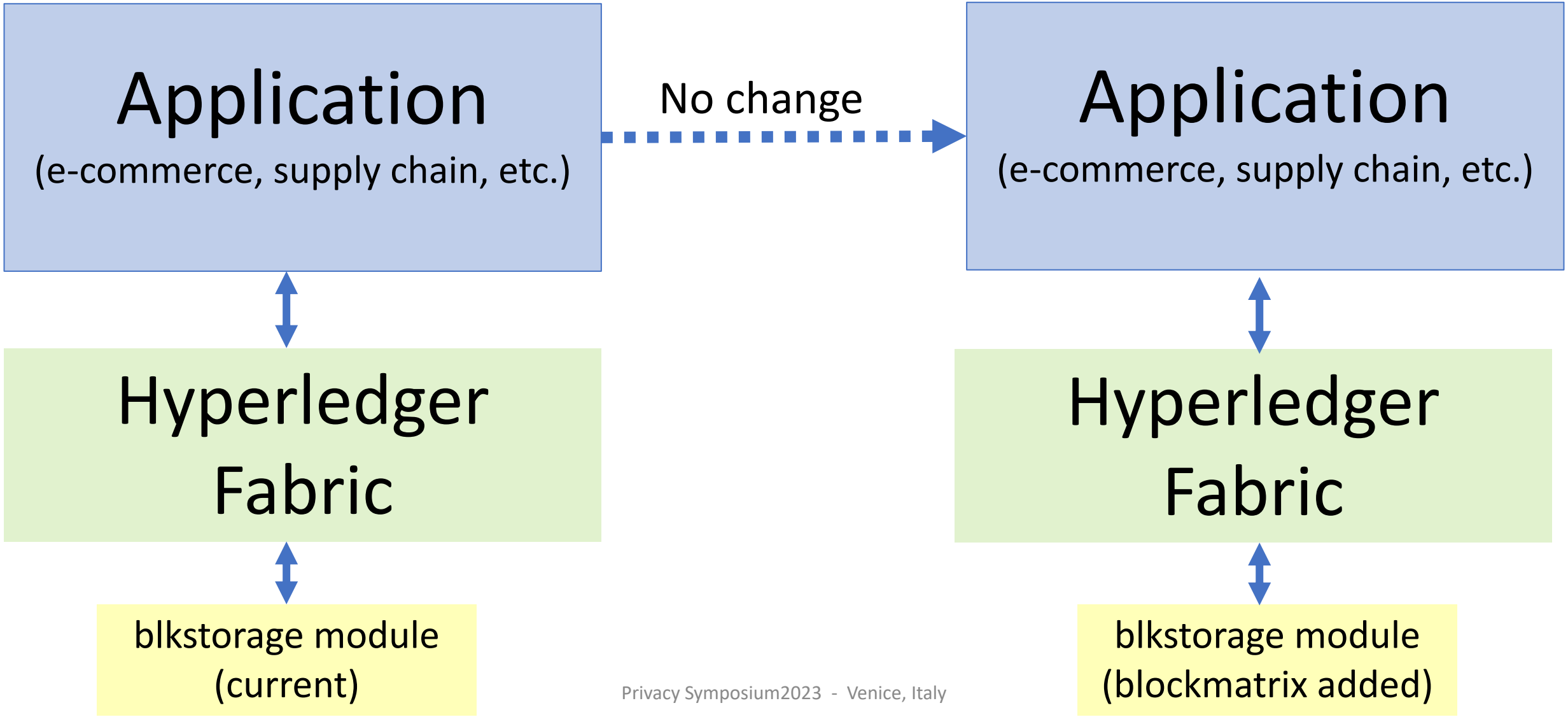
YES

You may have a useful data block matrix use case

# Hyperledger blockmatrix implementation

- Designed to use existing API as closely as possible
  – add blocks in same manner as adding to blockchain
- Blockmatrix is configurable by channel (private subnet)
- Configure to use conventional blockchain or blockmatrix
  - If a deployment uses two channels, one can be a blockchain and the other can be a blockmatrix
- RED Ledger = Redactable Enhanced Distributed Ledger
- https://csrc.nist.gov/projects/redactable-distributed-ledger

# Compatible with current Hyperledger applications



# Applications in clinical trials

- “An Infrastructure for Secure Sharing of Clinical Data” – to be presented at Healthcare Information and Management Systems Society, HIMSS 23 https://www.himss.org/global-conference/session-infrastructure-secure-sharing-clinical-data-6

- “Data Block Matrix and Hyperledger Implementation: Extending Distributed Ledger Technology for Privacy Requirements”, J. Roberts, J. DeFranco, R. Kuhn *ACM Distributed Ledger Technologies: Research and Practice*. https://dl.acm.org/doi/10.1145/3585539

# More information:


- Kuhn, R., Yaga, D. and Voas, J., 2019. Rethinking Distributed Ledger Technology. *Computer*, *52*(2), pp.68-72.
- Kuhn, D. R. (2018). A Data Structure for Integrity Protection with Erasure Capability. https://csrc.nist.gov/publications/detail/white-paper/2022/05/20/data-structure-for-integrity-protection-with-erasure-capability/final


## Project sites with links to source code and publications

- https://csrc.nist.gov/Projects/enhanced-distributed-ledger-technology
- https://csrc.nist.gov/projects/redactable-distributed-ledger

## Acknowledgements


- Josh Roberts, Jeff Voas, Dylan Yaga, Sylvain Chantreau, NIST
- Temur Saidkhodjaev, University of Maryland College Park
- Arsen Klyuev, Johns Hopkins University
- Gokhan Kocak, Asena, Inc.