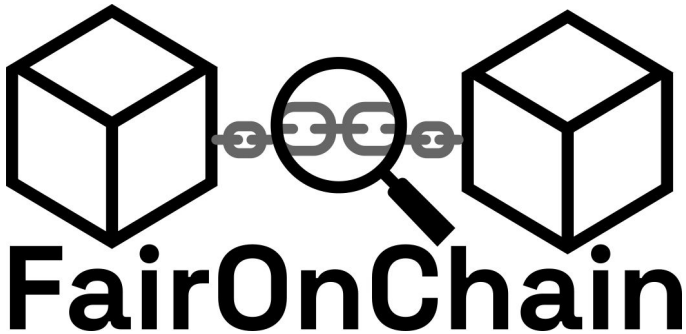


SoK: Unified Blockchain Data Structure

Natkamon Tovanich, Madalina I. Sas, Christophe Lebrun, Munthitra Thadthapong, William J. Knottenbelt, Arnaud Gaudinat, and Marco Mattavelli

EPFL

h e g
Haute école de gestion Genève

ÉCOLE POLYTECHNIQUE

INSTITUT POLYTECHNIQUE DE PARIS

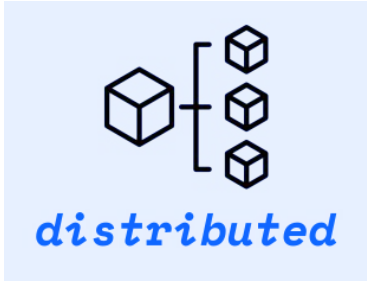
IMPERIAL

Unified Data Structure

Blockchain data is inherently accessible, but difficult to analyze due to the **lack of unified structure and standardisation**.

Our project goal is to build an open infrastructure which is **distributed, extensible and modular** based on the **FAIR** principles of open science.

To be able to **reproduce** research pipelines and provide **standardised** APIs, we require a **unified data format**.



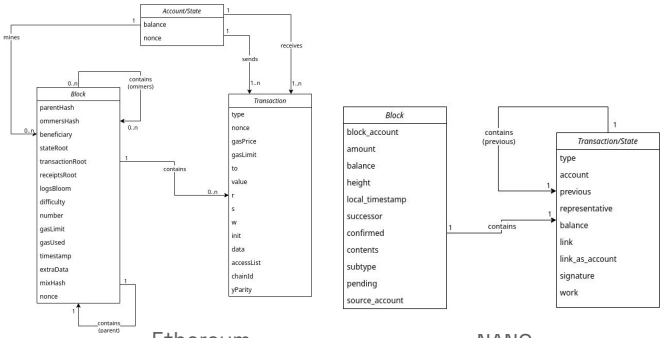
Challenges to Blockchain Data Unification

- Differences in **operationalisation & architecture**
  - Account vs UTxO vs other architectures (NANO, Cardano, etc)
- **Tokens**: Native assets vs Smart Contract tokens
- **Interoperability**:
  - Scaling solutions: Layer-2, Sidechains
  - Bridges, Exchanges
- **Decentralised tools** (dApps, DeFi, DAOs)

Theoretical Models

A Unified Data Model for Blockchains

(Meyer & dos Santos Mello, 2022)



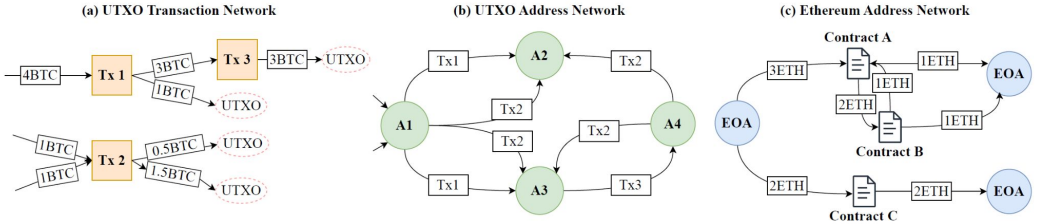
Ethereum NANO

Ethereum	Nano	Comments
Block/parentHash	Transaction/previous	Hash of the previous block in the chain
Account/balance	Transaction/balance	Current balance of the account
Block/timestamp	Block/local_timestamp	Timestamp of when the block/transaction was mined
Transaction/from	Transaction/source_account	Account that originated the transaction (sender)
Transaction/r,s	Transaction/signature	Cryptographic signature of the creator of the transaction (sender)
Block/nonce	Transaction/work	Proof of work

Mapping Similar Attributes in Ethereum and NANO

Blockchain Networks

(Akçora et al., 2022)



Chimeric Ledgers

(Zahnentferner, 2018)

```
Ledger := List[Transaction]
UTxOTx := (inputs: Set[Input], outputs:
Set[Output],forge: Value, fee: Value)
Output := (address: Address, value: Value)
Input := (tx : Id, index : Int)
```

```
AccTx := (sender: Option[Address],
receiver: Option[Address], value: Value,
forge: Value, fee: Value, nonce: Int)
```

```
HybridTx := (inputs: Map[Address, Value], outputs: Map[Address, Value],
forge: Value, fee: Value, nonce: Int)
DepTx := (inputs: Set[Input], depositor: Option[Address], forge: Value,
fee: Value)
WithTx := (withdrawer: Address, outputs: List[Output], forge: Value,
fee: Value, nonce: Int)
```

Open Source and Commercial Tools

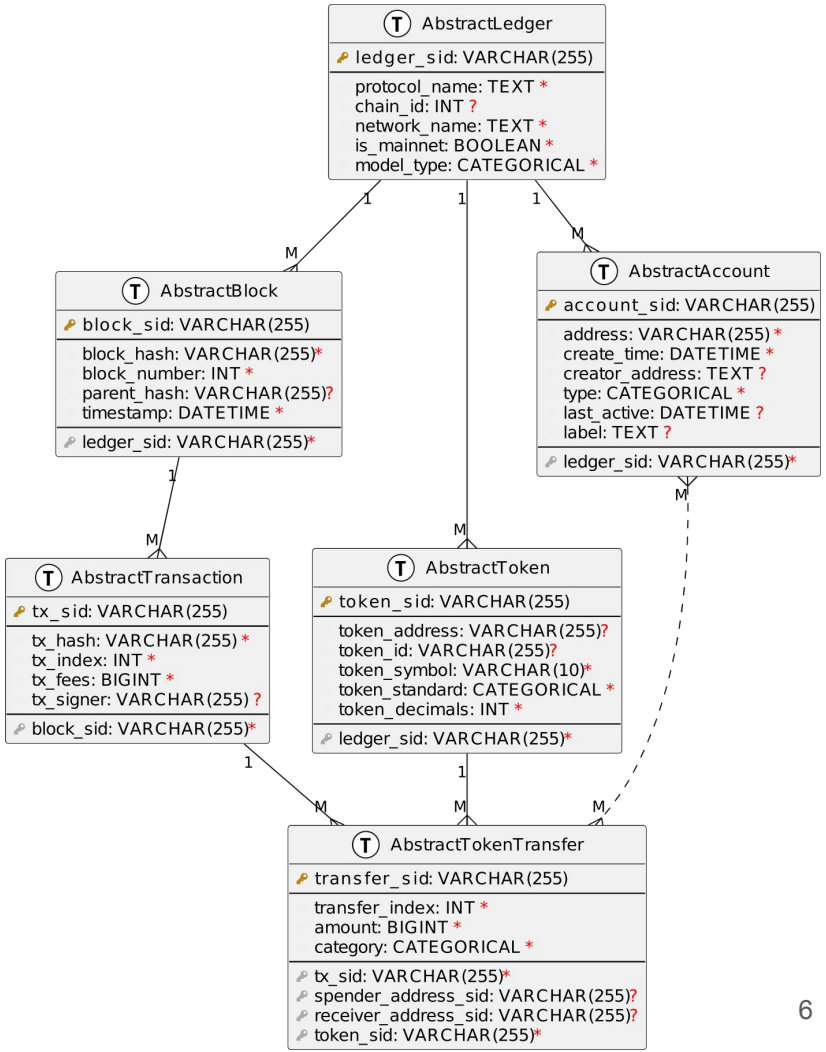
		Raw Data			Decoded Data						Off-Chain Data			
Tool	# Chains	UTxO	EVM	Layer 2	Event Logs	ERC-20 Transfers	NFT Trades	DEX Trades	Bridges	Protocol-specific	Token Price	Entity Tagging	Query Endpoint	Backend DB
Node-as-a-Service														
Infura	19		✓	✓									JSON-RPC	Unknown
QuickNode	60	✓	✓	✓		✓	✓	C			C		JSON-RPC	Unknown
Alchemy	38		✓	✓		C	C						JSON-RPC	Unknown
Moralis	18		✓	✓	C	C	C	C		C	C	✓	JSON-RPC	Unknown
Data Processing														
Blockchain ETL	18	✓	✓	✓		✓	✓						SQL	Google BigQuery
Cryo	6		✓	✓		C	C						Command Line, Python library	Raw table files
Data Provider														
Blockchair	17	✓	✓			✓							JSON-RPC	Unknown
BitQuery	10		✓	✓		C	C	C					GraphQL	Unknown
GoldRush	75	✓	✓	✓	C	C	C				C		JSON-RPC	Unknown
Transpose	12	✓	✓	✓		✓	✓	✓	✓	✓	✓		JSON-RPC, SQL	Unknown
The Graph (and Messari Subgraphs)	71	✓	✓	✓		C	C	C	C	C			GraphQL	Firehose
Data Query & Analytics Tool														
Blockscout	47		✓	✓		✓	✓					✓	JSON-RPC, GraphQL	PostgreSQL
Dune Analytics	47	✓	✓	✓	✓	C	C	C	C	✓	C	✓	SQL, JSON-RPC	DuneSQL
Flipside Crypto	30	✓	✓	✓	✓	✓	✓	C	C	C	C	✓	SQL, JSON-RPC	dbt, Snowflake
Footprint Analytics	31		✓	✓		✓	C	C	C	✓	✓	✓	SQL, JSON-RPC	SQL
Allium	62	✓	✓	✓	✓	✓	C	C	C	C	✓		SQL, JSON-RPC	PostgreSQL, Snowflake, DataBricks, Bigquery, S3, GCS

Limitations

- Fragmented solutions
- No interoperability
- Not designed for research reproducibility
- Commercial cost barriers

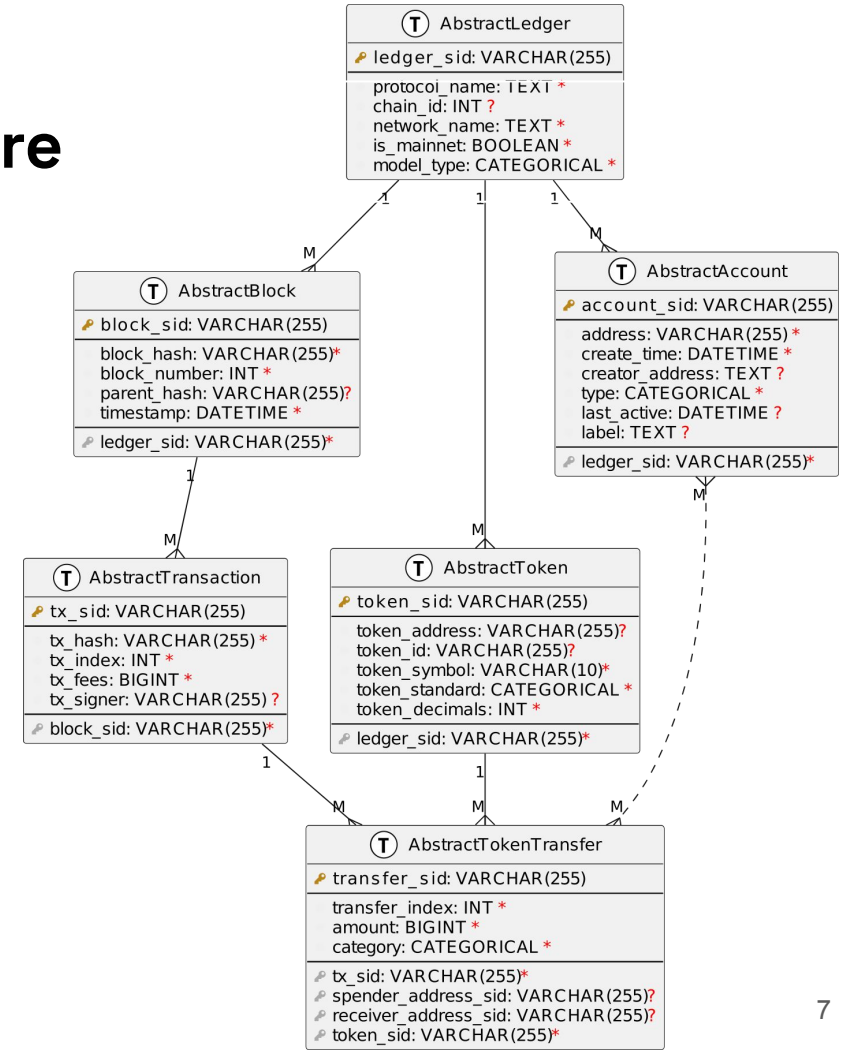
Our Solutions

- Core idea: **token transfers** as atomic unit
- Use abstraction to focus on transfers
- Works across UTxO & account-based systems
- Flatten UTxO to store inputs & outputs as deposits & withdrawals



Abstract Unified Data Structure

- ***Ledger***: general information about protocol, chain ID, network name
- ***Block***: hash, number/hight, parent hash, timestamp
- ***Token***: general info on native tokens or uniquely identify NFTs
- ***Transaction***: general info such as hash, fees, index in block, signer(s)
- ***TokenTransfer***: amount, index in Tx or SC, category, spender, receiver



Abstract Token Transfers & Transactions

`AbstractTokenTransfer`

- Inspired by Chimeric Ledgers:
  UTxO inputs → *Deposits*
  UTxO outputs → *Withdrawals*
- Flatten multi-token Tx (e.g. ERC 1155) & transfers in SC
- Use spender/receiver address rather than Tx hash
- Capture the net effect of address balances across architectures

`AbstractTransaction`

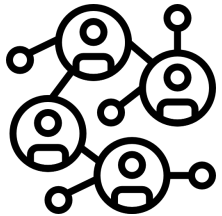
- Enrich token transfers with contextual tables for ledgers, tokens, blocks, Tx
- Surrogate keys to identify data
- Store fields compatible with multiple chain architectures: hash, index, fees, signer

Applications

Unified data can facilitate the creation of reproducible research pipelines across chains for many use-cases, as well as easier cross-chain analytics.

Transaction & network statistics

Account Forensics & Anti Money Laundering (AML)



Token transfer networks

Discussion

Trade-offs

- Data volume
- Granularity vs simplicity
- Limited applicability for fine-grained provenance data (e.g. traces)
- Anonymity & privacy concerns
- Protocol evolution, forks, backwards compatibility
- Address extraction in UTxO requires parsing whole chain history

Limitations

- No evaluation/benchmarking metrics applicable to all reviewed tools
- Lack of full access to all reviewed commercial platforms

Future work

- Implement unification pipeline (still in development)
- Develop evaluation metrics

faironchain.org

FairOnChain is an ambitious European collaboration aiming to develop a publicly accessible infrastructure that enables easy access and searchability of **blockchain data** in accordance with the **FAIR** principles of open science. This infrastructure aims to promote complete transparency and reproducibility of scientific analysis results in the blockchain field, thereby facilitating the growth and collaboration of new and existing applications.

This project is based upon participation in the CHIST-ERA 2022 call for Open & Re-usable Research Data and Software (ORD).