Dynamic Data Quality for Static Blockchains

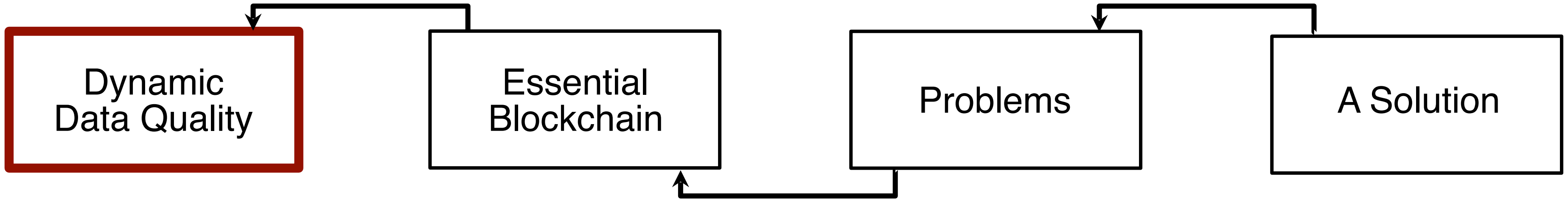
Alan G. Labouseur, Ph.D.
Alan.Labouseur@Marist.edu

Carolyn C. Matheus, Ph.D.
Carolyn.Matheus@Marist.edu

BlockDM @ ICDE 2019

Blockchain's popularity has changed the way people think about data access, storage, and retrieval. Because of this, many classic data management challenges are imbued with renewed significance. One such challenge is the issue of Dynamic Data Quality.

This is a story about the friction between static blockchains and Dynamic Data Quality, and how to fix it.



Dynamic Data Quality

Essential Blockchain

Problems

A Solution

Daily Deluge of Data

We are awash in data deluge.

- It's constantly growing.
- It's constantly changing.
- It's constantly evolving.

It's complex.

- structured
- unstructured
- semi-structured

Piling up data is easy.

- Gaining insight from the data pile is hard.

Big Data Characteristics[1]

- volume
- velocity
- variety
- … and don't forget **veracity**

Can we believe it?

[1] Shankaranarayanan, G. & Blake,R. (2017). From content to context: The evolution and growth of data quality research. *Journal of Data and Information Quality* 8(2), 9:1–9:28.

Data Quality

Errors associated with data …

- collection
- storage
- retrieval
- representation

… are **long-standing** problems with serious implications. If your is low quality, then what good is it?

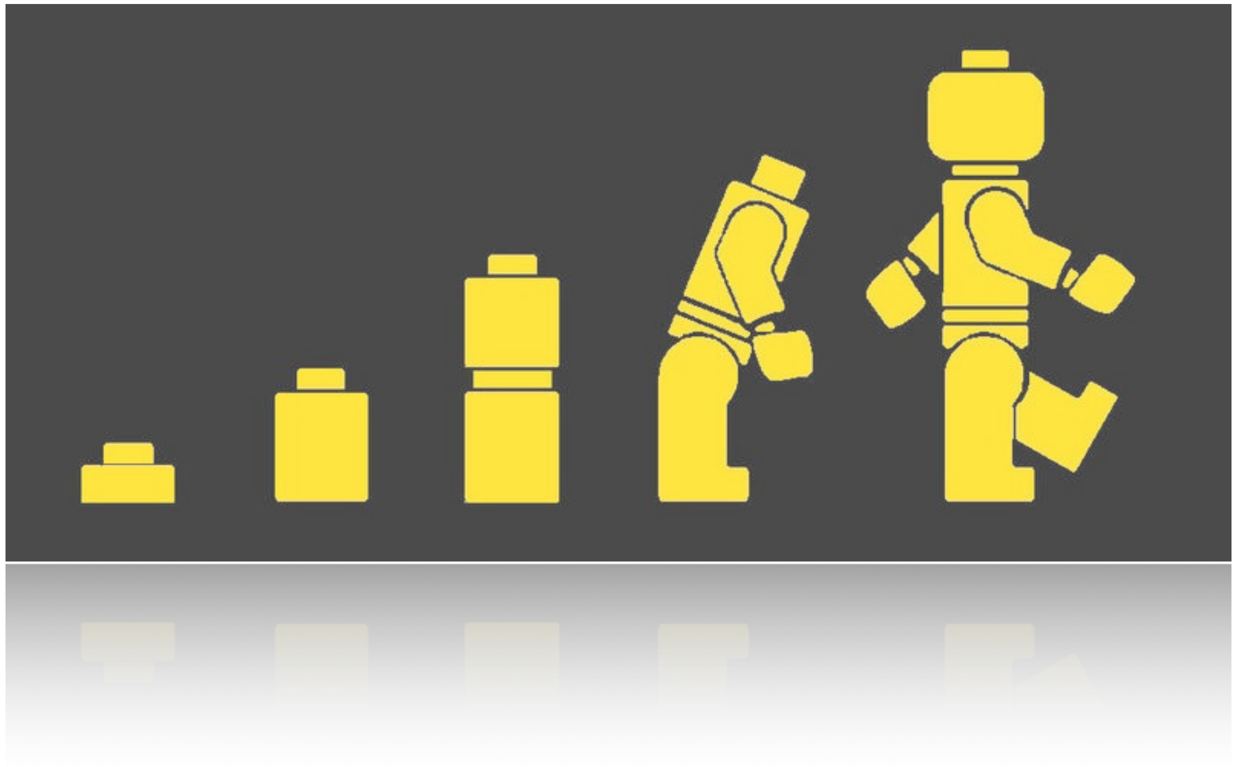
How long? Since before Big Data. Since the 1990s.

- computers and digital records on the rise
- data increasingly generated, stored, and transferred in greater volumes by more people and machines.
- the Web was gaining traction beyond Gopher and Veronica
- more and more data from a hodgepodge of hardware, storage systems, and software platforms led to problems with data storage and accessibility affecting overall quality.

Data Quality

Consider the evolution of Data Management

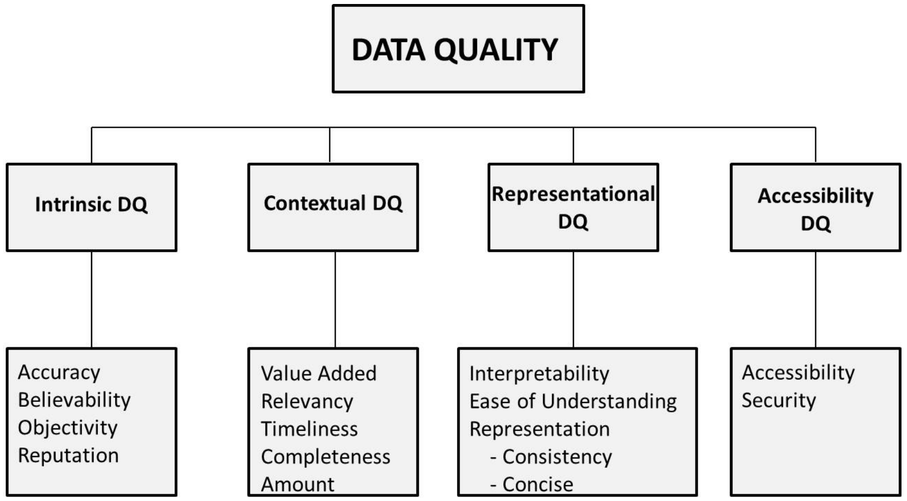
- stone tablets
- punched cards
- flat files on tape
- hierarchical databases on DASD
- network databases on disk
- relational databases
- object stores
- object-relational databases (Third Manifesto?)
- graph databases



Data Quality

Consider the evolution of Data Management

- stone tablets
- punched cards
- flat files on tape
- hierarchical databases on DASD
- network databases on disk
- relational databases
- object stores
- object-relational databases
- graph databases



DATA QUALITY

Intrinsic DQ

Accuracy
Believability
Objectivity
Reputation

Contextual DQ

Value Added
Relevancy
Timeliness
Completeness
Amount

Representational DQ

Interpretability
Ease of Understanding
Representation
- Consistency
- Concise

Accessibility DQ

Accessibility
Security

Data Quality has been a **big deal** in all data management technologies for the last 30 years.

If blockchain is to flourish and evolve, Data Quality has to be a part of it.

```
< cue dramatic music />
```

Source: Wang, R. Y. and Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4), 5-33.

Data Quality Dimensions

Accessibility

Accuracy

Appropriate Amount

Believeability

Coherence

Compatibility

Completeness

Representation

Consistency

Ease of Manipulation

Fitness for Use

Free of error

Interpretibility

Objectivity

Precision

Relevance

Reputation

Security

Specificity

Timeliness

Understandability

Value-Added

Sources:
Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.
Wang, R. Y. and Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4), 5-33.

Data Quality Dimensions

Accessibility

Accuracy

Appropriate Amount

Believeability

Coherence

Compatibility

Completeness

Representation

Consistency

Ease of Manipulation

Fitness for Use

Free of error

Interpretibility

Objectivity

Precision

Relevance

Reputation

Security

Specificity

Timeliness

Understandability

Value-Added

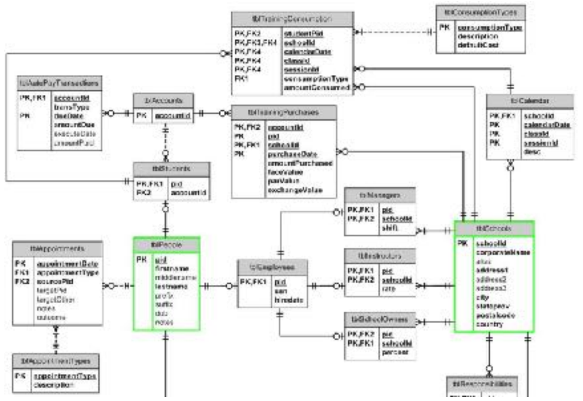
Some dimensions are well studied, particularly in the *relational* world, because they are well defined. But things ~~change~~ evolve and there are more possibilities…

Sources:
Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.
Wang, R. Y. and Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4), 5-33.

Dynamic Data Quality

Modern data comes in many formats, structures, representations.

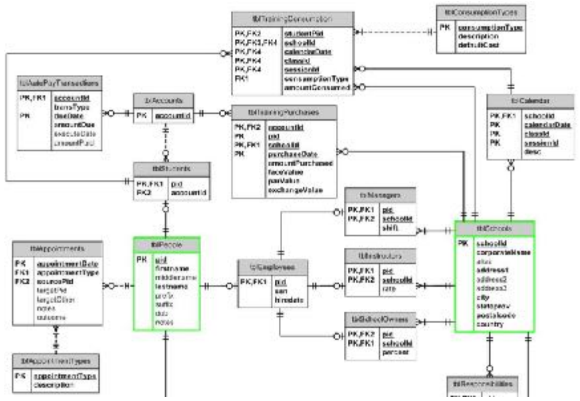
- One size does not fit all.
  - **Relational systems** are well suited for managing data structured as tables of rows and columns and performing common analytic tasks that graph systems are bad at such as creating segmentations based on attributes and combining data based on matching values.



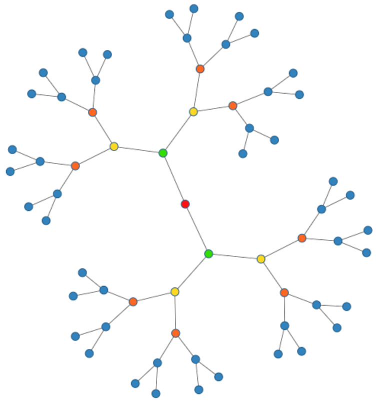
Dynamic Data Quality

Modern data comes in many formats, structures, representations.

- One size does not fit all.
  - **Relational systems** are well suited for managing data structured as tables of rows and columns and performing common analytic tasks that graph systems are bad at such as creating segmentations based on attributes and combining data based on matching values.



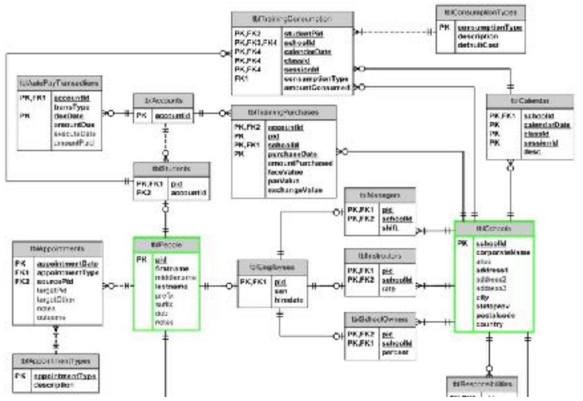
  - **Graph systems** are well suited for managing data structured as vertices and edges and performing common analytic tasks that relational systems are bad at such as finding clusters, determining shortest paths, and computing influence.



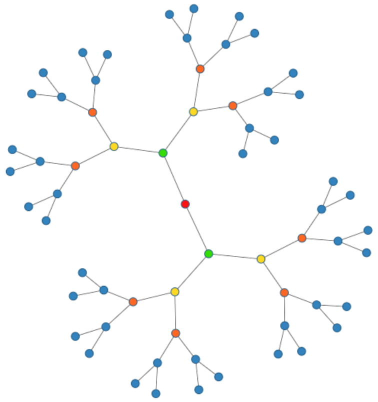
Dynamic Data Quality

Modern data comes in many formats, structures, representations.

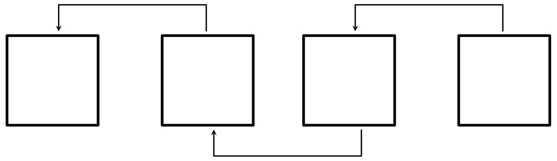
- One size does not fit all.
  - **Relational systems** are well suited for managing data structured as tables of rows and columns and performing common analytic tasks that graph systems are bad at such as creating segmentations based on attributes and combining data based on matching values.



  - **Graph systems** are well suited for managing data structured as vertices and edges and performing common analytic tasks that relational systems are bad at such as finding clusters, determining shortest paths, and computing influence.



  - **Blockchain systems** are well suited for managing append-only data preserved in trusted permanent stasis.



- The general challenge: **Fitness for Use** over time.

Dynamic Data Quality

We live in an evolving world.
Data is dynamic.
Our needs change.
Therefore Data Quality is dynamic.

Dynamic Data Quality requires flexible approaches for recasting the structure and representation of data as our needs change.

Source: Labouseur, A.G. & Matheus, C.C. (2017). An introduction to dynamic data quality challenges. *Journal of Data and Information Quality 8*(2), 6:1–6:3.

Dynamic Data Quality

We live in an evolving world.
Data is dynamic.
Our needs change.
Therefore Data Quality is dynamic.

Dynamic Data Quality requires flexible approaches for recasting the structure and representation of data as our needs change.

Questions for another time:

- What happens to Data Quality dimensions as we change the underlying representation of the data?
- What Data Quality trade-offs occur when we cast data from one representation to another?
- Can we enhance Data Quality as a side effect of changing its representation?

The question for now is…

Source: Labouseur, A.G. & Matheus, C.C. (2017). An introduction to dynamic data quality challenges. *Journal of Data and Information Quality* 8(2), 6:1–6:3.

Dynamic Data Quality

Data Quality is dynamic.

But blockchain is static.

How can we align Dynamic Data Quality with a static structure like blockchain?

The friction between static blockchain and dynamic data quality gives rise to new research opportunities.

Dynamic Data Quality Dimensions

Accessibility

Accuracy

Appropriate Amount

Believeability

Coherence

Compatibility

Completeness

Representation

Consistency

Ease of Manipulation

Fitness for Use

Free of error

Interpretibility

Objectivity

Precision

Relevance

Reputation

Security

Specificity

Timeliness

Understandability

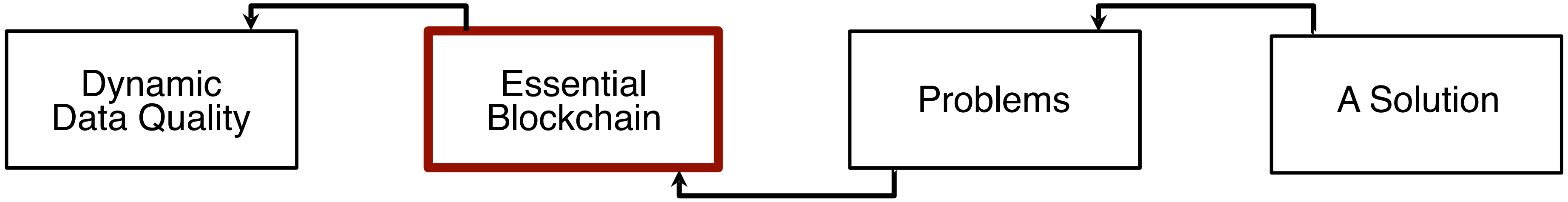
Value-Added

We consider these dimensions in the blockchain context.

But first…

Sources:
Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.
Wang, R. Y. and Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4), 5-33.

Essential Blockchain



Dynamic Data Quality

Essential Blockchain

Problems

A Solution

Essential Blockchain

What is essential “*blockchain*-ness” ?

Defining essential blockchain lets us avoid getting mired in (trivial and non-trivial) variations found among Bitcoin, Ethereum, Hyperledger, and all of the other blockchain implementations.

HYPERLEDGER

IBM

ETHEREUM

Essence and Accidents

From Aristotle…

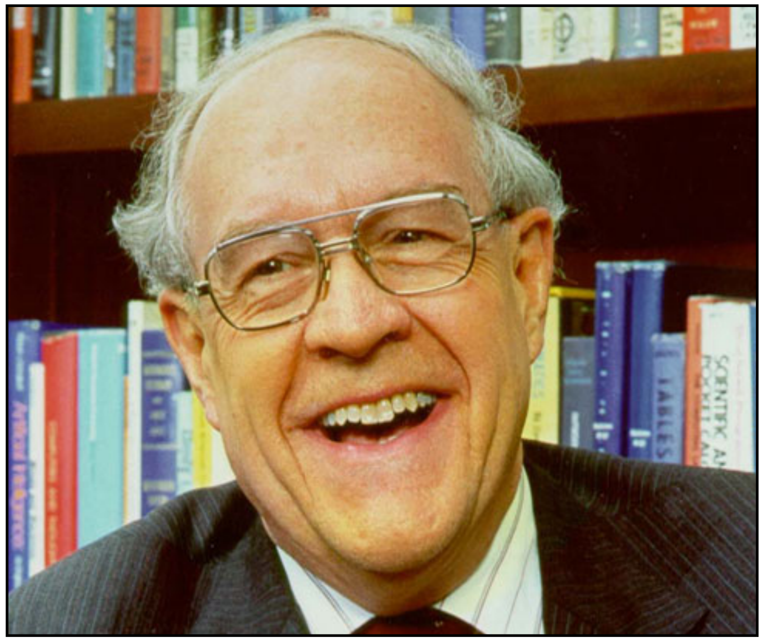
- Aristotle
  - *Categories* (350 BCE) — a philosophy of substance and being
  - four-fold system of classification:
    - accidental universals
    - essential universals
    - accidental particulars
    - non-accidental particulars

Source: Stanford Encyclopedia of Philosophy - https://plato.stanford.edu/entries/aristotle-categories/

Essence and Accidents

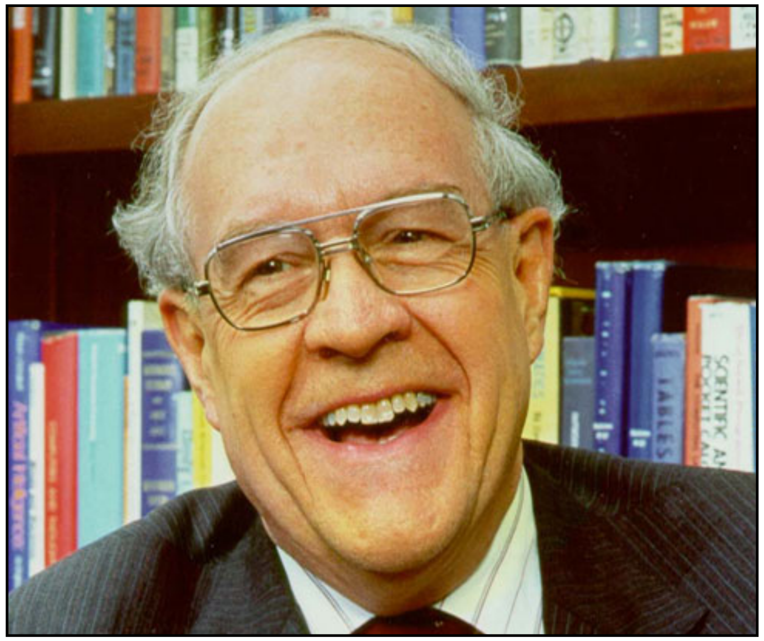
From Aristotle to Fred Brooks

- Fredrick Brooks, in “No Silver Bullet” (1987), on the difficulties inherent in software development:
  - bridges the chaotic world of *arbitrary complexity, forced without rhyme or reason by many human institutions and systems* with the abstract, yet precise domain software affords.



Essence and Accidents

From Aristotle to Fred Brooks



- Fredrick Brooks, in “No Silver Bullet” (1987), on the difficulties inherent in software development:
  - bridges the chaotic world of *arbitrary complexity, forced without rhyme or reason by many human institutions and systems* with the abstract, yet precise domain software affords.
  - As a result, managing complexity is a primary software development challenge.
  - Complexity can be broken down into two Aristotelian areas:
    - **Essence** Difficulties inherent in the nature of software
    - **Accidents** Difficulties that attend its production but are not inherent
  - Blockchain can be broken down into the same two Aristotelian areas.
  - We’re interested in blockchain’s *essence*.

Essential Blockchain

Transaction

Container for arbitrary data

“Yuzhe graduated with a 3.6 GPA from Marist”

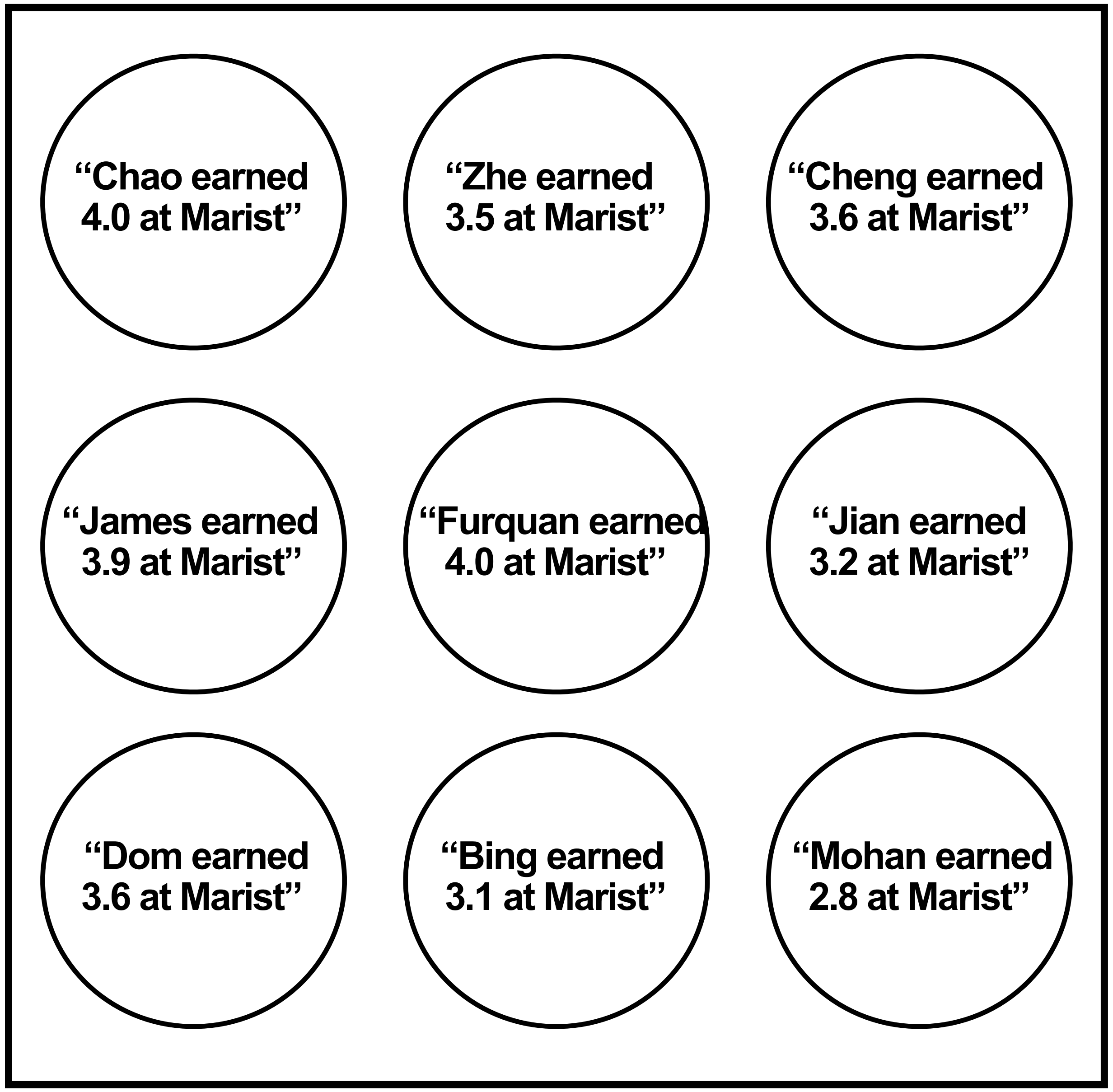
“Matt transfers 2112 ICDE-coins to Alan”

“James buys 42 shares of BitBook”

Block

Container for transactions

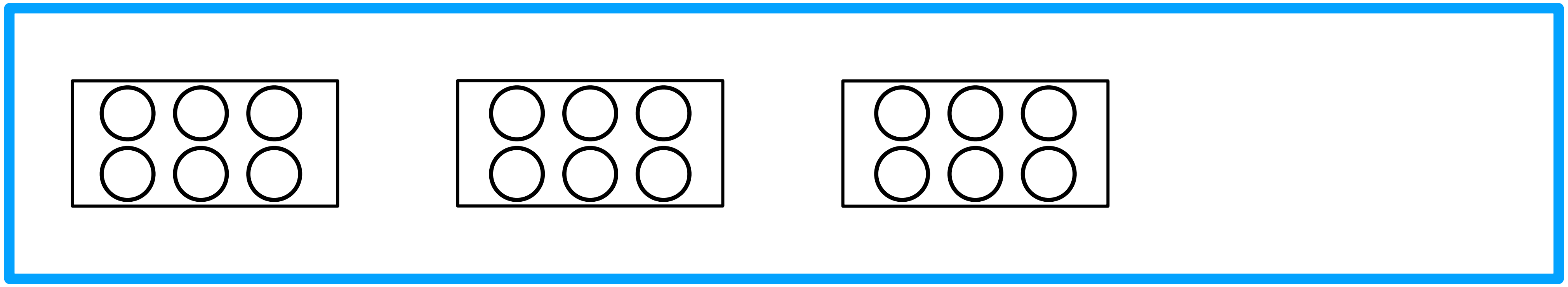
- Created by grouping transactions
- Groupings often span a time period or some limit of transactions.


“Chao earned 4.0 at Marist”
“Zhe earned 3.5 at Marist”
“Cheng earned 3.6 at Marist”
“James earned 3.9 at Marist”
“Furquan earned 4.0 at Marist”
“Jian earned 3.2 at Marist”
“Dom earned 3.6 at Marist”
“Bing earned 3.1 at Marist”
“Mohan earned 2.8 at Marist”

Essential Blockchain

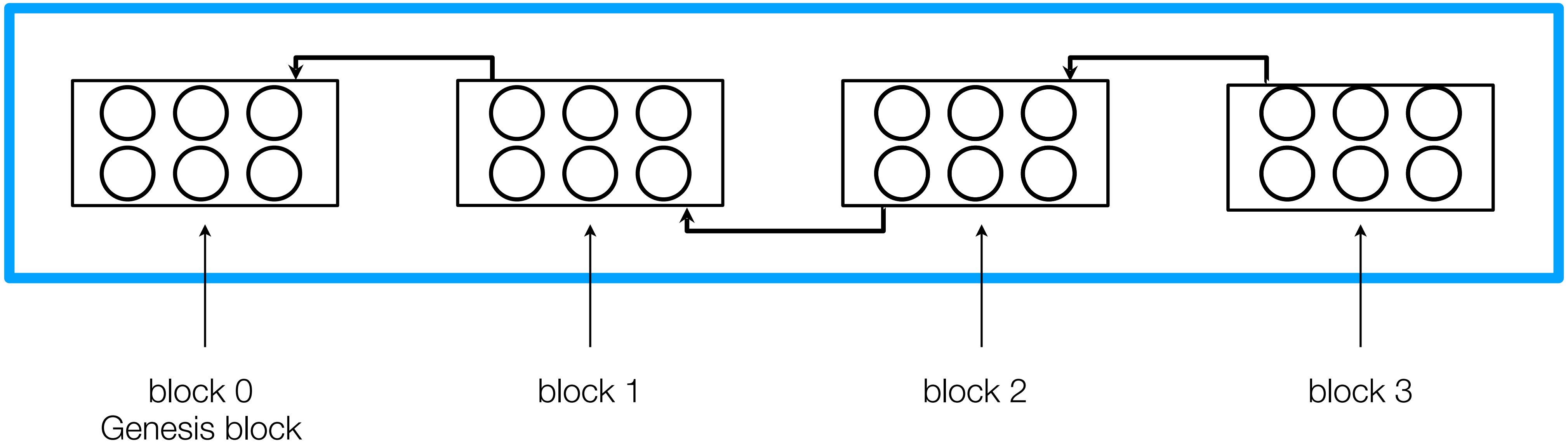
Blockchain

Append-only container for one or more blocks …



Blockchain

Append-only container for one or more blocks where **blocks are ordered**, and where the i^th^ block b_i depends on the prior block b_{i-1} to confirm b_i's **permanent stasis** where $i \geq 1$.



Essential Blockchain

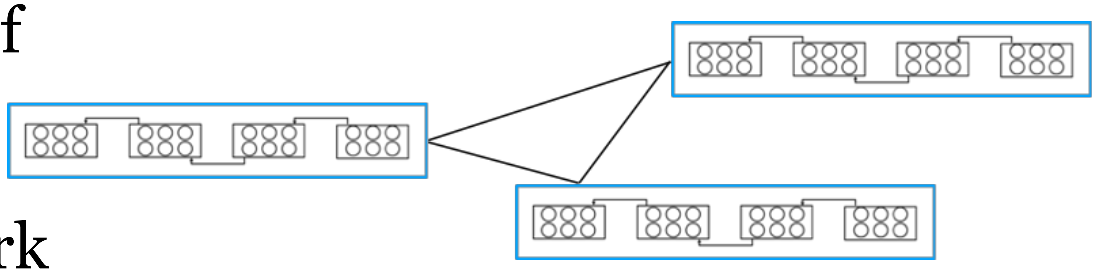
Essential “*blockchain*-ness”

Transaction – a container for arbitrary data.

Block – a container for one or more transactions.

Blockchain – an append-only container for one or more blocks, where blocks are ordered, and where the i^{th} block b_i depends on the prior block b_{i-1} to confirm b_i’s permanent stasis where $i \geq 1$.

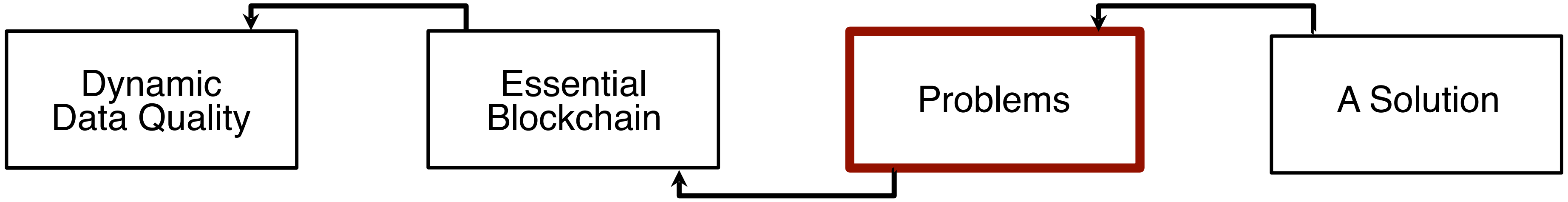
Blockchain is more than a data structure.
It’s also a consensus network of peer instances of that data structure.



Essential Blockchain – a peer-to-peer network of blockchain instances cooperating for consensus.

With this powerful abstraction we are now ready to explore **dynamic** problems of *accessibility*, *representation*, and general *fitness for use* in the **static** world of blockchain.

Problems


Dynamic Data Quality
Essential Blockchain
Problems
A Solution

Problems

General challenges in Dynamic Data Quality: *fitness for use*.

Some specific challenges: *availability* and *retrievability*

Other challenges involve transforming data into varying *formats* and *representations* to fit our evolving needs for its use. Remember, we'd like to…

- use relational tables to slice and dice our data into segments
- use graphs for measuring influence and finding clusters
- use blockchain for distributed trust

These problems of Dynamic Data Quality are currently being explored in the context of graph and relational systems.

We explore them in the context of blockchain.

Problem: Accessibility

… the extent to which data are *available* and *retrievable*.

- encompasses data in both detail and aggregate form
- covers whether data are *formatted* and *represented* to be easily retrievable for a desired task.
- includes time lapse spanning request, retrieval, and delivery

Source: C. Batini, et. al, Methodologies for data quality assessment and improvement," *ACM Computing Surveys, vol. 41, no. 3, pp. 16:1–16:52*

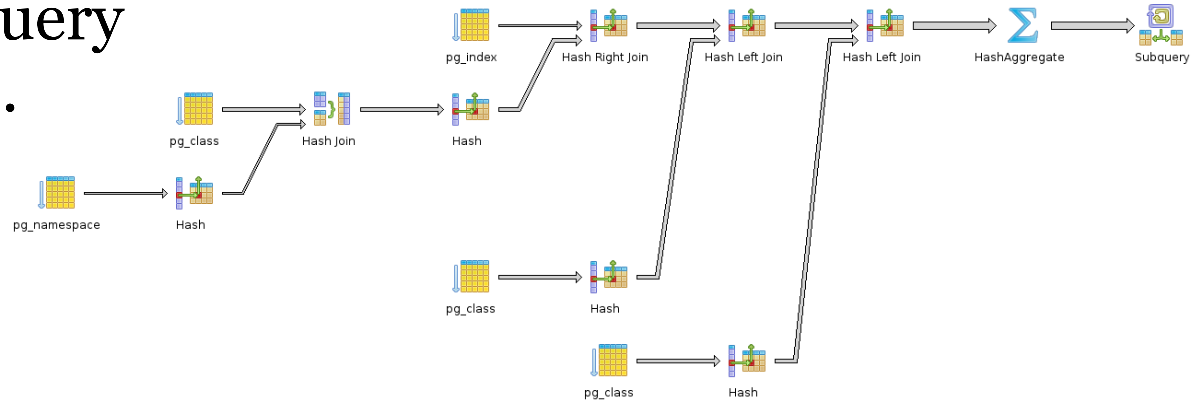
Problem: Accessibility

… the extent to which data are *available* and *retrievable.*

- encompasses data in both detail and aggregate form
- covers whether data are *formatted* and *represented* to be easily retrievable for a desired task.
- includes time lapse spanning request, retrieval, and delivery

Query performance is often used to measure *accessibility*

- Addressed in traditional systems with query optimization and indexes or summaries.



Source: C. Batini, et. al, Methodologies for data quality assessment and improvement," *ACM Computing Surveys, vol. 41, no. 3, pp. 16:1–16:52*

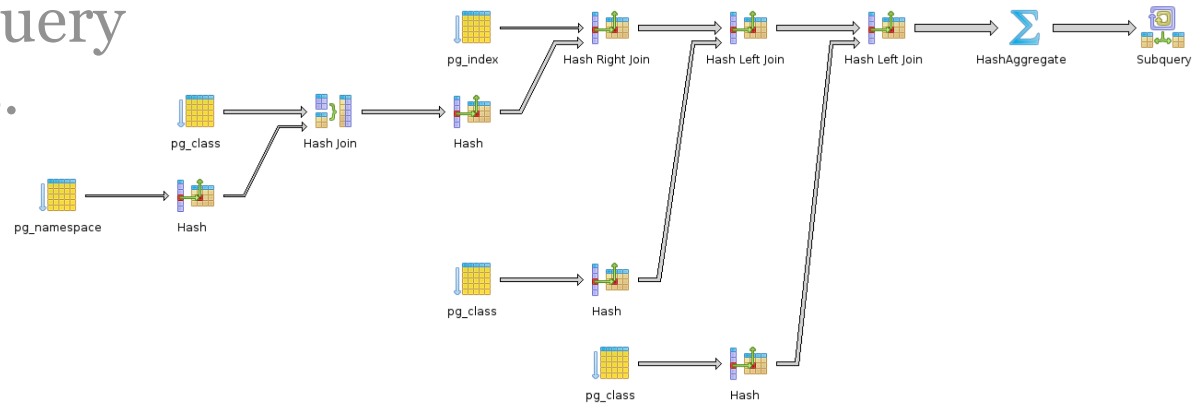
Problem: Accessibility

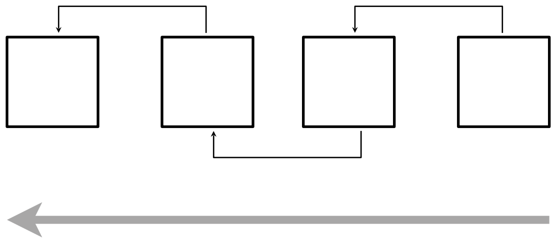
… the extent to which data are *available* and *retrievable*.

- encompasses data in both detail and aggregate form
- covers whether data are *formatted* and *represented* to be easily retrievable for a desired task.
- includes time lapse spanning request, retrieval, and delivery

Query performance is often used to measure *accessibility*

- Addressed in traditional systems with query optimization and indexes or summaries.
- Problem for blockchain because we cannot generally **query** a blockchain in the common sense of the word.
- Rather, we must **crawl** from the most recent block backwards towards the Genesis block, searching.
- Without structures and metadata to support log-time search functions, we are stuck with linear search.





Source: C. Batini, et. al, Methodologies for data quality assessment and improvement," *ACM Computing Surveys, vol. 41, no. 3, pp. 16:1–16:52*

Problem: Representation

… the extent to which data are *concisely presented*, *well organized*, and *well formatted* for extracting meaningful information.

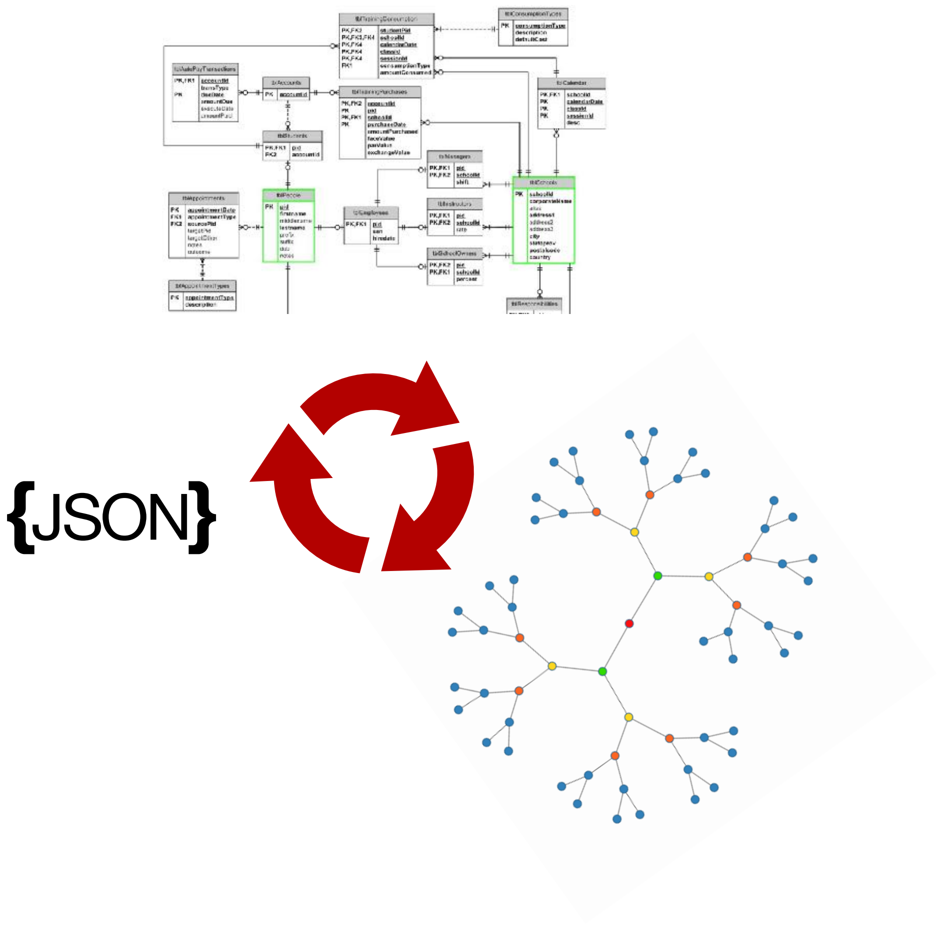
- Meaning requires context, which changes and evolves over time.

Source: Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.

Problem: Representation

… the extent to which data are *concisely presented*, *well organized*, and *well formatted* for extracting meaningful information.

- Meaning requires context, which changes and evolves over time.
- Addressed in traditional systems with flexibility to change the underlying format of our data to align with our dynamic *fitness for use* needs.
  - Example: Data initially captured in JSON format but later transformed to a graph for influence queries and then to relational tables for segmentations.

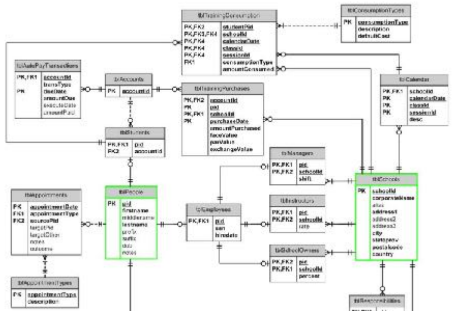


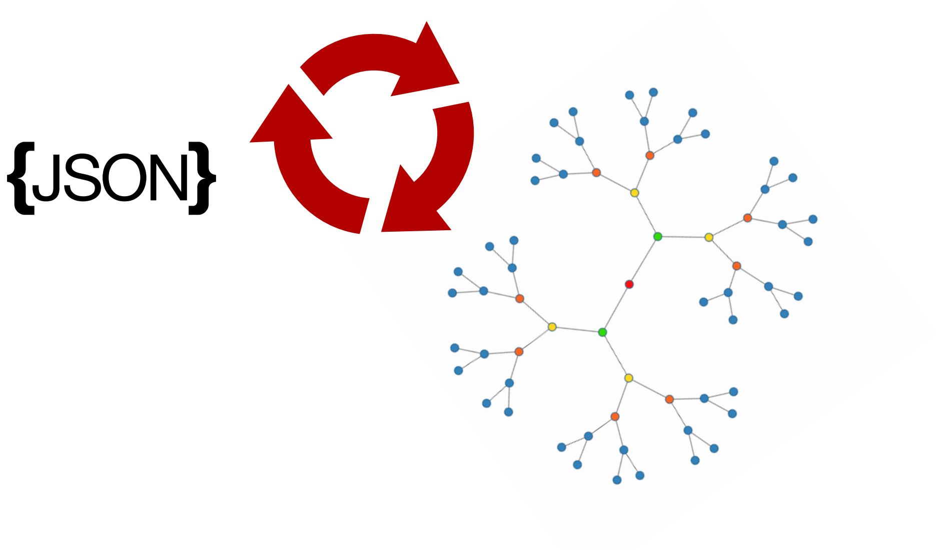
Source: Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.

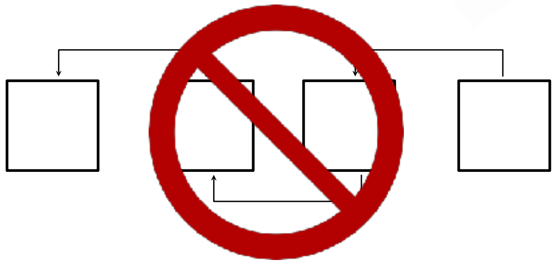
Problem: Representation

… the extent to which data are *concisely presented*, *well organized*, and *well formatted* for extracting meaningful information.

- Meaning requires context, which changes and evolves over time.
- Addressed in traditional systems with flexibility to change the underlying format of our data to align with our dynamic *fitness for use* needs.
  - Example: Data initially captured in JSON format but later transformed to a graph for influence queries and then to relational tables for segmentations.
- Problem for blockchain because its essential **static** nature does not permit flexibility to change its underlying format to suit our dynamic needs.
  - Any representation that requires crawling potentially lengthy portions of a blockchain to extract meaningful information cannot be considered concise.



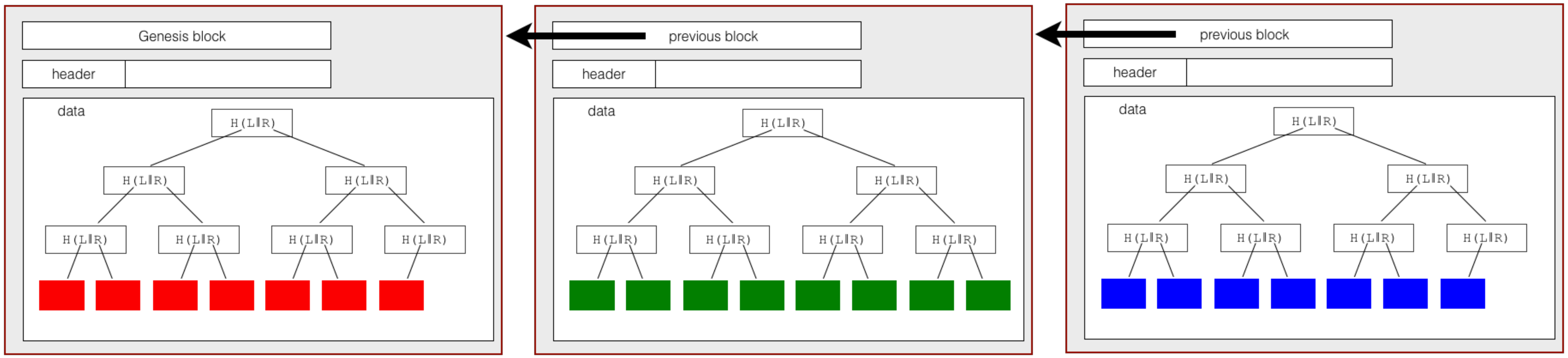




Source: Pipino, L.L., Lee, Y.W., & Wang, R.Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.

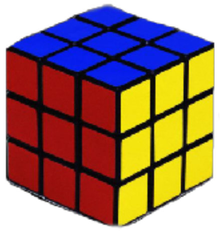
Problems

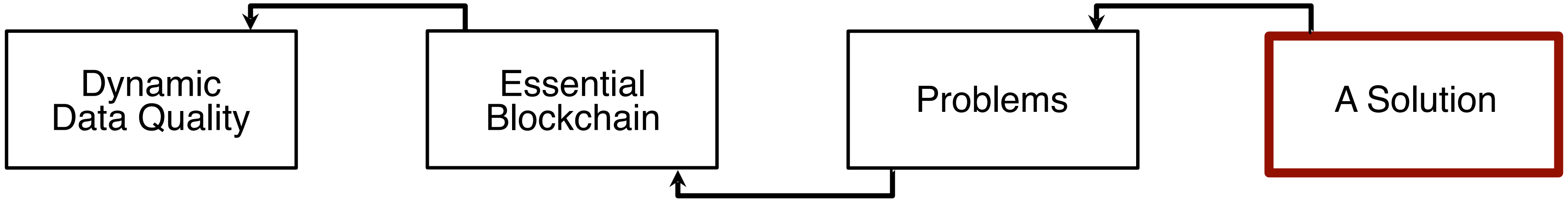
Problems of *accessibility* and *representation* stem from **misalignment** between these dynamic data quality dimensions and the essential static and linear nature of blockchain.



tiny blockchain

A Solution





Dynamic Data Quality

Essential Blockchain

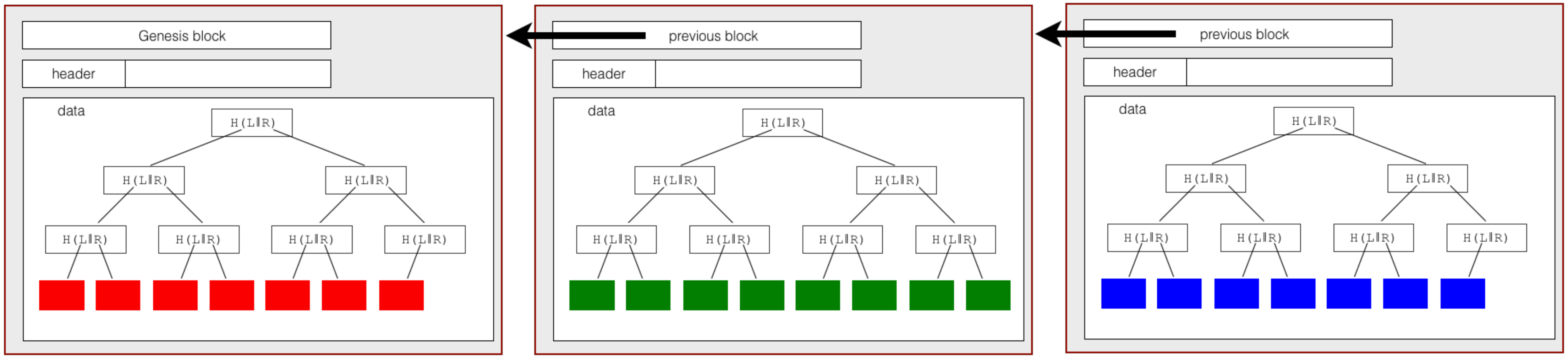
Problems

A Solution

A Solution

We can align Dynamic Data Quality with a static structure like blockchain by using graphs.

Here's our tiny blockchain with transactions in red, green, and blue.



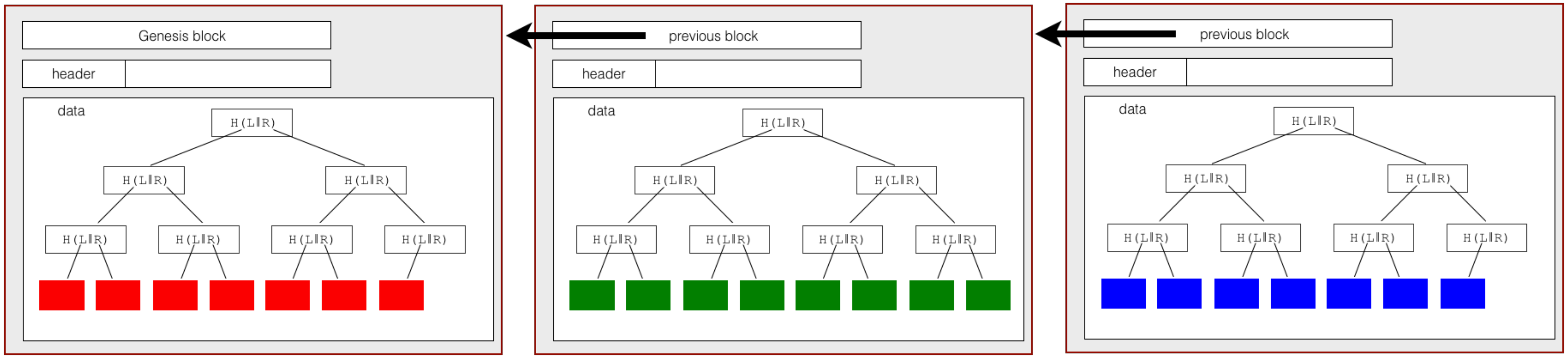
Blockchains are naturally graph-like.

- Blocks form a linked list, a special case of a graph.
- Transactions are leaf nodes of a (Merkle) tree, also a special case of a graph.

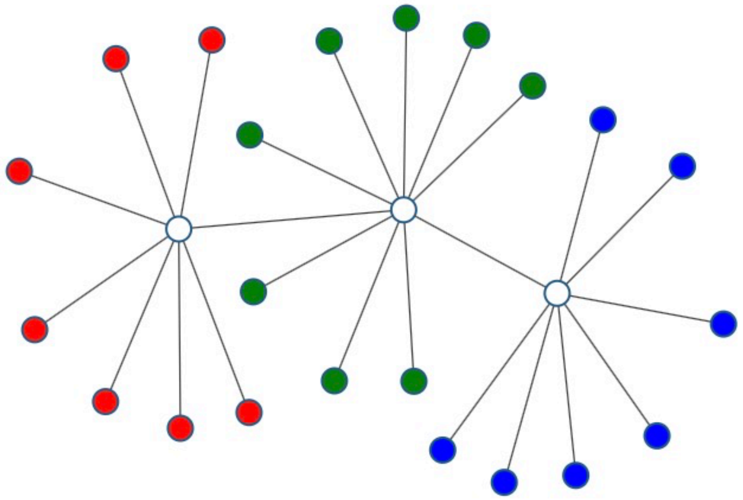
A Solution: Blockchain Snapshots as a Graph

We can align Dynamic Data Quality with a static structure like blockchain by using graphs.

Here’s our tiny blockchain with transactions in red, green, and blue.



Here’s a tiny graph with transaction vertices in red, green, and blue.



How?

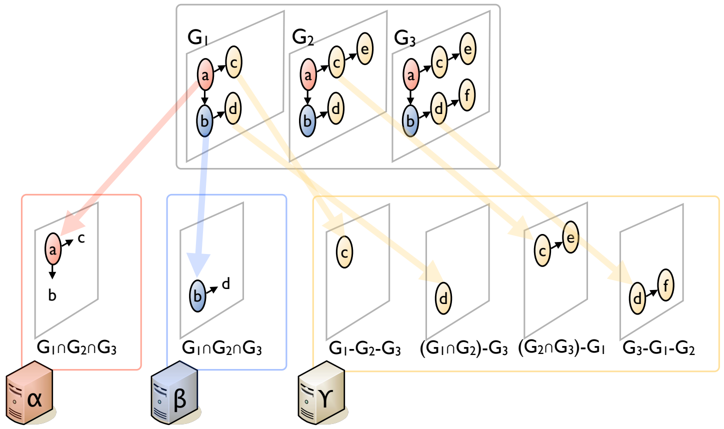
A Solution with Graph Tools

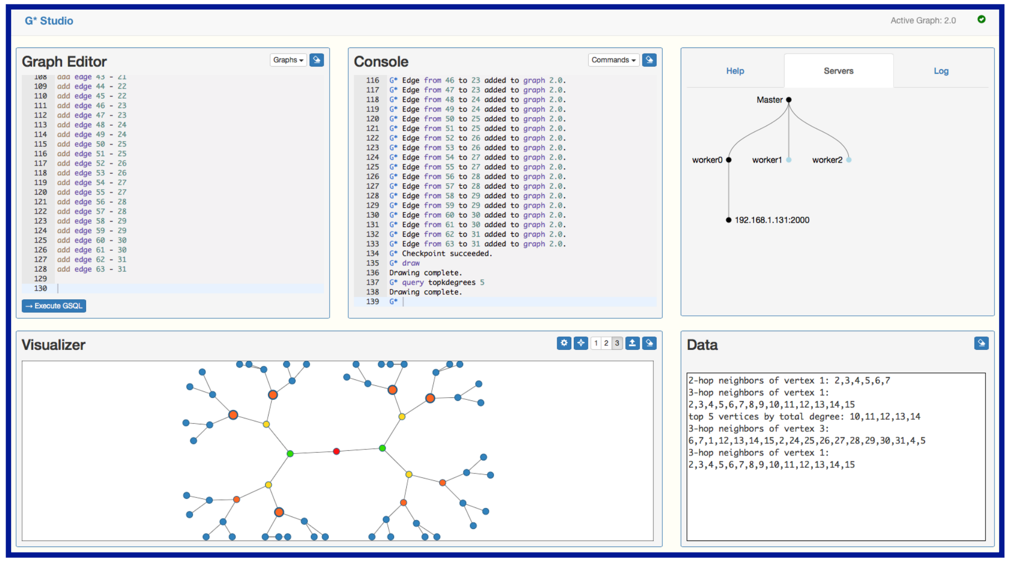
Blockchains are naturally graph-like. So we can use graph tools.

- Distributed graph databases can handle high velocity high volume data.

G*studio

G* The Dynamic Graph Database

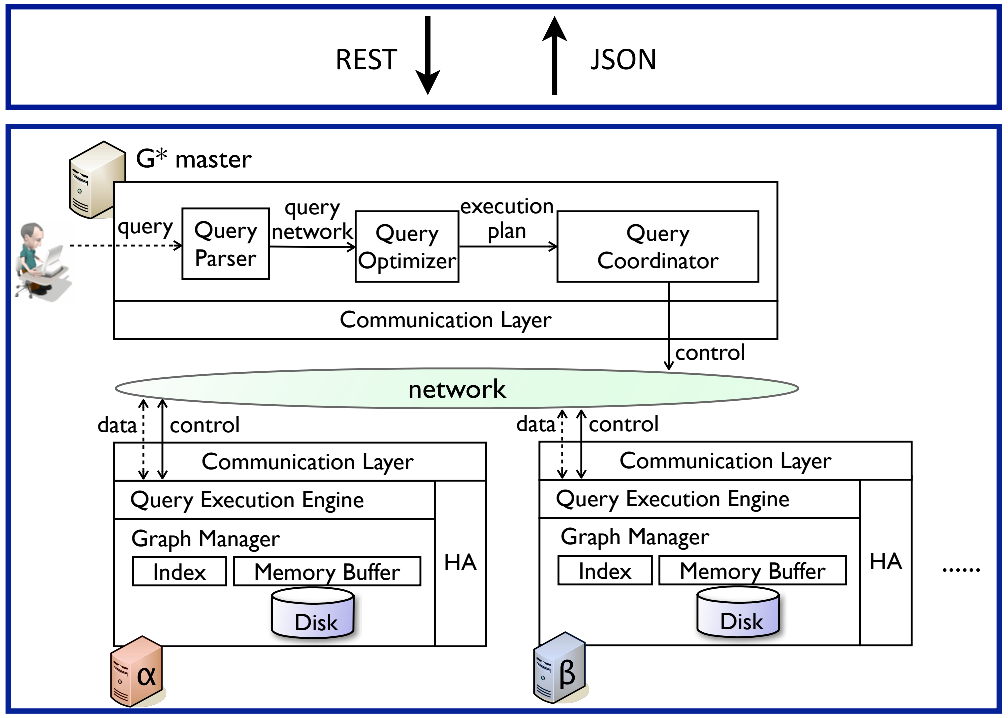




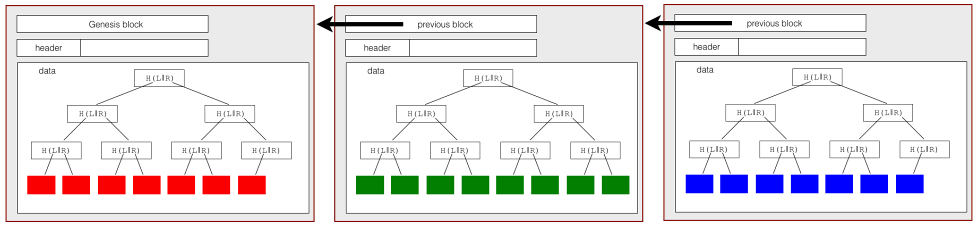
```
add edge 43 - 21
add edge 44 - 22
add edge 45 - 22
add edge 46 - 23
add edge 47 - 23
add edge 48 - 24
add edge 49 - 24
add edge 50 - 25
add edge 51 - 25
add edge 52 - 26
add edge 53 - 26
add edge 54 - 27
add edge 55 - 27
add edge 56 - 28
add edge 57 - 28
add edge 58 - 29
add edge 59 - 29
add edge 60 - 30
add edge 61 - 30
add edge 62 - 31
add edge 63 - 31
```

```
G* Edge from 46 to 23 added to graph 2.0.
G* Edge from 47 to 23 added to graph 2.0.
G* Edge from 48 to 24 added to graph 2.0.
G* Edge from 49 to 24 added to graph 2.0.
G* Edge from 50 to 25 added to graph 2.0.
G* Edge from 51 to 25 added to graph 2.0.
G* Edge from 52 to 26 added to graph 2.0.
G* Edge from 53 to 26 added to graph 2.0.
G* Edge from 54 to 27 added to graph 2.0.
G* Edge from 55 to 27 added to graph 2.0.
G* Edge from 56 to 28 added to graph 2.0.
G* Edge from 57 to 28 added to graph 2.0.
G* Edge from 58 to 29 added to graph 2.0.
G* Edge from 59 to 29 added to graph 2.0.
G* Edge from 60 to 30 added to graph 2.0.
G* Edge from 61 to 30 added to graph 2.0.
G* Edge from 62 to 31 added to graph 2.0.
G* Edge from 63 to 31 added to graph 2.0.
G* Checkpoint succeeded.
G* draw
Drawing complete.
G* query topkdegrees 5
Drawing complete.
G*
```

```
2-hop neighbors of vertex 1: 2,3,4,5,6,7
3-hop neighbors of vertex 1:
2,3,4,5,6,7,8,9,10,11,12,13,14,15
top 5 vertices by total degree: 10,11,12,13,14
3-hop neighbors of vertex 3:
6,7,1,12,13,14,15,2,24,25,26,27,28,29,30,31,4,5
3-hop neighbors of vertex 1:
2,3,4,5,6,7,8,9,10,11,12,13,14,15
```



A Solution: Blockchain Snapshot as Graph

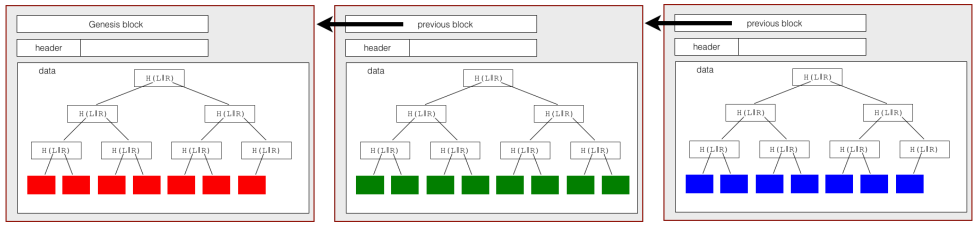


Algorithm 1: Generating a graph from a blockchain API

```
new graph;
$lastBlockId$ ← null;
$blockCount$ ← api/status?q=getBlockCount;
for $i$ ← $blockCount$-1 down to 0 do
    $hash_i$ ← /api/block-index/$i$;
    $block$ ← /api/block/$hash_i$;
    $thisBlockId$ ← "block" ‖ $i$;
    add vertex $thisBlockId$;
    $transactions[]_i$ ← /api/txs/?block=$hash_i$;
    foreach $tx$ in $transactions[]_i$ do
        $thisTxId$ ← $thisBlockId$ ‖ "tx" ‖ $tx.id$;
        add vertex $thisTxId$;
        add edge $thisTxId$ – $thisBlockId$;
    end
    if $lastBlockId \neq null$ then
        add edge $lastBlockId$ – $thisBlockId$;
    end
    $lastBlockId$ ← $thisBlockId$;
end
```

`BlockExplorer` API calls
and G*Studio `DGQL` code

A Solution: Blockchain Snapshot as Graph



Algorithm 1: Generating a graph from a blockchain API

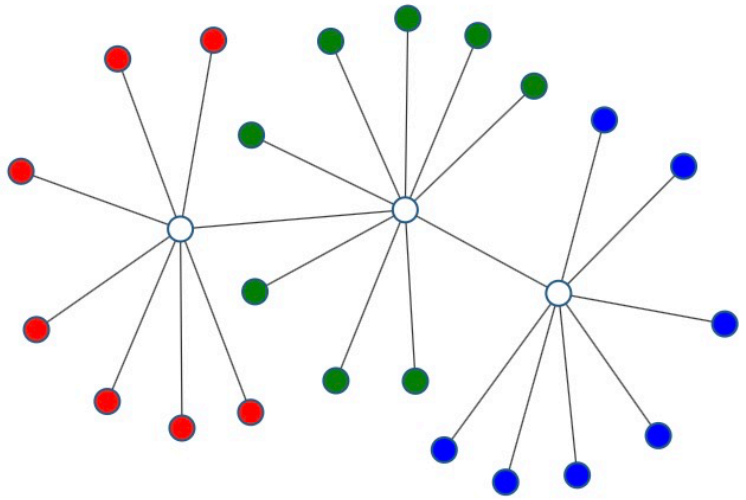
```
new graph;
$lastBlockId \leftarrow$ null;
$blockCount \leftarrow$ api/status?q=getBlockCount;
for $i \leftarrow$ blockCount-1 down to 0 do
    $hash_i \leftarrow$ /api/block-index/$i$;
    $block \leftarrow$ /api/block/$hash_i$;
    $thisBlockId \leftarrow$ "block" $\|$ $i$;
    add vertex $thisBlockId$;
    $transactions[]_i \leftarrow$ /api/txs/?block=$hash_i$;
    foreach $tx$ in $transactions[]_i$ do
        $thisTxId \leftarrow thisBlockId$ $\|$ "tx" $\|$ $tx.id$;
        add vertex $thisTxId$;
        add edge $thisTxId - thisBlockId$;
    end
    if $lastBlockId \neq null$ then
        add edge $lastBlockId - thisBlockId$;
    end
    $lastBlockId \leftarrow thisBlockId$;
end
```

```
new graph

add vertex block2 with attributes (color=white)
add vertex block2tx0 with attributes (color=blue)
add edge block2tx0-block2
add vertex block2tx1 with attributes (color=blue)
add edge block2tx1-block2
add vertex block2tx2 with attributes (color=blue)
add edge block2tx2-block2
add vertex block2tx3 with attributes (color=blue)
add edge block2tx3-block2
add vertex block2tx4 with attributes (color=blue)
add edge block2tx4-block2
add vertex block2tx5 with attributes (color=blue)
add edge block2tx5-block2
add vertex block2tx6 with attributes (color=blue)
add edge block2tx6-block2

add vertex block1 with attributes (color=white)
 ⋮
add edge block2-block1

add vertex block0 with attributes (color=white)
 ⋮
add edge block1-block0
```

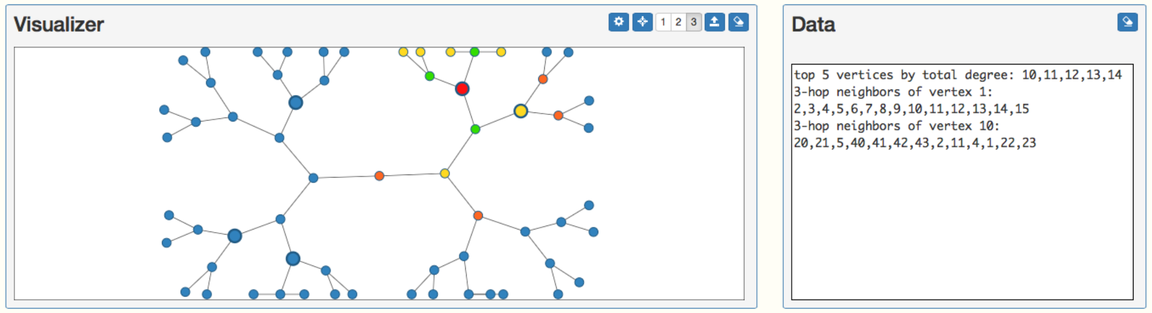


BlockExplorer API calls
and G*Studio DGQL code

A Solution for Accessibility

Improve *Accessibility* with Graph Analytics

- Perform Optimized Queries
  - top-*k* queries
  - *n*-hop neighborhoods
  - pathfinding
  - influence measures by
    - degree centrality
    - betweenness centrality
    - PageRank

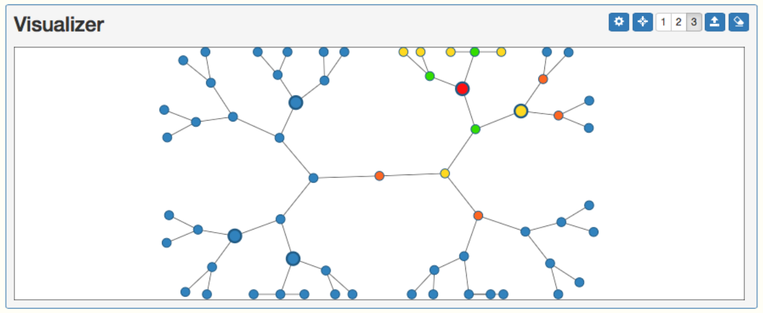


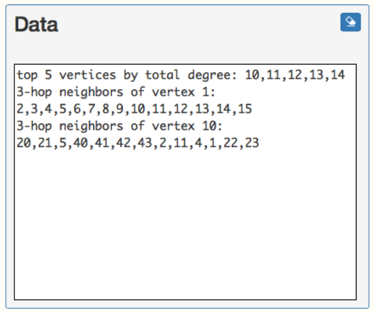
```
top 5 vertices by total degree: 10,11,12,13,14
3-hop neighbors of vertex 1:
2,3,4,5,6,7,8,9,10,11,12,13,14,15
3-hop neighbors of vertex 10:
20,21,5,40,41,42,43,2,11,4,1,22,23
```

A Solution for Accessibility

Improve *Accessibility* with Graph Analytics

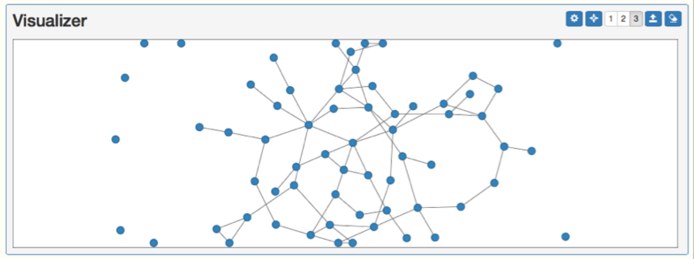
- Perform Optimized Queries
  - top-*k* queries
  - *n*-hop neighborhoods
  - pathfinding
  - influence measures by
    - degree centrality
    - betweenness centrality
    - PageRank



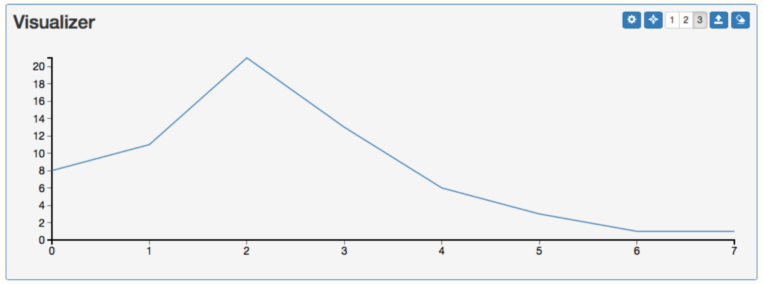

Data

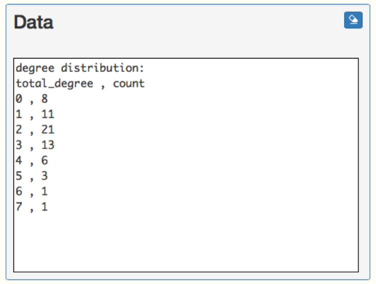
```
top 5 vertices by total degree: 10,11,12,13,14
3-hop neighbors of vertex 1:
2,3,4,5,6,7,8,9,10,11,12,13,14,15
3-hop neighbors of vertex 10:
20,21,5,40,41,42,43,2,11,4,1,22,23
```

- Discover clusters and components
  - clustering coefficient
  - connected components



- Compute aggregates and summaries
  - count and max
  - degree distribution
  - network diameter




Data

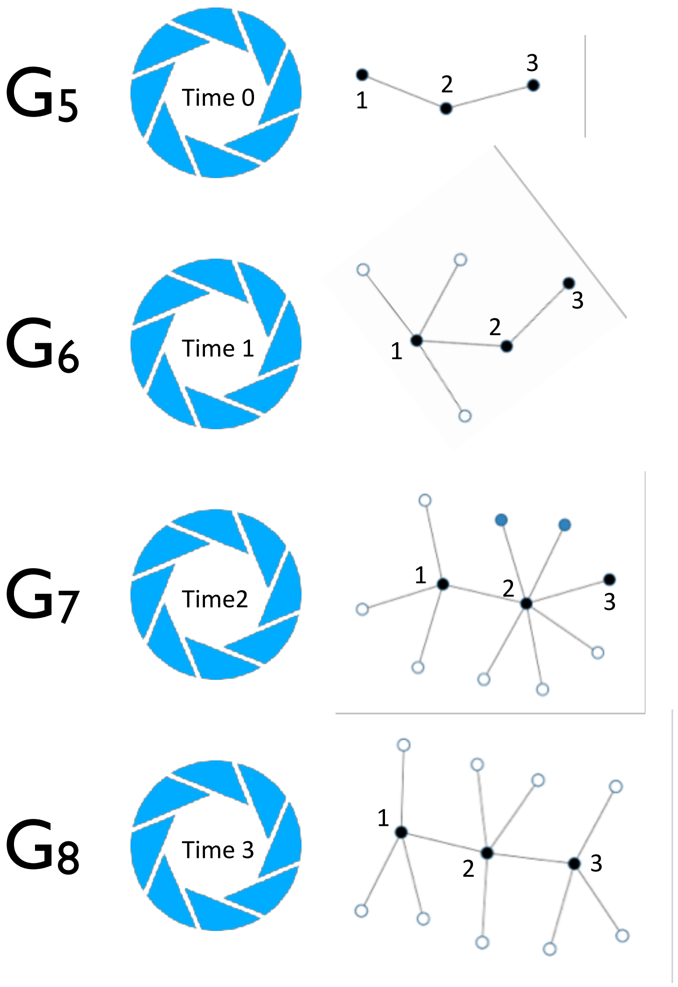
```
degree distribution:
total_degree , count
0 , 8
1 , 11
2 , 21
3 , 13
4 , 6
5 , 3
6 , 1
7 , 1
```

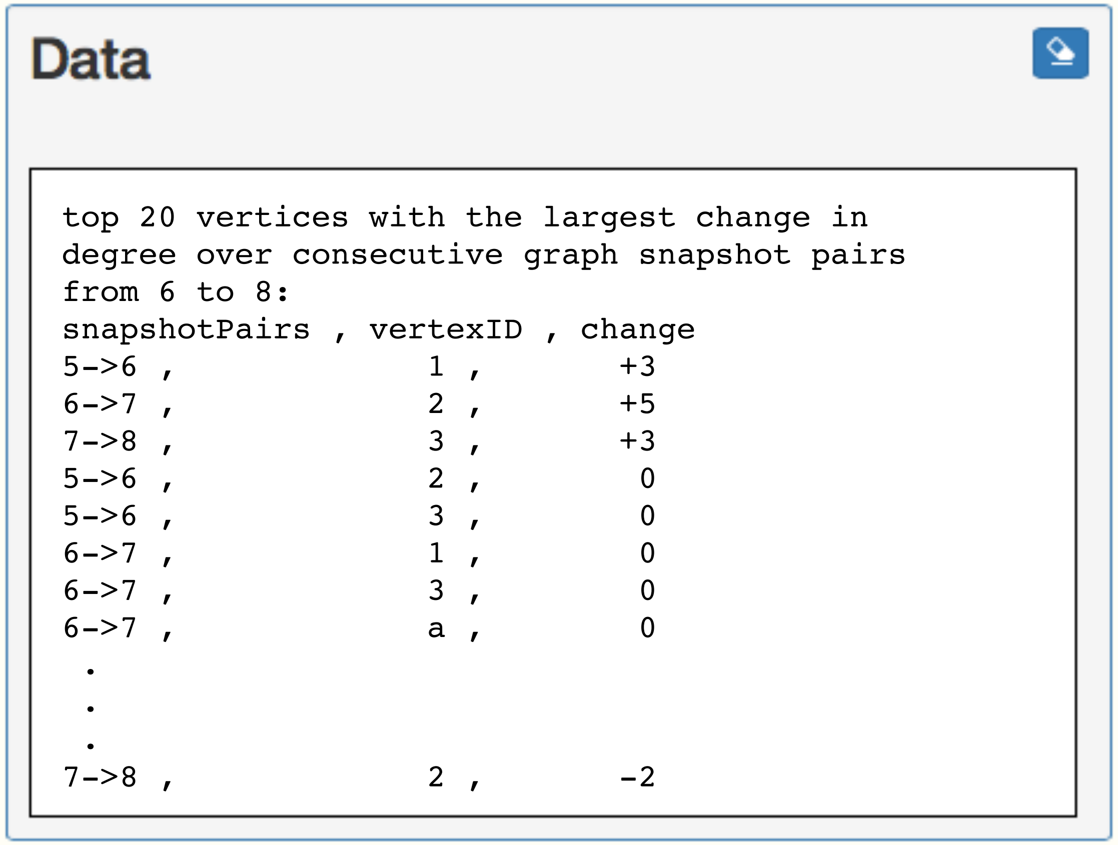
These tools are fit for resolving the misalignment between Dynamic Data Quality dimensions and static blockchains.

A Solution for Accessibility

Improve *Accessibility* with Graph Analytics

- Analyze pairwise snapshots of a blockchain peer network over time





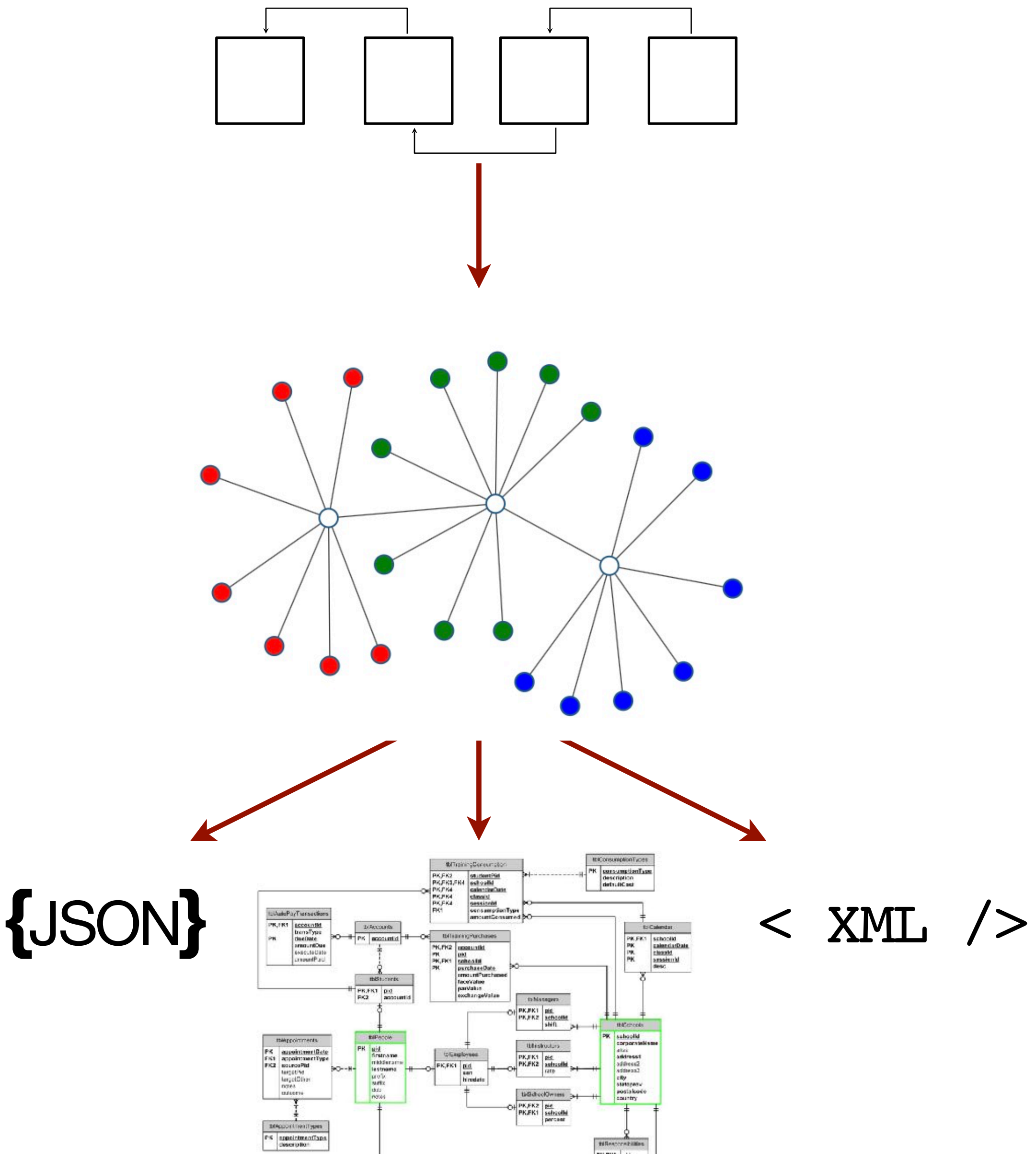
```
top 20 vertices with the largest change in
degree over consecutive graph snapshot pairs
from 6 to 8:
snapshotPairs , vertexID , change
5->6 ,           1 ,       +3
6->7 ,           2 ,       +5
7->8 ,           3 ,       +3
5->6 ,           2 ,        0
5->6 ,           3 ,        0
6->7 ,           1 ,        0
6->7 ,           3 ,        0
6->7 ,           a ,        0
 .
 .
 .
7->8 ,           2 ,       -2
```

There is interesting cyber security research to be done here.

A Solution for Representation

Improve *Representation* with Graph Storage

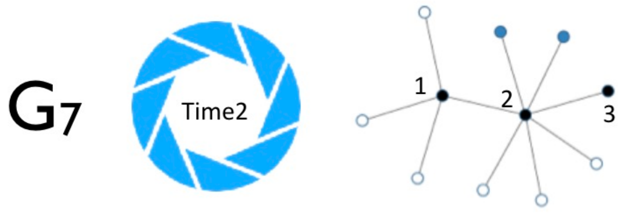
- Structural transformations with graph queries that output
  - JSON
  - SQL
  - XML
  - other formats as our needs evolve

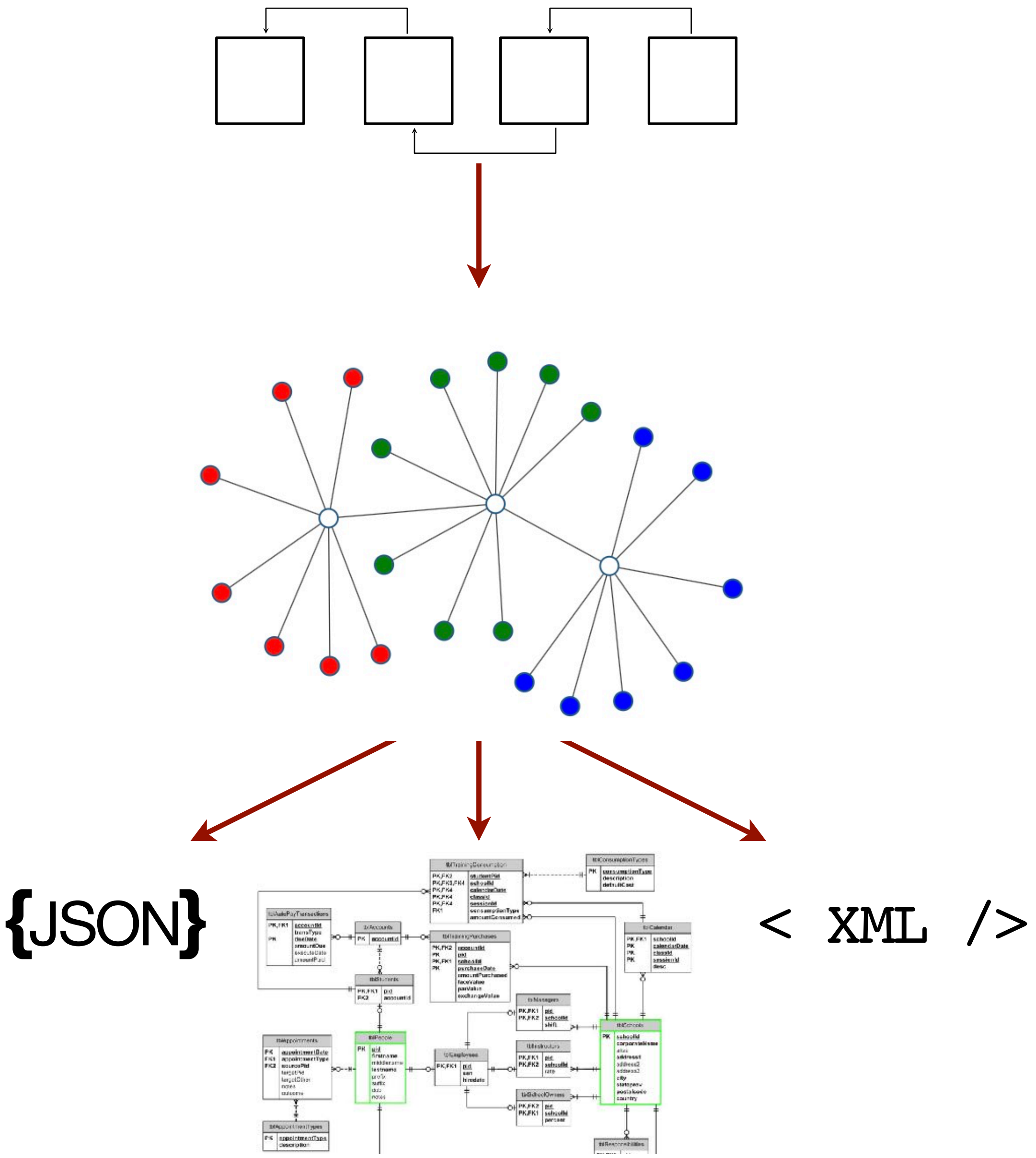


A Solution for Representation

Improve *Representation* with Graph Storage

- Structural transformations with graph queries that output
  - JSON
  - SQL
  - XML
  - other formats as our needs evolve
- Improve *Concise Representation* with summaries and snapshots
  - support query efficiency
  - aid in visualization





These tools are fit for resolving the misalignment between Dynamic Data Quality dimensions and static blockchains.

Conclusions

Graph systems can resolve the misalignment between Dynamic Data Quality dimensions and static blockchains.

- Distributed storage and optimized queries support *Accessibility*.
- Queries computing summaries and aggregates support *Concise Representation* and visualization.
- Structural transformations support general *Representation*.

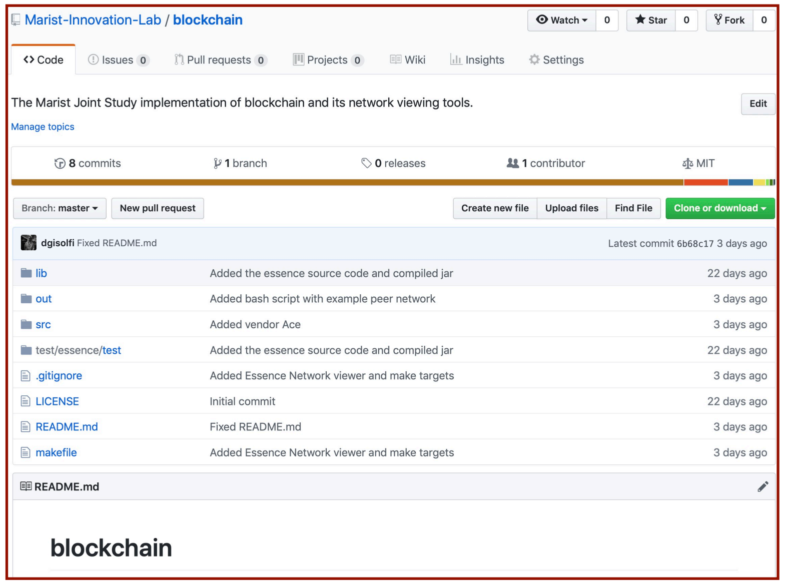
Conclusions and Future Work

Graph systems can resolve the misalignment between Dynamic Data Quality dimensions and static blockchains.

- Distributed storage and optimized queries support *Accessibility*.
- Queries computing summaries and aggregates support *Concise Representation* and visualization.
- Structural transformations support general *Representation*.

Future work

- Experiment with Algorithm 1 on larger data sets
- Develop new block structures/attributes to support summarization and log-time search functions using our *Essence Blockchain* research code
    - available to everyone at `https://github.com/Marist-Innovation-Lab/blockchain`
        - *Essential Blockchain* code in Java
        - blockchain network peer viewer (with graph snapshot export to G*Studio)
        - *Demia* demo application



The Marist Joint Study implementation of blockchain and its network viewing tools.

8 commits · 1 branch · 0 releases · 1 contributor · MIT

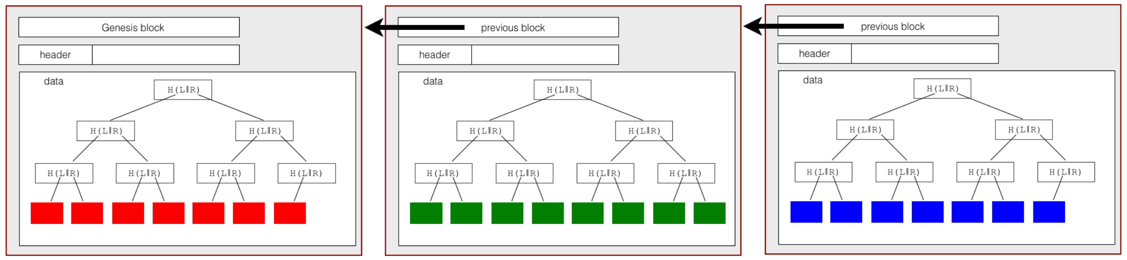
dgisolfi Fixed README.md — Latest commit 6b68c17 3 days ago

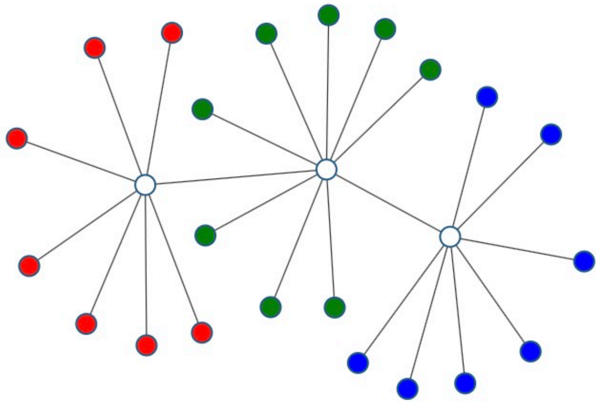
lib	Added the essence source code and compiled jar	22 days ago
out	Added bash script with example peer network	3 days ago
src	Added vendor Ace	3 days ago
test/essence/test	Added the essence source code and compiled jar	22 days ago
.gitignore	Added Essence Network viewer and make targets	3 days ago
LICENSE	Initial commit	22 days ago
README.md	Fixed README.md	3 days ago
makefile	Added Essence Network viewer and make targets	3 days ago

README.md

blockchain

Dynamic Data Quality for Static Blockchains





Thank You.

Questions? Suggestions?

BlockDM @ ICDE 2019