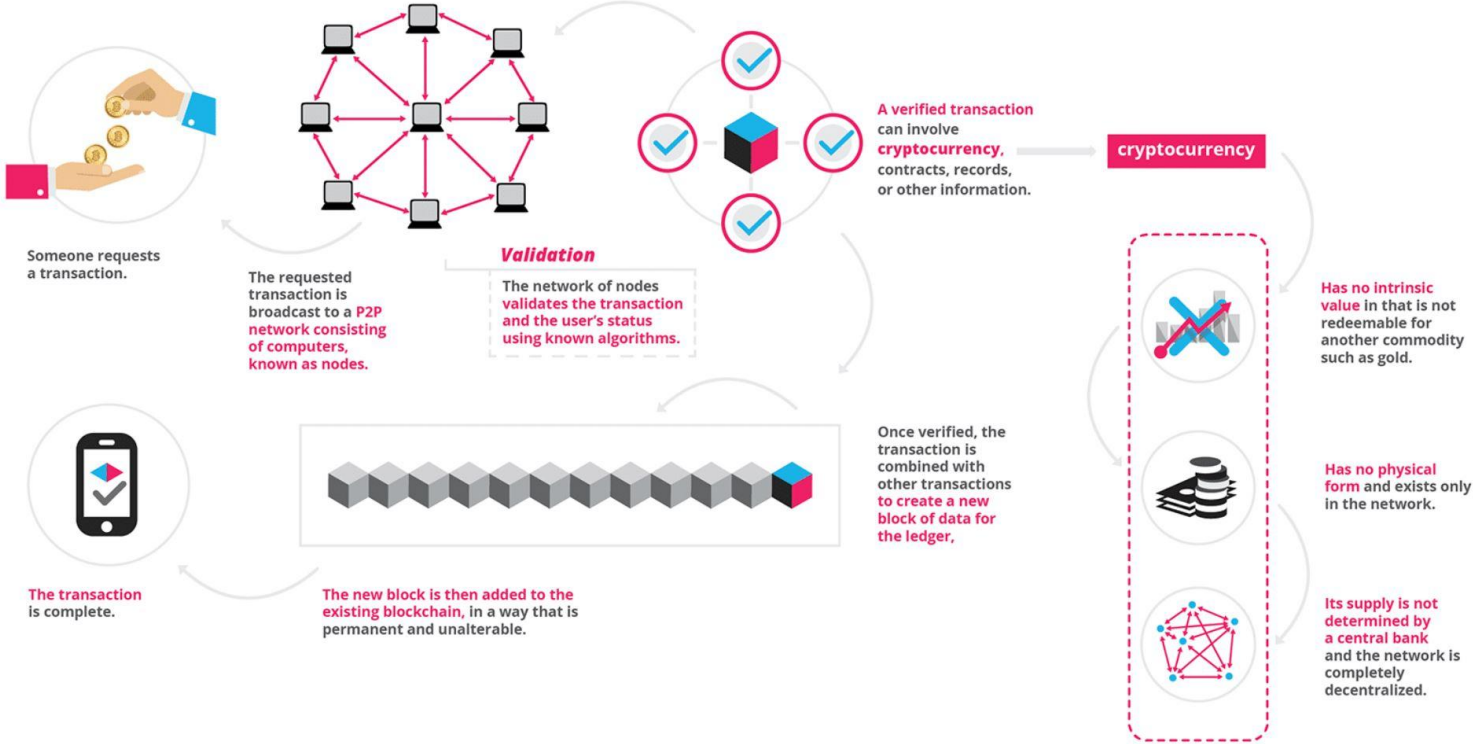
Blockchain on Relational and Nosql Databases

Vedant Chauhan (892758)
Siddharth Malhotra (934336)
Saksham Agrawal (866102)

Goal

Here, we explore the underlying storage requirements associated with blockchain while exploring the current ecosystem of databases and form a comparative review of what would be the best architecture suited for blockchain.

Introduction to Blockchain



Advantages and disadvantages of blockchain

Advantages

1. Disintermediation
2. Empowered users
3. Transparency and Immutable
4. Ecosystem simplification
5. Lower transaction cost
6. Faster transaction

Disadvantages

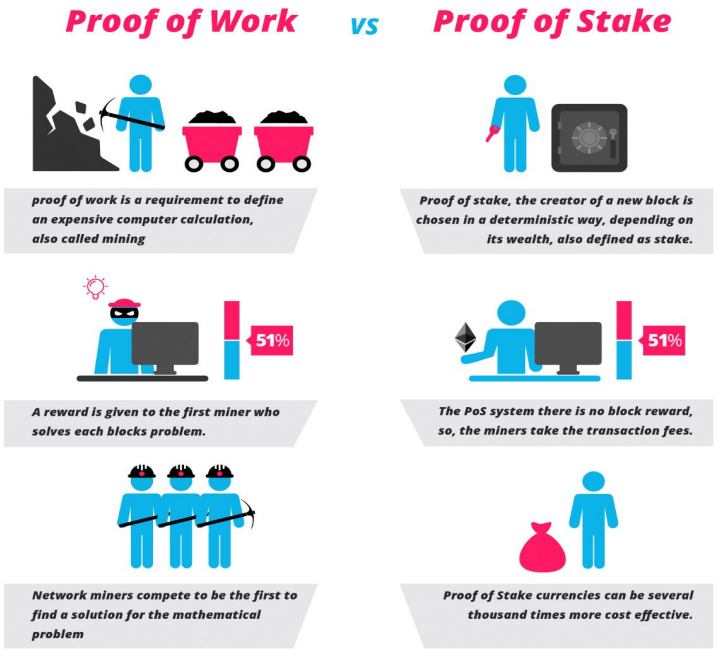
1. Performance
   a. Signature verification
   b. Consensus mechanism
   c. Redundancy
2. Cost
3. High latency

Concept used in Blockchain

Consensus Mechanism- A general agreement between al the nodes.

Can be achieved by -

1. Proof-of-work
2. Proof-of-state



Difference between Relational and NoSQL

Relational Database

1. It does not allow adding data without defining schema
2. It was mainly designed to handle structured data.
3. It is expensive as it does not contain scaling property.
4. Relational database is a closed source with licensing fee

NoSQL

1. It allows to build an application without defining schema
2. It is designed to handle unstructured data
3. It is cheaper to scale a NoSQL database
4. It is a open source hence does not involve huge investment

Blockchain v/s Database

- Regular databases are centralised. Using a client server architecture.
- Blockchain is similar to distributed databases, i.e. with the objective of partitioning larger information retrieval and processing problems into smaller ones.
- Multi-master modifications modifications are made to any copy of data and then sent to other- this is resolved in blockchain.
- A smart contract is the same as stored procedures in centralised databases.
- Blockchains allow permanent, immutable recordkeeping and are much slower than data stores designed to handle and distribute more perishable data.

Relational Database v/s blockchain

- Relational databases have a decisive advantage when it comes to performance.
- Blockchains have a decisive advantage when it comes to providing a robust, fault-tolerant, way to store critical data.
- In a relational database, data can be easily modified or deleted. Single entity controls the data.
- Distributed Ledgers (DL): Leverage cryptography to provide a decentralized multi-version concurrency control mechanism and to maintain consensus about the existence and status of shared facts in trustless environments
- Byzantine-fault tolerant, pseudo-anonymity, auditability (public), immutability, accountability (time-stamping) and non-repudiation (signature) at transaction level.

Blockchain with Relational Database

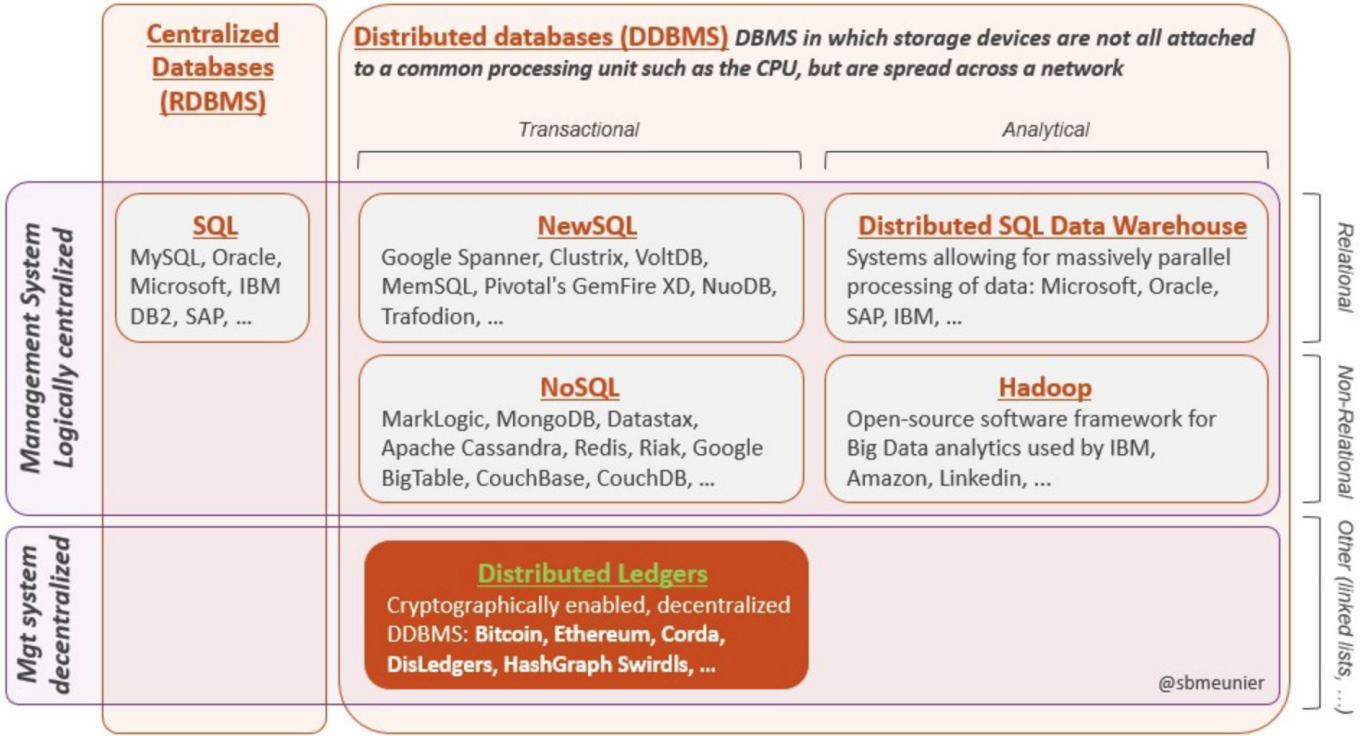
Currently, bitcoin is using Flat file/Level DB to store and retrieve block information.
Bitcoin Architecture = BerkeleyDB (key/value) store + Flat file Database + LevelDB
Block in database is its Index Value (from Level DB) + Its data (stored in serialized format)
Advantages of using SQLite:

- ➔ Memory reduction: Instead of storing and retrieving serialized and unserialized data, a relational database of tables is used to store information.Eliminate the need of leveldb
- ➔ The use of SQLite make that the old system of storage of indices (in leveldb) will be eliminated.
- ➔ The access time of data may be reduced by suppressing the module of serialization/unserialization.

A Comparison

	Blockchain	Relational Database	Blockchain with Relational
Immutability	O		O
No Central Authority	O		O
Assets over Network	O		O
High Throughput		O	O
Low Latency		O	O
High Capacity		O (Expensive)	O (Expensive)
Schema less			
Integrated Caching			

Where does blockchain fit?



Relation between blockchain and NoSQL

Few questions…

- Why NoSQL to blockchain?
  - ➔ Schema-less
  - ➔ Partitioning
  - ➔ Unstructured
  - ➔ Query capabilities

- How can NoSQL match blockchain?
  - ➔ Scalable by partitioning
  - ➔ Low latency
  - ➔ High performance
  - ➔ P2P with nodes permissions (Decentralized)
  - ➔ Immutability by ordering of transactions

Blockchain with NoSQL

	BigChainDB
Description	● Database-style decentralized storage ● Emphasis on scale ● Voting process between nodes provided decentralized control ● Tamper-resistance via ordering of blocks ● Improved performance
Applications	● Database reliability ● Legal contracts and Certificates ● Supply chain ● Intellectual property

A Comparison

	Blockchain	NoSQL	Blockchain with NoSQL
Immutability	O		O
No Central Authority	O		O
Assets over Network	O		O
High Throughput		O	O
Low Latency		O	O
High Capacity		O	O
Schema less		O	O
Integrated Caching		O	O

References

- Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system.
- Peters, G.W. and Panayi, E., 2016. Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. In *Banking Beyond Banks and Money* (pp. 239-278). Springer, Cham.
- Mbinkeu, R.C.N. and Batchakui, B., 2015. Reducing Disk Storage with SQLite into BitCoin Architecture. *International Journal of Recent Contributions from Engineering, Science & IT (iJES)*, *3*(2), pp.10-14.
- McConaghy, Trent, Rodolphe Marques, Andreas Müller, Dimitri De Jonghe, Troy McConaghy, Greg McMullen, Ryan Henderson, Sylvain Bellemare, and Alberto Granzotto. "BigchainDB: a scalable blockchain database." *white paper, BigChainDB* (2016).
- Gaetani, E., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A. and Sassone, V., 2017. Blockchain-based database to ensure data integrity in cloud computing environments.


Questions?