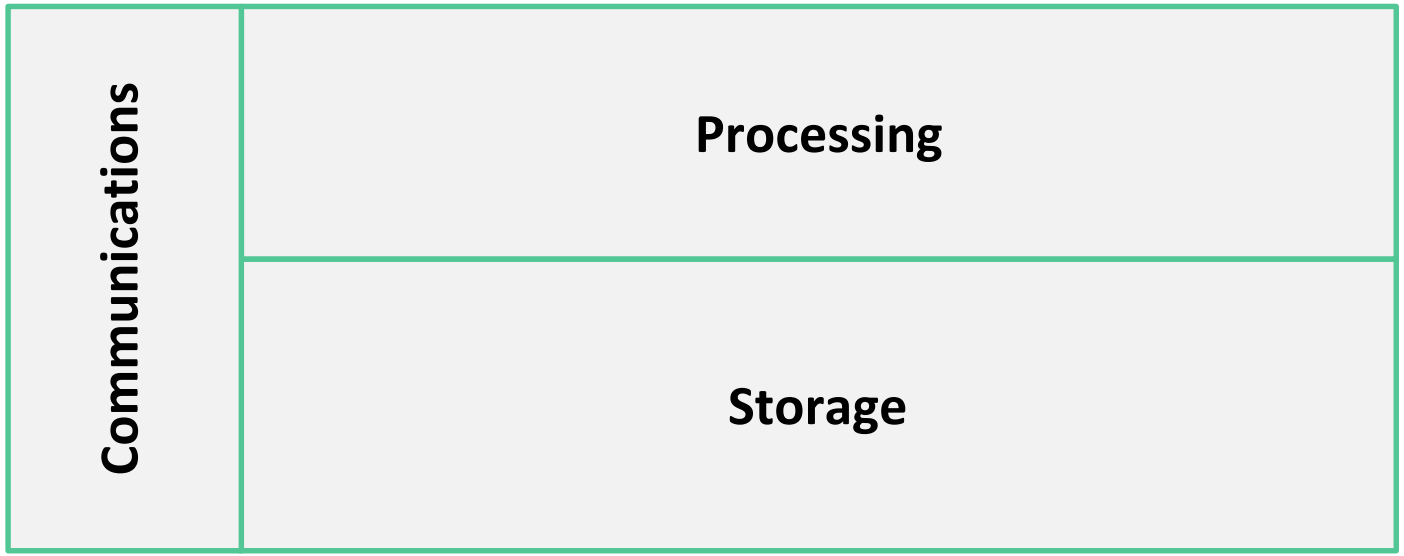
Unlocking Enterprise Blockchain Applications with Big Data Thinking

EFMA, Barcelona, Oct 2016
Trent McConaghy
@trentmc

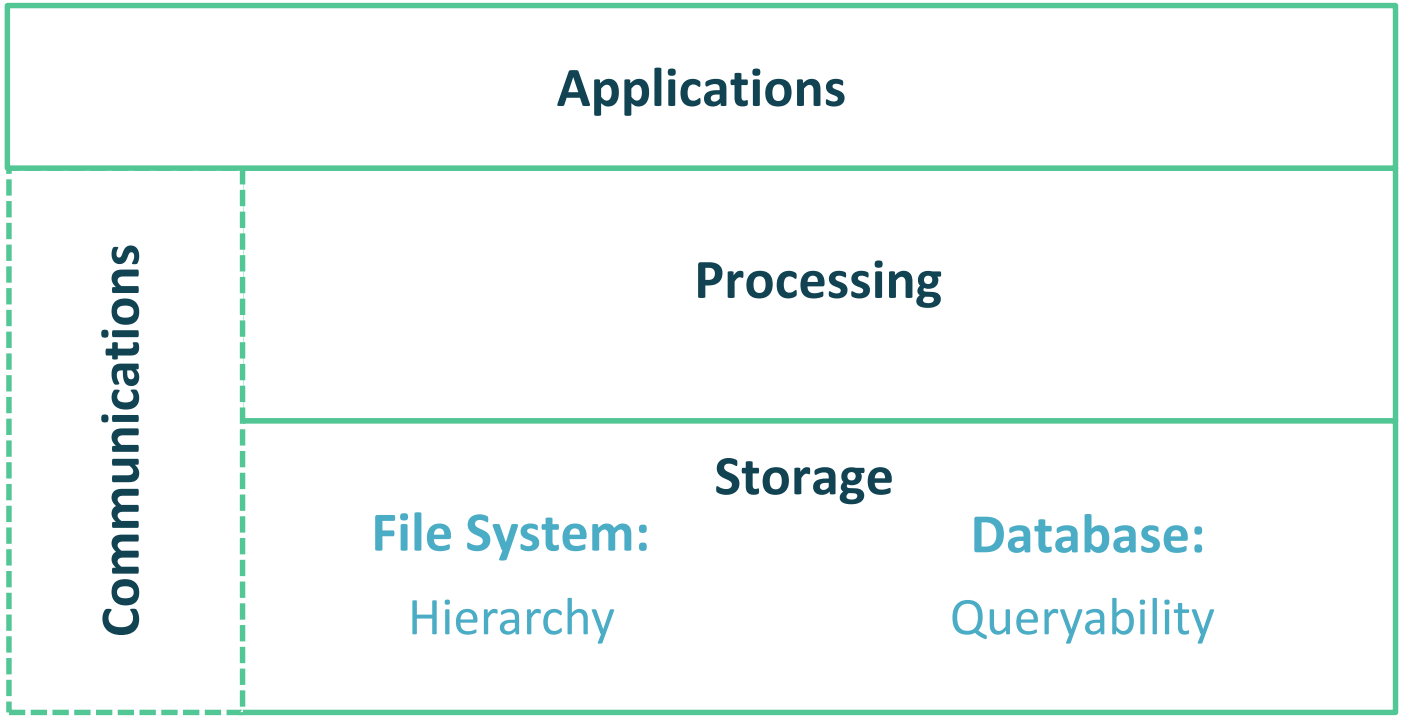
BIGCHAINDB

A scalable blockchain database for people who are changing the world

The Elements of Computing



Modern Application Stacks

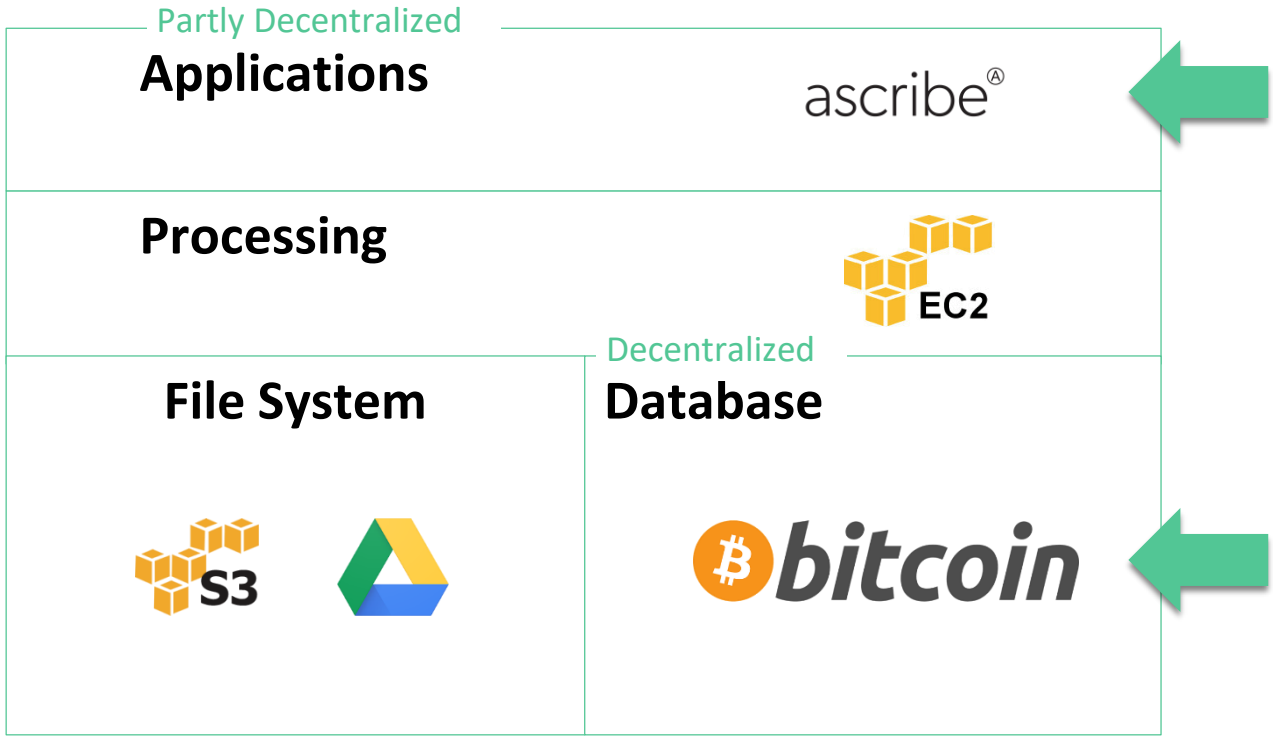


Modern Cloud Application Stack

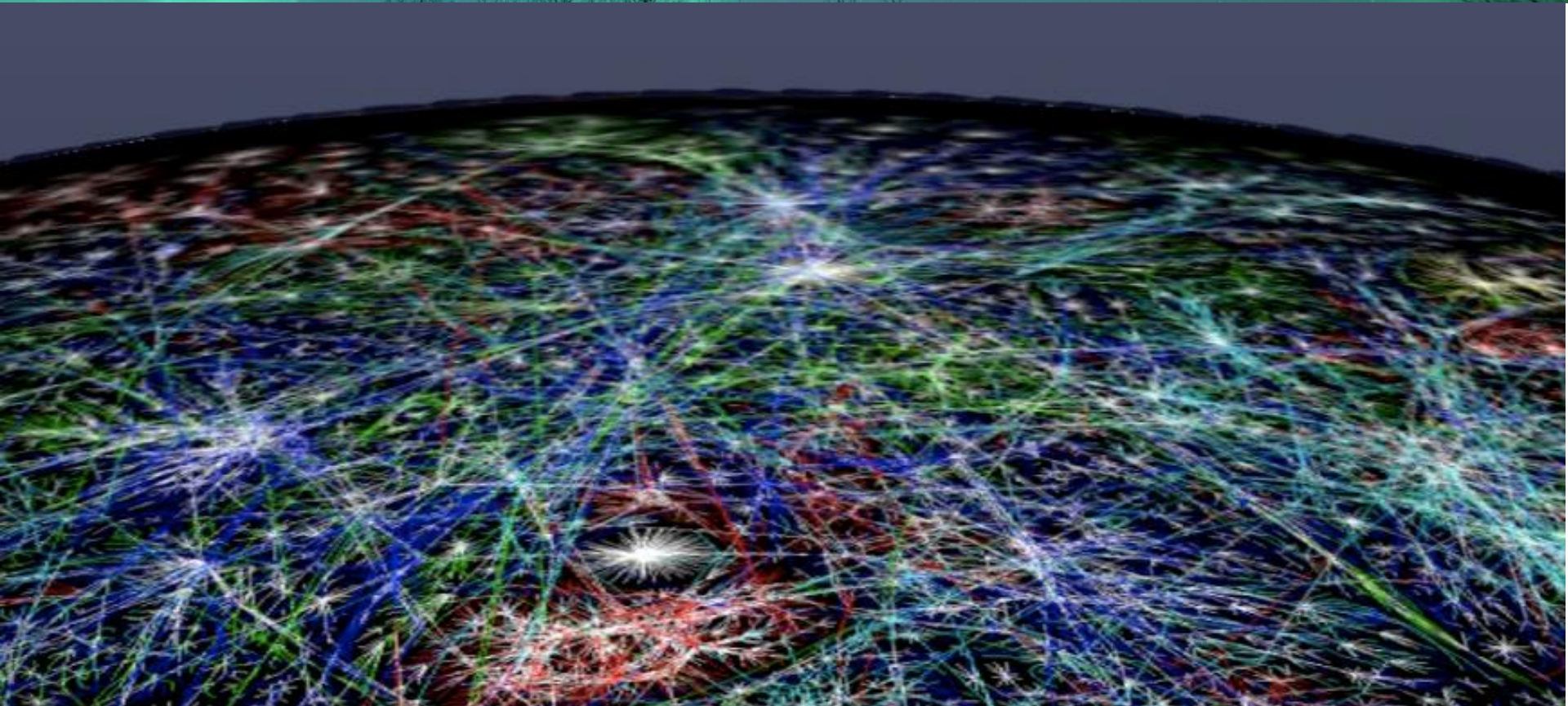
Applications	  
Processing	
File System  	**Database**  

Along Came Bitcoin

Bitcoin sparked a revolution

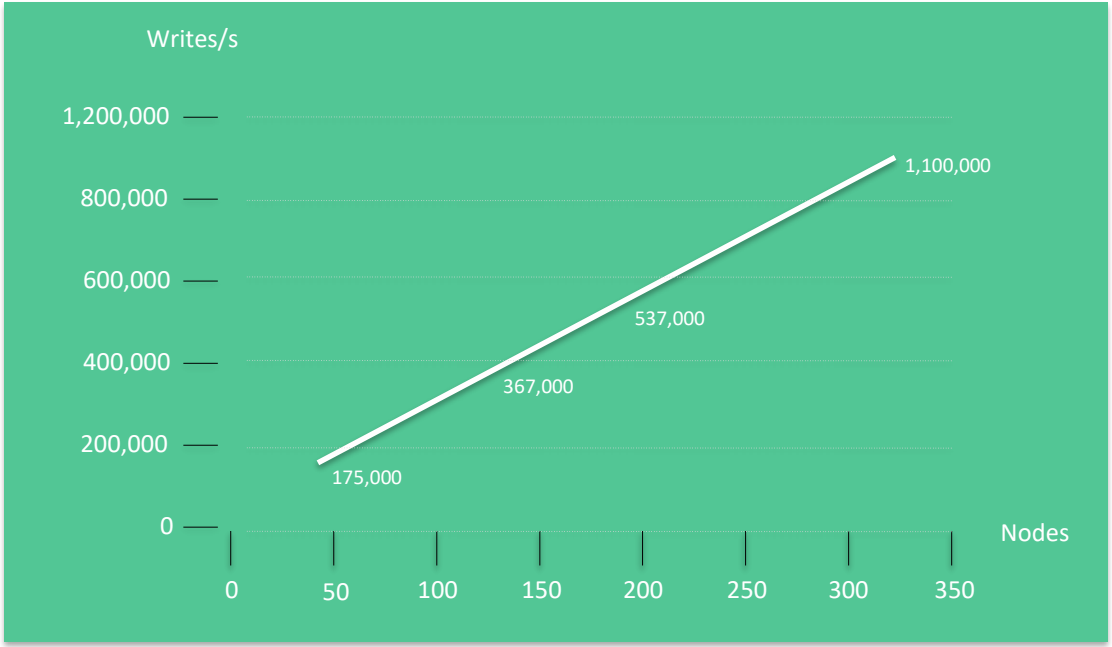


Planetary Scale? Enterprise Scale? (Count the dead POCs…)

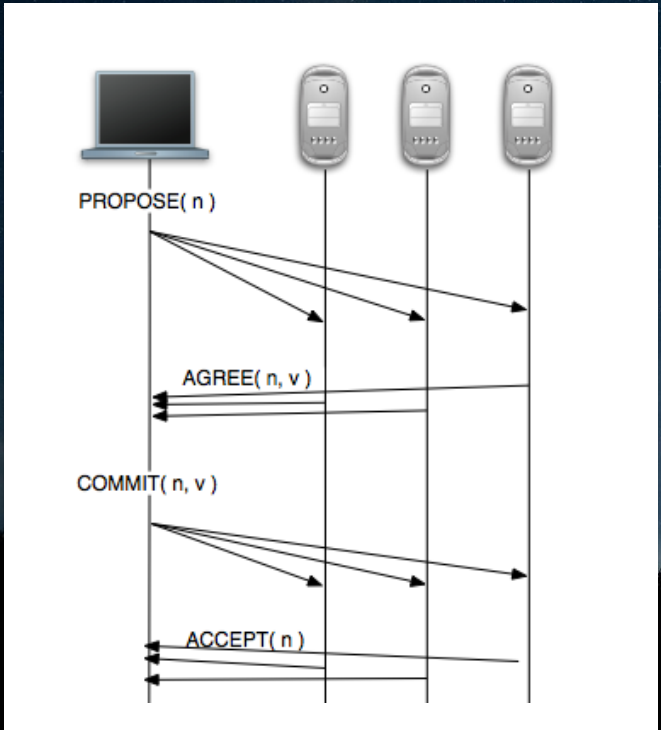


Netflix Uses 37% Of The Internet Bandwidth

Using a modern distributed “big data” database



To be Distributed,
Big Data DBs Must Solve Consensus



Byzantine Consensus (1982)

Paxos (1990/1998)

Practical Byzantine Consensus (1999)

Two Ways to Scale Up

1

Big Data-fy Blockchains

- Build on decades of work
- Significant scalability hurdles

or

2

Blockchain-ify Big Data

- Build on centuries of work
- Scalability challenges already resolved. Sharding etc.

… but how to blockchain-ify?

“Blockchain-ify”

Decentralization: no single entity owns or controls

Immutability: tamper-resistant

Assets: Can issue & transfer assets. Owner based on private key.

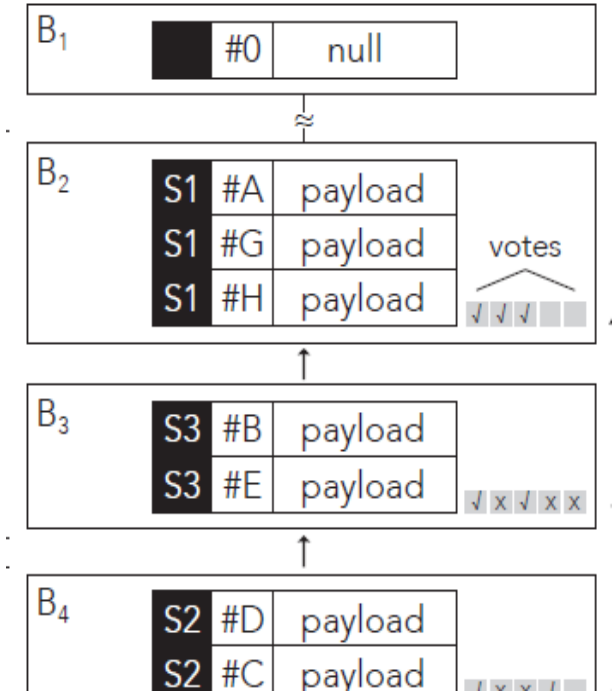
Blockchain (noun): hashed-together chain of blocks (1991!)

Blockchain (noun): storage that is decentralized + immutable + assets

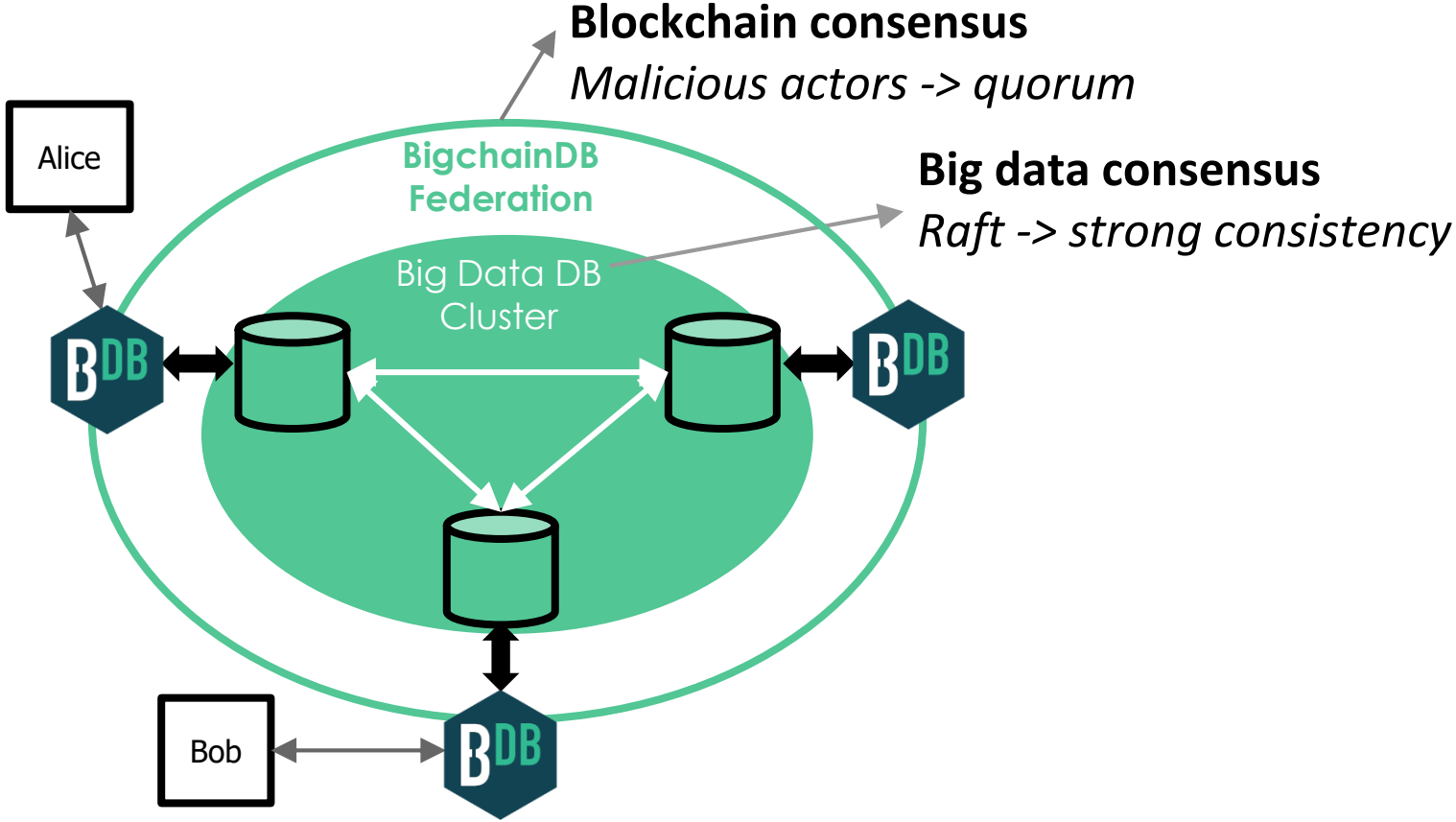
Blockchain (*adj):* decentralized + immutable + assets

How to Blockchainify Big Data

Decentralized	• No single entity in control • Each DB node is a federation node
Immutable	• Strong tamper-resistance • Hash on previous blocks • Append only
Asset Autonomy	• Asset issuance by trusted parties • Control via private/public keys



Architecture – Decentralized Federation



Interface 1/3

Database part : data

Via ReQL (JSON meets SQL)

“Make it look, act, and feel like a database”

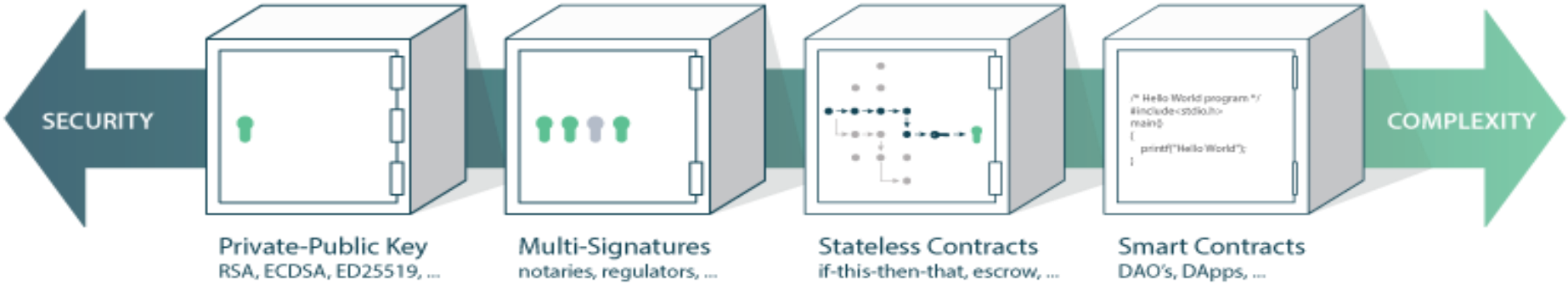
Interface 2/3

Database part : data

Via ReQL (JSON meets SQL)

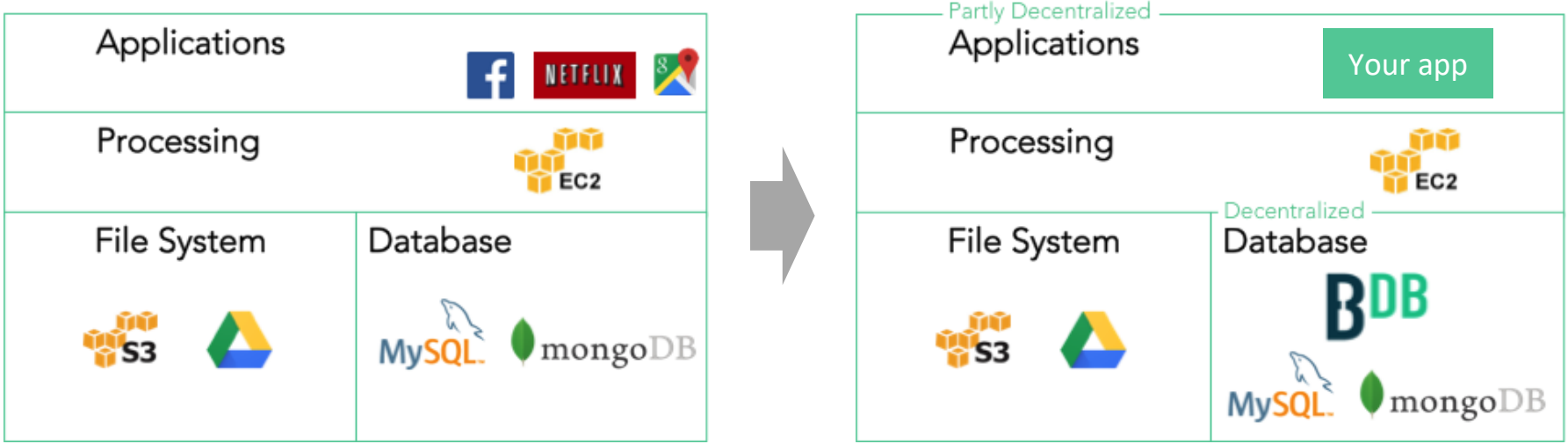
+ Blockchain part : assets, transaction-style

Via Simple Contracts (Conditional circuits rather than sequential circuits)



Interface 3/3

No need to re-do the whole stack,
Just add one more (special) database.



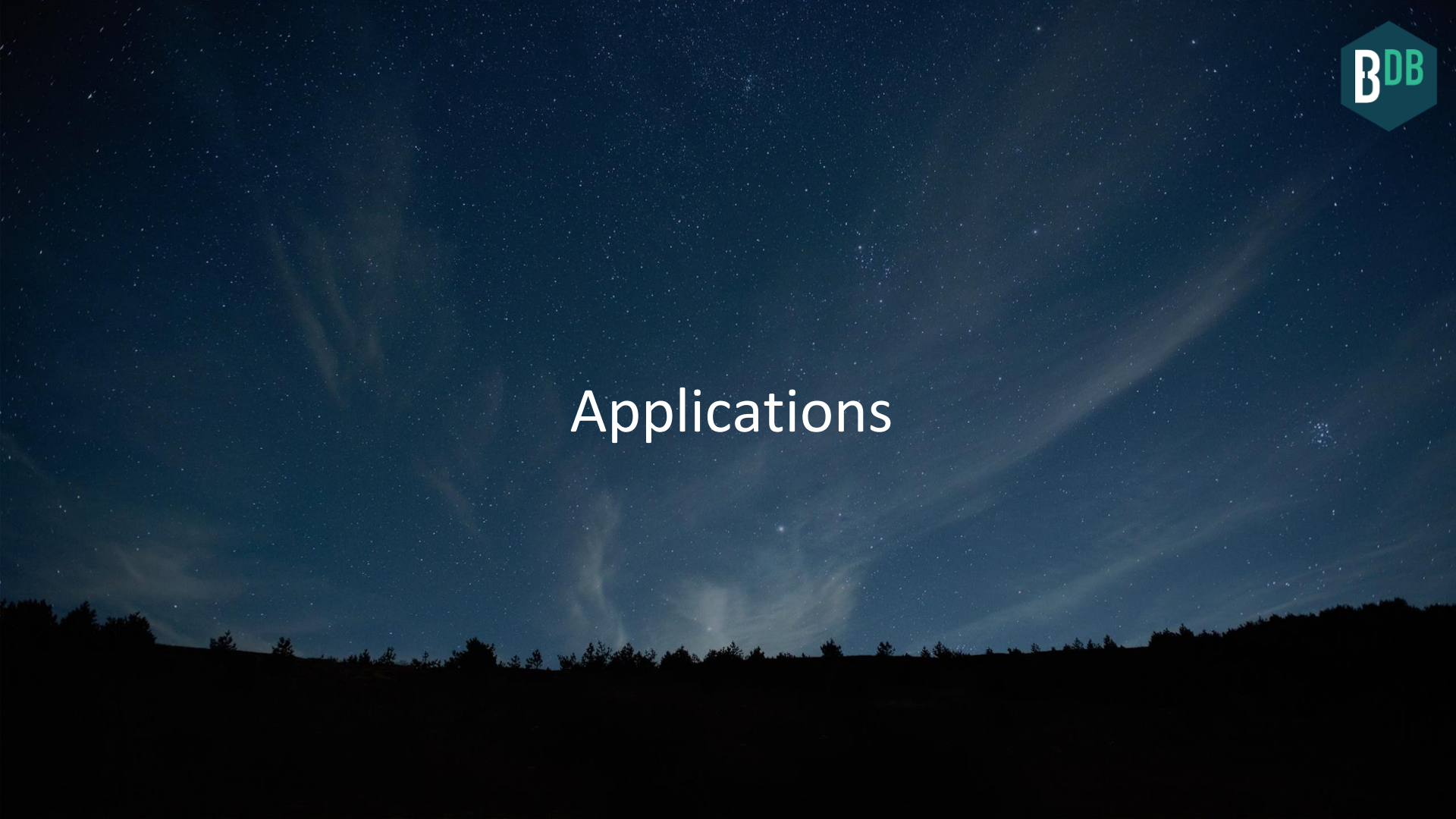
By Re-thinking Blockchain Assumptions…

	Assumption	Our Experience for Actual Needs
Throughput (Transactions per Second)	1,000	100,000+
Security	Byzantine Fault Tolerance, Full nodes	Fault Tolerance ++, Sharding
Query & search	Blockchain explorer	Structured queries
Business Logic	Smart Contracts	Simple Contracts

...We Get the Best of Both Worlds

	Bitcoin	Distributed Databases	BigchainDB
Immutability	✔		✔
Decentralized Control	✔		✔
Native Assets	✔		✔
High Throughput		✔	✔
Low Latency		✔	✔
High Capacity		✔	✔
Access Permissioning		✔	✔
Query & Search		✔	✔

More at: bigchaindb.com/whitepaper


Applications

RWE

Vertical:
Energy

Value proposition:
manage $ flow in energy deregulation

Tangent90

Vertical:
Supply Chain / Health

Value proposition:
government-mandated
transparent $ flow

RECRUIT

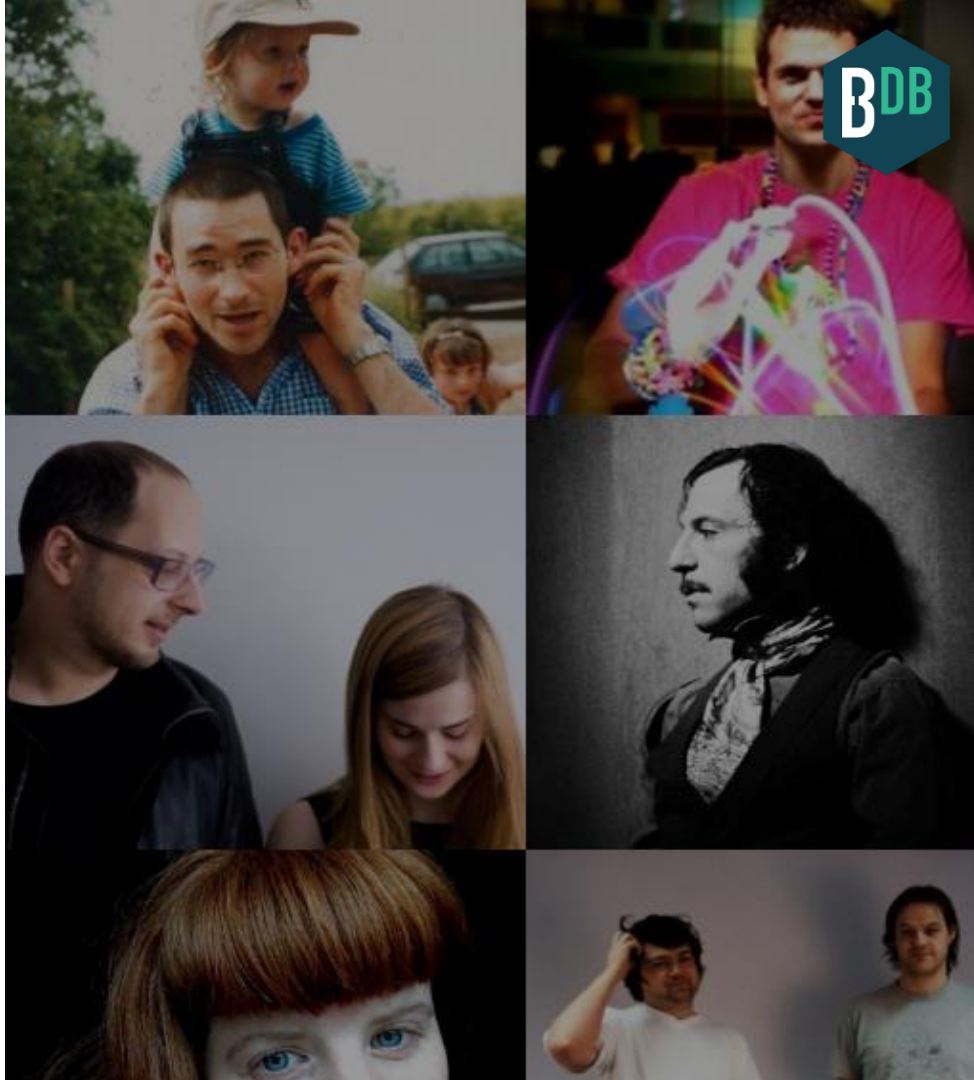
Vertical:
ID - Education Credentials

Value proposition:
reduce fraudulent degrees, lower HR friction

res( )nate

Vertical:
IP – Music rights

Value proposition:
A streaming service owned by all



Interledger

Vertical:
Payment networks & interoperability

Value proposition:
exchange value *across* networks

3 Blockchain Benefits

Decentralized / shared control
Immutability / audit trail
Tokens / exchanges

Generating Opportunities: Vertical x Benefit

	Decentralized / Shared Control	Immutability / Audit trail	Tokens / Exchanges
Intellectual Property			Resonate
Identity	Recruit		**You?**
Finance		**You?**	
Energy	**You?**		RWE
Supply Chain		Tangent90	
Government			

One last use case

Enterprises & financial institutions moving from POCs to scale

How to unlock enterprise blockchain applications?

Merge big data with blockchains

- Big data: Scale + Query
- Blockchain: Decentralized + Immutability + Assets

Trent McConaghy

@trentmc0

BIGCHAIN DB

bigchaindb.com