

BigchainDB: A Scalable Blockchain Database

ascribe GmbH, Berlin, Germany
February 2016

ABSTRACT

This primer describes BigchainDB, a blockchain database capable of 1 million writes per second throughput, storing petabytes of data, with sub-second latency, a proper querying system, and rich permissioning. It supports public and private deployments. Being a decentralized database, it is complementary to decentralized processing technologies like Ethereum Virtual Machine, and decentralized file systems like IPFS. It can be used within decentralized computing platforms like Ethereum-Stratos or Eris-Tendermint. This primer¹ outlines the need for a scalable blockchain database, provides a high-level description of BigchainDB, presents benchmarks, and concludes with use cases.

1 INTRODUCTION

1.1 Towards a Decentralized Application Stack

The introduction of Bitcoin has triggered a new wave of decentralization in computing. Bitcoin illustrated a novel set of benefits: decentralized control, where “no one” owns or controls a network; tamper-resistance, where writes to storage on the network are not easily deleted; and the ability to create & transfer assets on the network, without reliance on a central entity.

The initial excitement around Bitcoin was as a token of value, for example as an alternative to government-issued currencies. As people learned more about the underlying blockchain technology, they extended the scope of the technology itself (e.g. smart contracts), as well as applications (e.g. intellectual property).

With this increase in scope, single monolithic “blockchain” technologies are being re-framed and refactored into building blocks at four levels of the stack:

1. Applications
2. Decentralized computing platforms (“blockchain platforms”)
3. Decentralized storage (ledgers, file systems, databases), decentralized processing (“smart contracts”), and decentralized communication
4. Cryptographic primitives, consensus protocols, and other algorithms

¹ A full whitepaper is coming in early February 2016.

1.2 Blockchains and Databases

We can frame a traditional blockchain as a database (DB), in the sense that it provides a storage mechanism. If we measure the Bitcoin blockchain by traditional DB criteria, it's terrible: throughput is just a few transactions per second (tps), latency before a single confirmed write is 10 minutes, and capacity is a few dozen Gb. Furthermore, adding nodes causes more problems: with a doubling of nodes, network traffic quadruples with no improvement in throughput, latency, or capacity. It also has essentially no querying abilities: a NoQL² database.

In contrast, a modern distributed DB can have throughput exceeding 1 million tps, capacity of Petabytes and beyond, latency of a fraction of a second, and throughput and capacity that increases as nodes get added. Modern DBs also have rich abilities for insertion, queries, and access control in SQL or NoSQL flavors; in fact SQL is an international ANSI and ISO standard.

1.3 The Need for Scale

Decentralized technologies hold great promise to rewire modern financial systems, supply chains, creative industries, and even the Internet itself. But these ambitious goals need scale: the storage technology needs throughput of up to millions of transactions per second (or higher), sub-second latency³, and capacity of petabytes or more. These needs exceed the performance of the Bitcoin blockchain by many orders of magnitude.

2 BIGCHAINDB : BLOCKCHAINS MEET BIG DATA

2.1 High-Level Design

We introduce BigchainDB, which is for database-style decentralized storage: a *blockchain database*. BigchainDB combines the key benefits of distributed DBs and traditional blockchains, with an emphasis on scale, as Table 1 summarizes.

We built BigchainDB on top of an enterprise-grade distributed DB, from which BigchainDB inherits high throughput, high capacity, a full-featured NoSQL query language, efficient querying, and permissioning. Nodes can be added to increase throughput and capacity.

BigchainDB has the traditional blockchain benefits of decentralized control, tamper-resistance, and creation & transfer of assets. The decentralized control is via a federation of nodes with voting permissions, that is, a super-peer P2P network. The voting operates at a layer above the DB's built in consensus. Tamper-resistance is via an ordered sequence of blocks where each block holds an ordered sequence of transactions; and a block's hash is over its transactions and

² We are introducing the term NoQL to describe a database with essentially no query abilities. This term is not to be confused with the company noql (www.noql.com).

³ It takes light 140 ms to make one trip around the world, or 70 ms halfway around. Some financial applications need 30-100 ms latency, though due to speed-of-light constraints those necessarily need to be more locally constrained.

related data, and the previous block’s hash; that is, a block chain. Any entity with asset-issuance permissions can issue an asset; any entity with asset-transfer permissions and the asset’s private key may transfer the asset. This means hackers or compromised system admins cannot arbitrarily change data, and there is no single-point-of-failure risk.

Scalable capacity means that legally binding contracts and certificates may be stored directly on the blockchain DB.




A Comparison	 Bitcoin	 Big Data	
Immutability	✓		✓
No Central Authority	✓		✓
Assets Over Network	✓		✓
High Throughput		✓	✓
Low Latency		✓	✓
High Capacity		✓	✓
Rich Permissioning		✓	✓
Query Capabilities		✓	✓

Table 1: BigchainDB compared to the Bitcoin blockchain and traditional distributed DBs

The permissioning system enables configurations ranging from private enterprise blockchain DBs to open, public blockchain DBs.

2.2 BigchainDB in the Decentralization Ecosystem

BigchainDB can sit side-by-side with other decentralized storage (e.g. IPFS), processing (e.g. Ethereum Virtual Machine, Enigma), and communication building blocks; it can be included in higher-level decentralized computing platforms and applications (e.g. Ethereum-Stratos, Eris-Tendermint) as shown in Figure 1; it will work with centralized computing blocks and platforms too as shown in Figure 2.

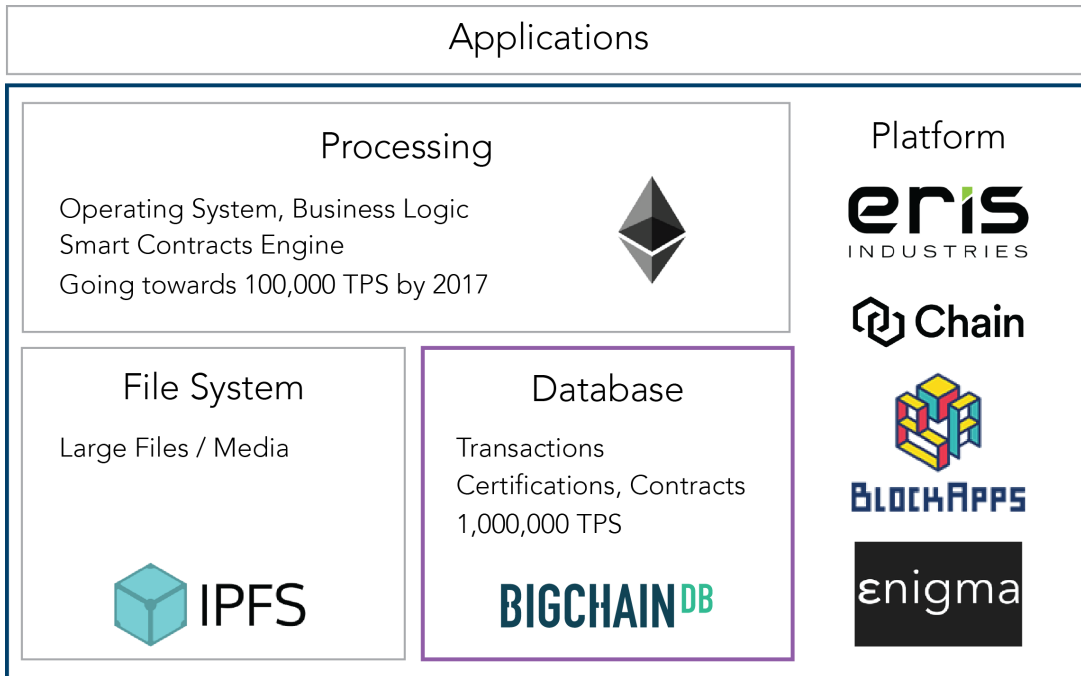


Figure 1: BigchainDB fills a missing gap in the emerging decentralized stack as a blockchain database that complements existing platforms, processing (business logic) and file systems.

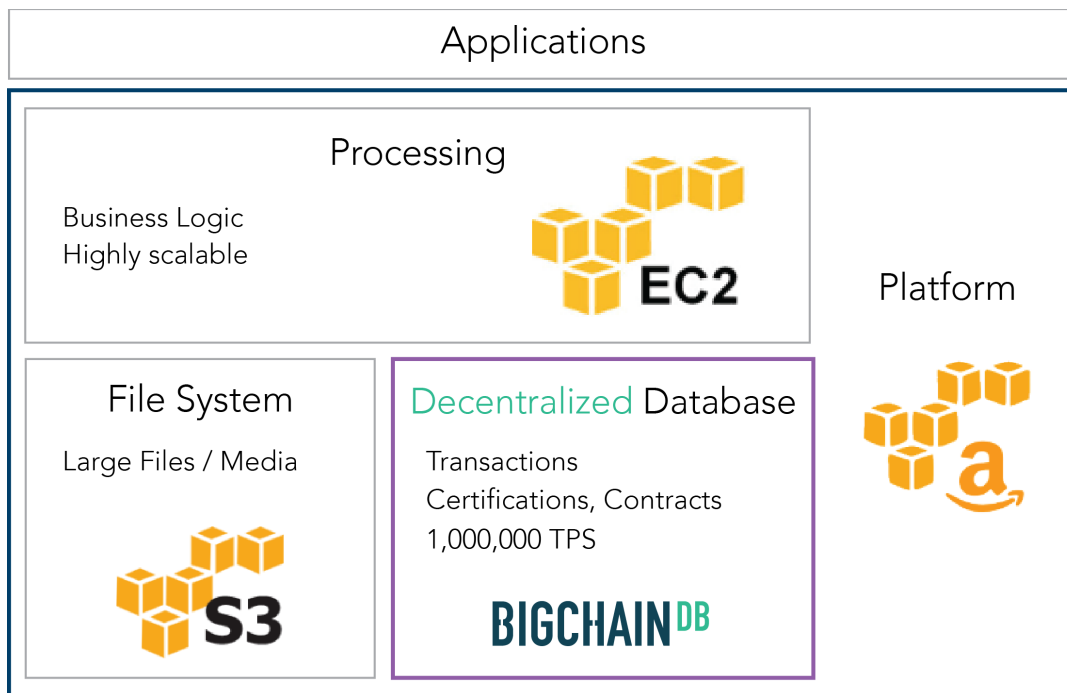


Figure 2: BigchainDB can be seamlessly integrated into the traditional stack as a blockchain database for decentralized time-stamping, certificates, smart contracts and transactions.

3 EXPERIMENTAL RESULTS

In our preliminary experiments with the running BigchainDB end-to-end, we found that the biggest limiter of performance was in how the datastore itself interacted with the physical compute resources (write speed, IO among nodes). This was not surprising, because BigchainDB's design is about "getting out of the way" of what the datastore itself is good at. Therefore, the experiments shown here focus on the datastore performance.

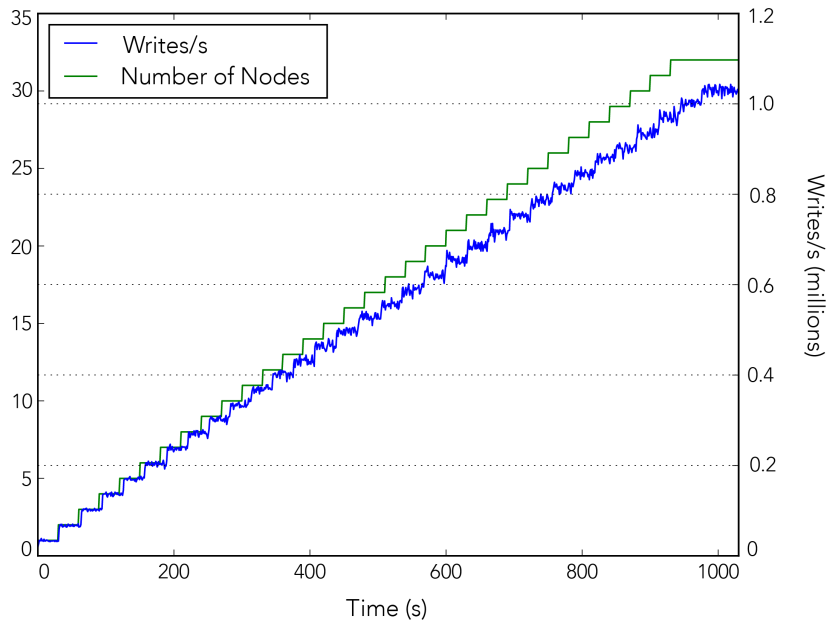


Figure 3: Time-series plot of BigchainDB performance. As we increased the number of nodes, the throughput in terms of writes/s increased accordingly

In one experiment, we increased the number of nodes every ten seconds, up to 32 nodes. Figure shows how write throughput increased every time a node was added. When the number of nodes reached 32, the write throughput was just over 1 million writes per second (i.e. 1,000 blocks per second, with 1,000 transactions per block).

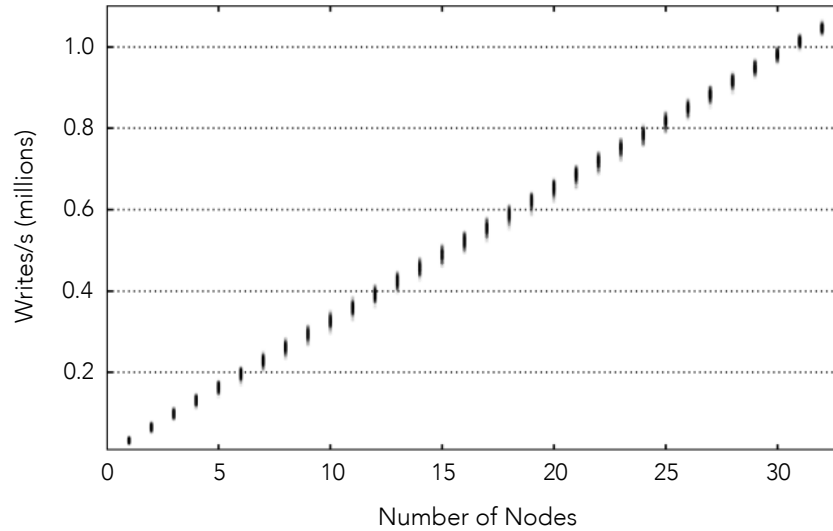


Figure 4: Writes/s versus number of nodes in BigchainDB. There is linear scaling in write performance with the number of nodes. 32 nodes gives >1,000,000 writes/s.

Figure 4 shows data from the same experiment, except it shows how BigchainDB write throughput is a function of the number of nodes, rather than time. The plot is both boring and exciting: it shows how write throughput increases linearly with the number of nodes. 32 nodes gives performance of >1,000,000 writes/s. By comparison, the Bitcoin network typically has 1-1.5 writes/s; a theoretical maximum of 7; and throughput stays flat as the number of nodes increases.

BigchainDB uses fractional replication, where each node holds some of the data, rather than full replication, where each node holds all of the data. This is essential to scalability. In these experiments, each node 48 TB of storage. In our system there are 32 nodes, so total capacity is 1536 TB, or 1.54 PB storage. A replication factor of 3 gives about 0.5 PB storage.

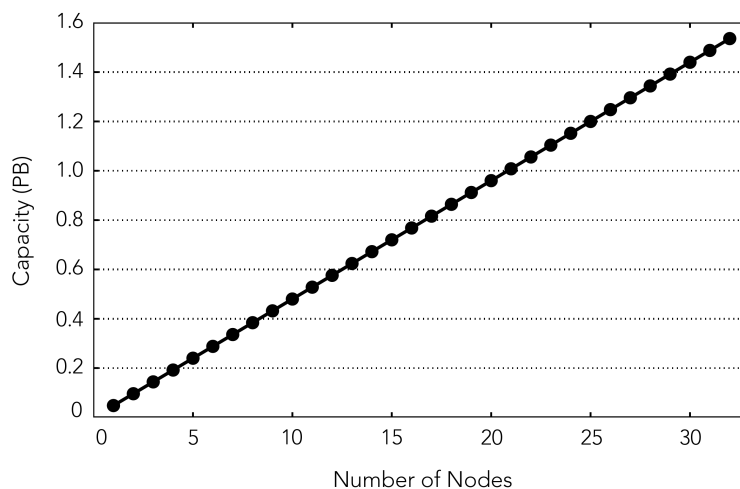


Figure 5: Each node adds another 48 TB of total storage capacity to BigchainDB.

Figure 5 shows how total BigchainDB capacity increases with the number of nodes. By comparison, the Bitcoin network currently holds about 50 GB; and capacity stays flat as the number of nodes increases.

4 BIGCHAINDB USE CASES

Many BigchainDB use cases are like traditional blockchain use cases, except focused on situations where higher throughput, more capacity, lower latency, better querying, or richer permissioning is necessary. For example, BigchainDB can handle the throughput of high-volume payment processors, and directly store contracts receipts, or other related documents on the DB alongside the actual transaction.

Some BigchainDB use cases are also like traditional distributed DB use cases, except focused where blockchain characteristics can benefit. For example, improving DB reliability by not having a single point of failure, or storage of documents with secure time-stamping.

BigchainDB use cases include:

- **Legally-binding contracts** can be stored directly on the BigchainDB next to the transaction, in a format that is readable by humans and computers.
- **Creation and real-time movement of high-volume assets.** Only the owner of the asset can move the asset, rather than the network administrator like in previous database systems. This capability reduces costs, minimizes transaction latency, and enables new applications.
- **Tracking high-volume physical assets** along whole supply chain. BigchainDB can help reduce fraud, providing massive cost savings. Every RFID tag in existence could be entered on a BigchainDB.
- **Tracking intellectual property assets** along the licensing chain. BigchainDB can reduce licensing friction in channels connecting creators to audiences, and gives perfect provenance to digital artifacts. A typical music service has 38 million songs—BigchainDB could store this information in a heartbeat, along with licensing information about each song and information about use by subscribers.
- **Time stamping, receipts, and certification.** BigchainDB reduces legal friction by providing irrefutable evidence of an electronic action. And, BigchainDB is big enough that supporting information like receipts and certificates of authenticity (COAs) can be stored *directly* on it, rather than linking to the document or storing a hash.
- **Improving database reliability** by creating resistance to single points of failure. This reliability helps move past the status quo where a single hack leads to massive data loss, like in Target, Sony, or the OPM.

5 BIGCHAINDB PRODUCTS & SERVICES

We envision the following products and services surrounding BigchainDB.

1. **BigchainDB: a blockchain database** with high throughput, high capacity, rich permissioning, query capabilities and low latency.
 - For industry consortia creating new private blockchains, to take advantage of blockchain capabilities at scale.
 - BigchainDB will be available in an out-of-the-box version that can be deployed just like any other DB, or customized versions (via services, or customized directly by the user).
 - BigchainDB will include interfaces such as a REST API, language-specific bindings (e.g. for Python), RPC (like bitcoind), and command line. Below that will be an out-of-the-box core protocol, out-of-the-box asset overlay protocol, and customizable overlay protocols.
 - BigchainDB will support legally binding contracts, which are generated automatically and stored directly, in a format readable by both humans and computers. There will be out-of-box contracts for out-of-the-box protocols, and customizable contracts for customizable protocols.
 - BigchainDB will offer cryptographic Certificates of Authenticity, which can be generated automatically and stored directly on the BigchainDB. There will be out-of-box and customizable versions.
 - BigchainDB is built on a large, open-source pre-existing database codebase that has been hardened on enterprise usage over many years. New code will be security-audited and open source.
2. **BigchainDB as a Service**, using a public BigchainDB instance, or a private BigchainDB with more flexible permissioning.
 - For developers who want the benefits of blockchains without the hassle of setting up private networks.
 - For cloud providers who want scalable blockchain as part of their service.
 - Main interfaces will be a REST API directly, REST API through cloud providers, and language-specific bindings (e.g. Python).

Stay tuned for our full announcement on February 10 at The Block Chain Conference in San Francisco (<http://www.theblockchainconference.com/>).

The whitepaper will be available on February 10. The living document will incorporate community feedback and can be accessed at www.BigchainDB.com.