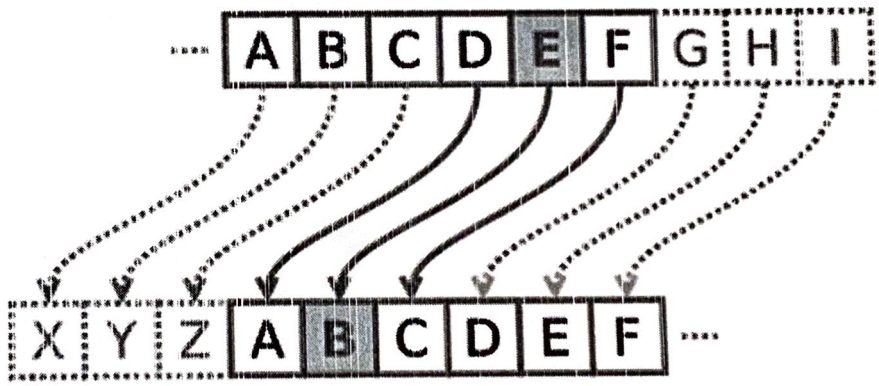
I. Cryptoeconomics

Economic principles help us to design a system so that actors are incentivized to make decisions in line with the goals of the greater good. We are able to **secure the future.** (e.g. block reward in Bitcoin and cost of mining to deter Sybil attacks)
Cryptography allows us to **secure the past** and ensure our decisions cannot be manipulated by observers (e.g. cryptographic signatures for authentication and hashes for immutability)

II. Cryptography

Cryptography aims to secure the integrity and confidentiality of information.

The need for cryptography is especially important in distributed systems, where unknown actors are a potential threat to the secrecy and safekeeping of information.
Encryption is the process of transforming information into an unintelligible intermediary piece of information which can be transformed back into its original state with **decryption**. An early example of encryption was the Roman Empire's use of the **Caesar Cipher**, in which messages are encrypted by shifting letters to the right by a previously set amount.



Be aware of various cryptographic primitives (review from previous course):

- **Cryptographic hash functions**, used to capture the identity of information without revealing anything about the information itself
- **Digital signatures**, used to prove your identity and that you sent a particular message
- **Erasure codes** lower the 100% data availability requirement
- **Timelocks** allow for a message to be easily encrypted but take a longer amount of time to decrypt.

III. Economics

Economics boils down to a fundamental question: how do you determine the best choice to make with your limited resources in order to maximize your profit? Economics also helps us to design a system so that everyone is incentivized to act in a certain way.

In **game theory,** we aim to deduce how an actor will act in a given situation. These decisions are influenced by the actions of others and the rewards and penalties associated with certain decisions. Therefore we aim to manipulate these factors.

In blockchain, **tokens** are used as economic incentives. Tokens are units of protocol defined cryptocurrency given out to miners and privileges miners can charge for. The assumption here is that the underlying objective for actors in a blockchain network is to maximize their profit, which equals their revenues minus their costs.

IV. Proof of Stake

Proof-of-Stake is a particular type of consensus mechanism that assumes all voting power is tied to financial resources. Fundamentally, the idea is: the more tokens or currency an actor holds within a Proof-of-Stake system, the stronger the incentive for them to be good steward of said system; if the system grows the wealthier the actor becomes. Thus in Proof-of-Stake, we give these individuals the most power as validators.

Major PoS implementations:

- **Tendermint** - First BFT-based PoS consensus mechanism, published in 2014
- **Casper the Friendly GHOST (CBC)** - a family of consensus algorithms designed from ground up i.e. Correct-by-construction, a proposed upgrade for the Ethereum network
- **Casper the Friendly Finality Gadget** - a Proof-of-Work and Proof-of-Stake hybrid; another upgrade proposed for the Ethereum network

V. Attacks

Each proof of stake attack represents a scenario in which the incentives of an individual are no aligned with the incentives of a group, i.e. giving an unfair advantage to any single actor. Because the resource consumed is monetary value, bad actors need to receive an explicit monetary penalty with each attempted attack to keep the system in check.

If there was zero penalty, the expected profit of any given attack would be some number greater than zero, providing an incentive. By penalizing users for incorrect or malicious action the system hopes to bring the expected value to less than or equal to zero.

Examples of attacks:

- **Nothing-at-Stake**: voting in favor of every fork in hopes of maximizing one's rewards i.e. guaranteeing you will not miss the reward from the chosen branch; solution: slashing an actor if they are caught voting on multiple forks, or a less popular scheme penalizes incorrect votes; keep in mind that voting takes place using cryptographically identifiable/verifiable signatures.

- **Stake grinding**: attack where a validator performs some computation or takes some other step to try to bias the randomness in their own favor; solution: require validators to deposit their coins well in advance, and avoid information that can be easily manipulated as source data for randomness

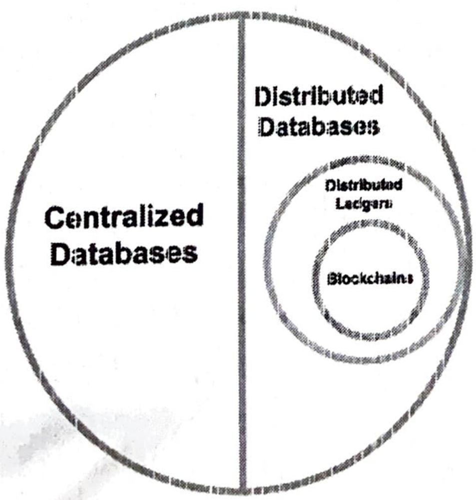
Weak subjectivity is a problem for new nodes or nodes that have been offline for a long time; the node does not know which chain is the main chain; solution: introduce a "revert limit" - a rule that nodes must simply refuse to revert further back in time than the deposit length

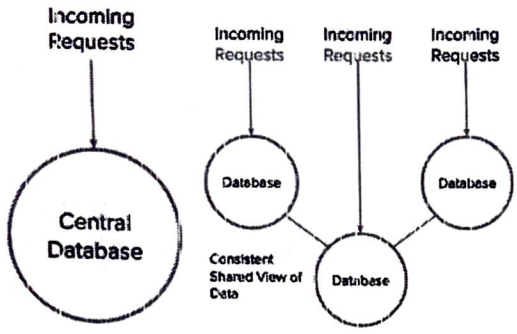
I. Enterprise Blockchain Overview

As Bitcoin and blockchain technology matured, banks and corporations took interest in developing what are now known as **permissioned**blockchains and distributed ledgers. They aimed to "take the blockchain out of Bitcoin."

Permissioned systems only allow trusted users into the system, allowing for a reduction of k properties pushed by public blockchains, resulting in systems with reduced levels of openne no guarantee of trustlessness, and fewer incentives built into the protocol.

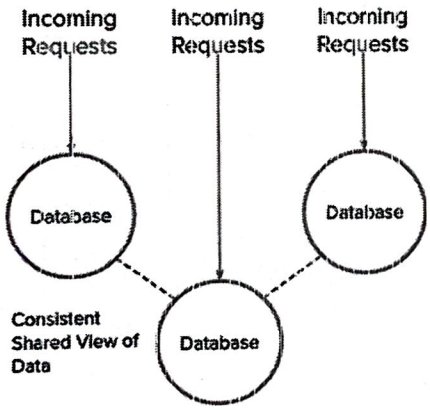
Primarily, enterprise blockchains of the time were used to solve issues in **coordination failu** boost **horizontal integration**, and create **self-sovereign** decentralized networks.





Distributed databases are run by a group of storage nodes that are connected to each other and work to maintain a consistent overall view of the entire system. Nodes are able to fully trust each other in some systems (hence the solid lines connecting storage nodes.)

Distributed ledgers are a specific type of distributed database in which the information is organized chronologically, mimicking a traditional ledger. Most often, storage nodes may not fully trust each other (hence the dotted lines in the diagram below). Instead, they must implement some form of consensus protocol to have a consistent view of the system.



Distributed ledgers that specifically implement a chain of blocks in their protocol are known as **blockchains**.

Blockchains exist in three broad categories, depending on their access types: public, consortium, and private blockchains. Together, consortium and private blockchains are known as permissioned ledgers, since they require some level of permission granted – as opposed to openly readable and writable public blockchains.

II. Enterprise Blockchain Platforms

There exist many enterprise blockchain platforms today – too many to mention in detail in this summary. The key things to look for when evaluating whether a particular enterprise blockchain platforms is right for a particular use case are:

1. Enterprise blockchain platforms usually specialize in particular use cases, or have been used in the past to address certain use cases
2. As they specialize in particular use cases, they make usage assumptions that affect overall system scalability, security, and decentralization
3. These properties are affected by the underlying consensus mechanism(s) an enterprise blockchain platform supports

III. Use Cases & Industries

Enterprise blockchains are being used today in a number of different use cases, including: auto/mobility, finance, travel/tourism, digital identity, and supply chain.

In general, the essential properties of a good blockchain use case are that:

1. Blockchain is not only viable, but is **necessary**. Otherwise, it's hard to justify a blockchain's low "efficiency"
2. Blockchain is used to **solve coordination failures**. Blockchain could be used to create arbitrary incentive structures and enable the cooperation of an untrusting consortium of companies and entities.
3. Blockchain aids in **horizontal integration**. Since data is now stored in a logically centralized blockchain, we can combine data silos and enforce a common API and data standard.
4. Blockchain achieves **pure decentralization**. This is not as relevant to enterprise blockchains, but blockchain in general (public ones) can be used to avoid centralized corruption.

Always keep in mind the advantages of centralized database solutions, and think of whether they, or a subset of blockchain technology, could be used to solve your business need – rather than an entire blockchain.

IV. ICO Schemas & Culture

An **initial coin offering (ICO)** is a novel, "unregulated" means of raising funds for a blockchain startup.

Longer Propagation Time

However, as with decreasing block time, there are some side effects. For one, increasing block size would imply hard forking, and depending on the community, this could be a less than pleasant experience. It would also make the blockchain grow in size at a much faster rate – a problem decreasing the block time also faced. And finally, increasing the block size is most likely not a one-time fix, since the scalability boost is only linear. The block size might need to be increased in the future again, leading to a "slippery slope" type of debate.

II.III. Decrease Transaction Size

Soft fork

Segregated Witness (SegWit) was an upgrade to Bitcoin that move transaction signatures from within the transaction to a separate structure at the end of the transaction, called the segregated witness. To non-SegWit nodes, this would be a decrease the effective transaction size since they wouldn't know to read into the segregated witness.

Non-SegWit nodes would see a transaction without a signature, but would mark the transaction as valid. SegWit nodes on the other hand would know to read into the segregated witness, and would verify it using the signature.

SegWit was originally designed to solve transaction malleability in Bitcoin. It also is implemented with a soft fork, and results in a smaller blockchain size. However, SegWit is only one time linear scalability boost.

BTC : Blocksize 1MB / SegWit Yes
BTC Cash Blocksize 8MB SegWit No

Recursive SNARKs also decrease transaction size. Instead of storing transactions themselves in the blockchain, we could instead store proofs that these transactions have indeed occurred, and the final balance sheet of who owns however much cryptocurrency. This leads to efficiency gains by decreasing transaction size, and also because machines can verify proofs within milliseconds. However, currently, a trusted environment setup is required in order to produce these style of proofs. And proof generation in practice is very costly.

III. Vertical Scaling Off-Chain

Given that the speed of a blockchain limits its scalability, we can consider entirely removing the more costly operations off the chain and only publishing when we require a global sense of truth.

Payment channels in Bitcoin could be implemented using HTLCs (hash time lock contracts), and could move transactions off the main Bitcoin blockchain and onto side chains. If Alice and Bob transact often, perhaps it makes sense for Alice and Bob to construct a private payment channel, where they conduct their transactions off-chain. Only when they want to settle their final balances do they post back to the main blockchain. This allows Alice and Bob to still conduct their transactions as they do, but the main blockchain only has to store Alice and Bob's initial and end balances.

Bitcoin Size: 180 GB Aug 2018
226 GB Q1 2019

II. Vertical Scaling

$$\frac{1\ MiB}{1\ block} \times \frac{1\ txn}{546\ bytes} \times \frac{1\ block}{10\ min} \approx 3.2\ tps$$

block size → 1 MiB / 1 block
546 bytes per transaction
3.2 tps ← transaction / second

Bitcoin processes less than 10 transactions per second, and without any scalability upgrades, it's bound to stay at low TPS. Looking at how we calculate TPS in the first place, namely in the rough dimensional analysis above, we can see that the fields we can attempt to modify in efforts to create new scaling solutions are:

scaling — Vertical: more RAM/CPU power
Horizontal: more machines
Diagonal: more powerful machines

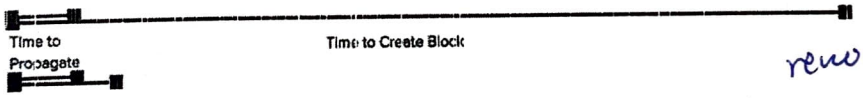
1. Block time
2. Block size
3. Transaction size

These parameters are all built into a blockchain system itself, and tuning these parameters directly constitute as layer 1 scaling solutions.

II.I. Decrease Block Time

We can't simply decrease the block time of a blockchain system, since that would result in a higher rate of naturally occurring forks, reducing system security. This is because while block time decreases, the time to propagate a block remains the same.

Time to broadcast block fixed while Block creation time decreases



reward uncle blocks

Ethereum has dealt with this problem historically by employing the GHOST (Greedy Heaviest Observed SubTree) protocol. With the GHOST protocol, miners no agree on the longest chain to be canon (as in Bitcoin), but rather the chain with the most "weight", where weight is a value calculated by both a chain's length and the number of uncle blocks it has.

II.II. Increase Block Size

Increasing block size would improve a blockchain's TPS. Since a block can now contain more transactions, it would also lower transaction fees.

ICOs are meant to allow developers to monetize open-source software despite the traditional incentive to make software proprietary. Additionally, it gives blockchain projects a much larger source of investors than only a relatively smaller set of VCs and other accredited investors.

However, ICOs also come with caveats. Because of a lack of regulation, scams are more capable of making their way into the view of investors, less doable when all investments were first screened either by VCs or government bodies, forcing investors to do more of their own due diligence. Additionally, many ICOs raise so much money that they have no incentive to actually finish up the project, leading to incentive misalignments.

V. Regulations & Caveats

As the world has never seen anything like blockchain before, there are still few regulations to specifically handle cryptocurrency and blockchain related matters.

First, because cryptocurrencies are inherently deregulated, they not only fail to abide by, but also may attempt to circumvent, laws such as anti money laundering (AML) laws and know your customer (KYC) regulations, leading to conflicts between regulatory bodies and cryptocurrency projects and exchanges. Exchanges are required to acquire licenses, such as a money transmitter license or a New York BitLicense in order to provide services. Some governments have taken steps towards regulating cryptocurrencies and blockchain, for better or for worse. Vermont and Arizona have declared that portions of the information on a blockchain can be considered legal evidence in court, but some countries have taken steps to restricting access to cryptocurrencies.

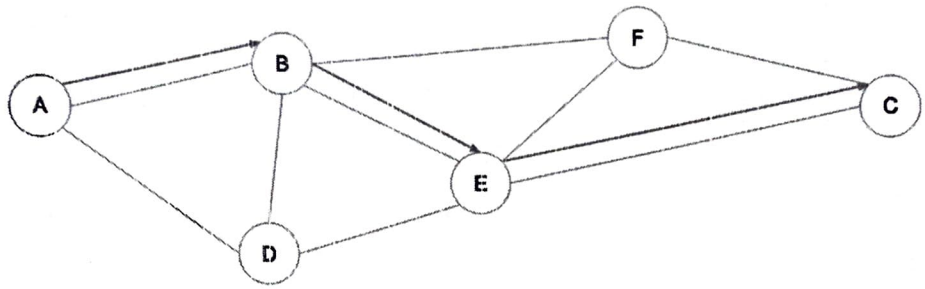
I. Background

Modern day public blockchains have been victims of their own success. Bitcoin and Ethereum especially are having scalability issues in that they aim to be global networks able to support global-scale transaction volumes, but currently both perform subpar in the transaction throughput.

Fundamentally, scaling solutions can either increase the transaction volume, or decrease the block time. This is self evident as scalability is measured in a blockchain's achievable TPS (transactions per second.)

Going forward, we can classify blockchain scaling solutions two ways. The first is a rough comparison with traditional cloud architecture scaling classifications: horizontal, vertical, and diagonal. Secondly, there are the blockchain-specific scaling classifications: layer 1 (on-chain) and layer 2 (off-chain).

The idea behind the Bitcoin Lightning Network is to create a network of payment channels



In the diagram above, Alice can pay Charlie without having a payment channel to Charlie directly, so long as there is a path from Alice to Charlie through the payment channel network.

Ethereum has a similar scalability solution in the works, appropriately named Raiden.

Payment channels and payment channel networks would allow us to keep many transactions off chain, delegating payments to simple bookkeeping. Since the main blockchain only sees the start and end balances of the parties in a payment channel, we can keep a majority of transactions off chain: scaling Bitcoin from under ten transactions to potentially hundreds of thousands of transactions.

Issues

Some problems include having to lock up capital in order to initiate a payment channel, and centralization concerns of payment channel networks converging to hub-and-spoke topologies.

IV. Horizontal Scaling

Node Categories { super-full, top-level, single-shard, light

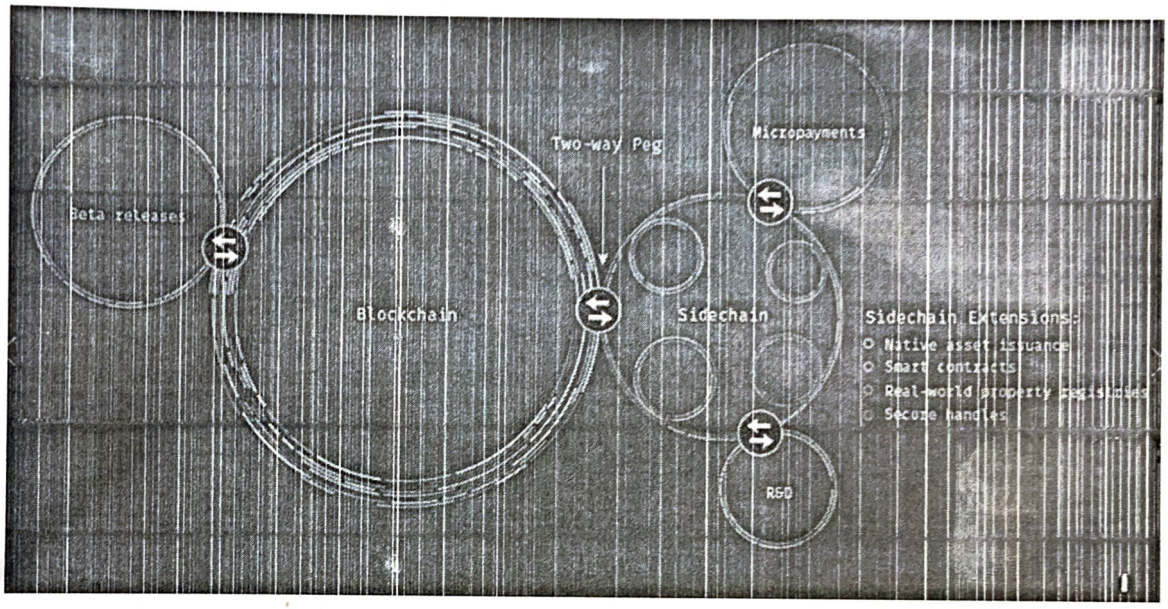
Sharding is database scaling strategy that breaks up a monolithic database into "shards", each a separate database that contains data from a subset of the original database, whose union is the original database. The same idea can be applied to blockchain, and is currently one of the active areas of research in Ethereum research.

The idea translated to blockchain implies that not every node keeps track of every block. It would be a layer 1 horizontal scaling solution. We could have multiple blockchains running in parallel, each containing a subset of all transactions. Issues currently being researched include the classification of various nodes in a sharded blockchain system (e.g. nodes that see a single shard vs nodes that see all shards), cross-shard communication, and defenses against single shard takeovers.

Sidechains are the idea that you can create multiple side chains for different purposes that plug into a main chain, effectively decreasing the traffic on it.

This does separate hashing power across multiple chains, which raises security concerns.

Here is an example of a sidechain setup:



Source: https://blockstream.com/technology/ Opens in new window

V. Advanced Scaling & Generalizations

Ethereum's Plasma can be seen as a diagonal scaling solution, since it enables horizontal scaling by implementing side chains and vertical scaling by increasing their speed through Tendermint and alternative consensus mechanisms. The security of off-chain transactions is derived from the root chain, the main source of truth within the system.

FourthState, a team comprised of Blockchain at Berkeley's members, wrote an implementation of Plasma using the Cosmos SDK, enabling further flexibility and scalability.

Blockchains have 3 main abstraction layers, from top to bottom:

- The application layer processes transactions and updates the state of the system
- The consensus layer makes sure the entire network agrees on transactions and updates to the database
- The networking layer makes sure all nodes get updates within a reasonable amount of time