Cosmos

Introduction

Cosmos labels itself as *The Internet of Blockchains* by offering an open-sourced framework for blockchain development denoted as the Cosmos SDK. The Cosmos SDK can be utilized for building public Proof-of-Stake (PoS) blockchains or derivatives of them such as permissioned Proof-of-Authority (PoA) blockchains. These chains are denoted as application-specific blockchains due to the customizability, allowing for a chain to be optimized to operate a single application. All chains generated using the SDK are naturally interoperable with one another, hence Cosmos' title. Some extremely popular chains were built using the SDK like Cosmos Hub, Binance Chain, Terra, and Cronos.

Section 1: Technical Design & Performance

Technical Architecture

Cosmos SDK

The Cosmos SDK is the main product offered written in the Go programming language that provides a secure, composable, and easy-to-use framework that facilitates the development of efficient, interoperable blockchains. The three main components for building a new blockchain are the state-machine, a consensus mechanism, and a method to connect the two.

Developers using the Cosmos SDK can fully customize how they define the state of their application, transaction types, and state transition functions. The created state-machine is naturally deterministic, meaning that the initial state processing the same sequence of transactions will have a predictable final state. A consensus method is needed to handle the verification and ordering of these transactions, and currently, Tendermint is the only mechanism available and consists of two components: Tendermint Core and Application Blockchain Interface (ABCI). Tendermint Core relies on a Byzantine-Fault-Tolerant algorithm to propagate and order transactions in the form of bytes. This ordering of transactions is partitioned into blocks which is appended and verified by a group of nodes called a validator set. Tendermint Core then passes transaction bytes to the application via the ABCI which then relays back to Tendermint whether the message contents were processed. This will be expounded upon in detail later on (see the Consensus section)

The chains generated using the SDK are application-specific blockchains, which are modular by nature in the ever-growing network of Cosmos chains. Unironically, the blockchains instantiated leverage this modular design internally to promote throughput and interoperability. SDK applications are created by aggregating a collection of interoperable modules. These modules each represent different pieces of the state-machine and manage their own message/transaction processor. The application layer routes each message to the correct module, or the aspect of state that is being updated. These modules can include but are not limited to authorization, banking, staking, and governance.[i]

Starport

Starport is an easy-to-use command-line interface (CLI) tool built on top of the Cosmos SDK to further streamline chain development so users can focus on the application's logic instead of the technicalities. This is referred to as a 'scaffolded' blockchain that includes a CLI that allows the management of keys, validators, and tokens. Using simple commands within the CLI it is possible to create a modular blockchain written in Go, scaffold modules/messages/etc., start a blockchain node, connect to other blockchains using a built in Inter-Blockchain Communication (IBC) relayer, and use auto-generated Typescript/Vuex clients. Starport has already experienced tremendous usage to create chains like Cronos, Polynetwork, Celestia, and Juno.[ii]

Inter-Blockchain Communication Protocol (IBC)

IBC is a protocol that allows arbitrary state-machines to communicate arbitrary data. Working within the framework of the SDK, this allows for any programming language, any state-machine, or any sort of Virtual Machine (VM) implementation to be used without affecting its ability to be interoperable. The protocol comprises two layers: the transport layer (TAO) and the application layer. The TAO provides infrastructure for blockchains to establish secure connections for data packet verification while the application layer identifies how the data packets should be packaged and interpreted by the involved chains. The IBC enables cross-chain functionality for various procedures including but not limited to token transfers, staking, governance actions, and oracle data feeds.[iii]

Cosmos Hubs & Zones

Cosmos Hub is the first blockchain instantiated on the Cosmos Network and it is home to the ATOM token. ATOM is used to secure the network via a Delegated-Proof-of-Stake (DPoS) methodology where validators stake their ATOM tokens and other users delegate their share to them. Cosmos Hub and its many zones are the first instance of the interchain network, but all new blockchains created via the Cosmos SDK are classified as 'hubs' or 'zones'. Hubs are designed to be the centerpiece of the zones, effectively connecting them together. Hubs are responsible for a central multi-token ledger to record the state for all associated zones and vice versa. The hub is not capable of tracking the state of the zones; however, the zones can use a light client to track the state of the hub. The hub updates the state of all zones tethered to it when zones send messages to other chains, which is handled by Tendermint and the IBC Protocol with the hub is the intermediary. There is no limit to the possible number of hubs and zones.[iv]

Cosmos Architecture Advantages & Disadvantages[v]	
Advantages	+ **Flexibility:** ABCI allows for developers to swap consensus engines build their state-machine in any programming language by wrapping the ABCI in a language of their choice. Developers have no constraints as to which VMs. This customizability also can bolster security when implemented correctly. + **Performance:** If made to be an application-specific blockchain, operating a single application means there are no other applications vying for the same computational resources. The chain being modularly reduces internal network congestion. The network being modular promotes versatile scalability when coupled with interoperability. + **Interoperability:** IBC Protocol allows for the communication of arbitrary data between arbitrary state machines. This allows for full customizability and no need for cross-chain liquidity. + **Sovereignty:** Most blockchain networks have mixed incentives for the network itself and the underlying applications, potentially resulting in a lack of participation. Application-specific blockchains' stakeholders have full control of the application and blockchain, narrowing any divisions or conflicting interests.
Disadvantages	– **Limited Functionality:** The nature of application-specific blockchains more than likely brings a myopic use case. This promotes modularity, but many of the chains created will have limited functionality on their own. – **Small Validator Set:** Cosmos Hub's small validator set (150) exposes it to potential centralization security threats. Also, SDK applications have their own unique validator sets which may be heavily centralized. – **Custom Flaws:** With full customizability comes the risk of poorly written code or potentially unpredictable security breaches. Standardizing development in many languages is not possible in the infancy of blockchain development.

Overall Cosmos Architecture Advantages & Disadvantages vs. Polkadot	
Advantages	+ **Consensus Finality:** Cosmos utilizes the Tendermint consensus mechanism involves the production and finalization of one block at time, creating a deterministic and reliable environment with near instant finality. Polkadot uses a hybrid consensus protocol with both BABE and GRANDPA that finalizes transactions in batches which delays the process and opens the possibility of more frequent network rollbacks. + **Scalability through Modularity:** Cosmos' SDK constructs each chain, or zone, to be modular and thus expediting finality. The macro view of the network is also modular in that each zone is application specific and connected to centralized hub. With myriad hubs and no limit to them, the modularity expands infinitely along with efficiency. Polkadot is similarly modular, but is curtailed by
Disadvantages	− **Upgrades:** Cosmos does not have a meta-protocol like Polkadot which means upgrades would always result in a fork of the network. Polkadot features WASM meta-protocol that allows for state transition function (STF), off-chain workers, or transaction queue upgrades without hard forking. − **Heterogenous Chain Security:** Both Cosmos and Polkadot employ a version of sharding, however Cosmos's zones have their own unique validator sets and governance. This prevents the network from wide scale vulnerabilities (as seen in Polkadot's parachain rollbacks), however many of these zones may run the risk of becoming very centralized or contain glaring security holes.

Governance

Cosmos' provides infrastructure to build blockchains that are unique in their architecture and governance systems. Common issues such as hacks and bugs often need rapid resolutions, and a governance system allows for an organization of ideas to be decided upon. Depending on the application, there might be different interests on what needs to be governed. Validators and delegators for Cosmos Hub can submit proposals for a 512 ATOM fee that will be refunded upon the proposal's acceptance within 2 weeks. The topic of these proposals can include a change in parameters that determine the network's economic structure, the coordination of network upgrades, as well as other policies. For instance, Cosmos Hub currently has a small validator set of the top 150 (was 100) validating stakers by size being active. A recent proposal has suggested the expansion of the set to 175 with the eventual goal being 300 validators.

Discussions for all proposals take place on the Cosmos Forum with a 2-week community deliberation after its acceptance. A quorum of 40% of the total staked ATOM is needed to enact a given proposal with the majority of the participating votes (>50%) needed to pass. Delegators can vote based on their tokens in each validators possession but refraining from a vote implies that the validator assumes their staking power. The proposal can also be vetoed with 33.4% of the token holders resulting in a validator wide fee (1 days worth of blocks) and an additional fee for the vetoing party (0.1% of their ATOM). The voting period then lasts 2 weeks with the implementation process happening immediately.[4]

Cosmos Governance Working Group

The Cosmos Governance Working Group (GWG) was established in 2019 to solve the problem with low community engagement. Soliciting engagement with a proposal proved to be quite difficult with the needs for a convincing argument, community engagement, and a large ATOM deposit. Without attention from the community, even from well-crafted proposals, there was a likelihood that validators with a large delegated stake become centralized powers effectively controlling the direction of the network. The delegator lack of participation was primarily due to many barriers adding friction to the action.

To shrink these barriers and encourage the community to vote, the GWG generated educational documentation and tutorial. These included *The Cosmos 'Parameter Change Wiki', Community spend Best Practices,* and the *Cosmos Parameter-Change Documentation*. The act of voting was a large issue that involved delegators needing the validator's private key to sign along with setting up private key protections, complicating the process. GWG offered a solution in the form of creating a subkeys or group keys to make voting easier. Validators can designate this less secure key without exposing their private key, simplifying the process.[vi]

Interchain Foundation

The Interchain Foundation (ICF) was established during the birth of the Cosmos Launch in 2017, where they raised a public fundraiser event that ended in April 2017. The ICF received an initial supply of 23,619,895 ATOM which was roughly

10% of the initial supply of ATOM. Since then, they have developed a team to help grow the Cosmos ecosystem. Ethan Buchman, cofounder of Cosmos and Tendermint is the current president of ICF. There is no community governance where the entire treasury is managed by the ICF Board and Foundation council. Their total assets as of Q1 2021 were 513 million, where this money is being used to help fund grants and promote growth in the Cosmos ecosystem. Their medium is very transparent and posts any updates to the ICF council as well as funding recipients.

Cosmos Governance Advantages & Disadvantages vs. Polkadot	
Advantages	+ **Permissionless Experimentation:** Each zone in a hub can employ their own unique governance system providing opportunities to see what works. At the hub level, there could be a mechanism in place to encourage immutability, removing the possibility of roll backs. + **Validator Participation:** Validators are required to vote on proposals and are subject to downtime penalties (default 1 week). This prevents validators from getting complacent.
Disadvantages	– **Powerful Validators:** If a delegator refrains from voting in the Cosmos Hub, their underlying validator assumes their staking power, giving a potential boost to their voting power. This can create a skewed distribution of tokens among active participants. Polkadot's validators never get any voting power from their respective nominators. – **No Adaptive Quorum Biasing:** Polkadot adjusts the quorum threshold based on the percentage of voter turnout. This keeps the passing of referendum's more secure, especially at lower participation rates. Cosmos employs no such methodology and has a strict 40% quorum which may prevent any governable actions. – **Simple Coin-Vote:** Cosmos Hub voting uses tokens amounts as the sole metric for decision making. Polkadot voters can increase the weight of their tokens by locking them for a period of time, adding another layer of complexity to maintain a dynamic and secure governance environment.

Network Decentralization

Cosmos generally has a slightly greater degree of validator centralization compared to other networks. Cosmos Hub currently has a max validator set of 150 with plans to continue its expansion to 300 by the 10th year since inception. Tendermint is a revolutionary consensus method that performs very well with a more centralized validator set. Tendermint's TPS is diminished as the validator size increases creating natural restrictions to the level of decentralization. With the size of per chain within the network being more rigid to optimize performance, the distribution of tokens across validators is another method to quantify decentralization.

The Nakamoto Coefficient is the minimum number of validators who can halt block production which is an important metric of a network's degree of decentralization. As of April 2022, the top 7 validators control more than ⅓ voting power of the Cosmos network meaning Cosmos can be effectively shut down if they go offline, either due to an attack or they decide to collude with each other. This is less than the 12 validators that control ⅓ of voting power on Terra, another blockchain built using the Cosmos SDK, and significantly less than Solana (21) and Avalanche (27), but not as dramatically low as Polygon, where 2 validators control ⅓ of the voting power.

Herfindahl scores are calculated as the sum of the squares of each validator's stake share ranging from 0-10,000, with a lower score indicating a higher degree of decentralization. Cosmos protocol has a Herfindahl score of 240 compared to Polkadot's score of 34. Overall, these numbers indicate a very even distribution of tokens amongst its validators, evident by Polygon's score of 963.

Since decentralization is a measure of the magnitude and distribution of a network's validator set, understanding the upfront costs and resources required for becoming a validator are a highly relevant, and perhaps leading indicator of decentralization. From a hardware perspective, validators must operate machines with at least 4 CPU cores, 16 GB RAM and 500 GB disk space. This would put Cosmos on the middle of the pack, less intensive than Solana (12 CPU cores, 128 GB RAM and 2000 GB disk space) and Polkadot (8 CPU cores, 32 GB RAM and 500 GB disk space), but slightly more intensive than proposed specs for Ethereum 2.0 validators (4 CPU cores, 8 GB RAM and 500 GB disk space).

However, Cosmos does also require a minimum stake of ATOM requiring validators to bond several tokens that is safely within seven figures at current market value. Cosmos also reimposes a 21-day unbonding period for any staked capital. While this is not extremely uncommon (Polkadot and Avalanche require similar unbonding period), it does impose cost in the form of capital inefficiency.

	Decentralization Metrics							
Platform	Validator Count	Total Value Staked ($MM)	Nakamoto Coefficient	Herfindahl Score	*Min. Staked*	*Max. Staked*	*Med. Staked*	*Avg. Staked*
Cosmos Hub	150	$5,475	7	260	0.04%	6.13%	0.26%	0.67%
Polkadot Relay Chain	297	$13,882	100	34	0.29%	0.54%	0.32%	0.34%

Security

With most PoS blockchain protocols, the 2/3 threshold of honest validators is crucial to keep healthy. A network takeover is unlikely, but the censorship of the network can be facilitated by a smaller group of malicious nodes (<1/3). Slashing is a consequence that disincentives negative behavior by eliminating staked funds, however, malicious nodes can reject blocks strategically if they have enough staking power to elongate the consensus process and potentially halt the network. This might be done to censor transactions or delegitimize the network which may render slashing inconsequential depending on context. If a powerful adversary existed, it could be possible that the wrong subset of validators could be blamed for the slow down, obfuscating the attack. This is not unique to Tendermint and is a possibility for most alternative PoS systems.[vii]

Interchain Security

To defend against such an attack or halt on Cosmos' smaller zones, Cosmos is developing Interchain Security which is enabled by Cross Chain Validation, an IBC-level protocol. To protect new or underdeveloped zones from being compromised, Interchain Security aims to assist smaller blockchains become more secure by allowing larger chains to aid them. This will be achieved by provider chains sharing their validators in charge of producing blocks, where said validators would be able to run a node on the provider chain and the consumer chain. These validators would naturally receive rewards from both chains in the process of fortifying the consumer chain's security. [viii]

Recorded High-Severity Vulnerabilities

Cosmos-SDK and Tendermint have experience nuanced vulnerabilities since its inception that were addressed swiftly by their development team. They have detailed reports on the sequence of events and how they were handled in the Cosmos forum.

Advisory Magenta

On September 20, 2019, a bug was discovered through Tendermint's bug bounty program that involved the handshake process between the Tendermint Core and the ABCI. The handshake process of passing a public address lacked a check of whether the address was valid. This opens the door to a Denial of Service attack, however the possibility of an attacker hunting down specific nodes with enough stake to impact the network that also happen to be taking requests openly is highly unlikely. An initial patch was released within a week that had its own issues. The vulnerability was fully fixed by October 12, 2019 with no exploits occurring. [ix]

Advisory Mulberry

On January 8, 2021, Crypto.com discovered a pair of vulnerabilities regarding Tendermint Core's evidence handling. The first showed it was possible for validators to propose blocks at heights that weren't fully committed yet. Due to the gossiping process, other validators would receive requests for essentially nonexistent blocks. This causes a remote panic and a Denial of Service. The second vulnerability was discovered a day after the patch for the previous problem was released on January 12, 2021. Timestamps were added to the evidence handling process and can be utilized to monitor double block signings. This evidence is formed during a double sign event but because the block has not been committed yet. Different nodes being passed the block will create evidence with different timestamps, rendering all but one of these invalid. The now invalid evidence will continue to circulate throughout the nodes causing a panic and a Denial of Service. Executing a DoS attack via these methods is tremendously expensive and unlikely and the Cosmos Hub was unaffected. A patch was released by January 19, 2021.[x]

Advisory Jackfruit

On October 8, 2021, the team at Provenance found a vulnerability in the Cosmos SDK that can result in a consensus halt. One of the modules created using the SDK, the x/authz module, contained a function that includes a user defined expiration time in reference to the node's local clock. Local clocks are inherently subjective and can cause confusion amongst nodes in the network. It is very possible some nodes would observe an expired function call while others did not, falling into a nondeterministic environment and subsequent consensus halt. A patch was released 4 days later that removed the method entirely, however it may be added back on a later date using the BFT clock that is agreed upon by consensus.[xi]

Consensus

Tendermint is an alternative consensus solution theorized by Jay Kwon in *Tendermint: Consensus without Minting* in 2014 and published by Jay and others as *The latest gossip on BFT consensus* in 2018. Tendermint is revolutionary because it not only solves the Byzantine General's Problem with a unique PoS consensus mechanism in Tendermint Core, but it is used in tandem with the Application Blockchain Interface (ABCI) that enables transactions to be facilitated to any application, no matter the programming language.

Tendermint Core

Tendermint Core is a deterministic BFT consensus mechanism implemented that is able to guarantee security and liveness with up to 1/3 of the network's validating stake acting maliciously. In other words, it is a state machine replication engine that is referenced before changing the actual state. Consensus between validators is reached via one or multiple 'rounds' comprising a sequence of three steps (Proposal, Prevote, and Precommit) as well as two special steps (Commit and NewHeight). This is done through varying degrees of synchronicity and centralization.

Block proposals are signed and published by a selected validating node each round. Tendermint Core is a 'leader based protocol' where the leader selection process takes place in a queue. The validator set associated with every round at a given block height is static but can change in between rounds as validators rotate in and out of relevancy, creating a deterministic environment. The ordering of the queue is a function based on validator staking power (amount skated) and their recency of block production. In other words, validators with larger stakes produce blocks more often and suffer less consequences in terms of queue ordering after a block has been produced. The result is a fair sequence of proposers whose frequency of production reflects their network stake. Amidst the last steps of block production, there is an optional round that hints towards the unlocking of the node, finalizing the block and ensuring the liveness of the network.

Tendermint Core employs a gossip mechanism to inform nodes of the current state, metadata, and other important details needed to maintain the network's liveness during the consensus process. Once a block has been proposed, each validator broadcasts its prevote vote. Validators locked in a previous round are allowed to unlock with a proof of their previous commit, allowing them to prevote on the next round. The validators prevote positively if the block is valid or prevote *nil* if the block is invalid. If a validator is still locked, they continue to prevote the previous block, hence a degree of asynchronicity. After one round of voting, if 2/3 of the validators have consensus on the proposal, validators continue to the precommit step. Otherwise, the rounds continue until consensus is reached either for or against a block's production.

If a validator still has not been unlocked for the current round, it precommits *nil* since its attention is elsewhere. Otherwise, each validator broadcasts its precommit vote waiting for the 2/3 consensus to fully commit or reject the given block. After a block commits, the height is incremented to ensure the following block is unique from the previous.[vii]

Tendermint Core Safety and Liveness

Assuming no more than 1/3 of validators are malicious, the safety of the network can be guaranteed through the rounds of consensus and locked nodes. If a block commits in a given round, that means at least 2/3 of validators precommited. The proof that unlocks nodes is not known by every block producer, however, one will be found eventually. There are safeguards like timeouts and capped proposal sizes that ensure the liveness of the network, but the fragmentation of nodes able to prevote or commit keeps the validators honest and inherently prevents a potential weak moment where the malicious 1/3 can gossip false information. The network must wait for the previous honest nodes to unlock before proceeding.[vii]

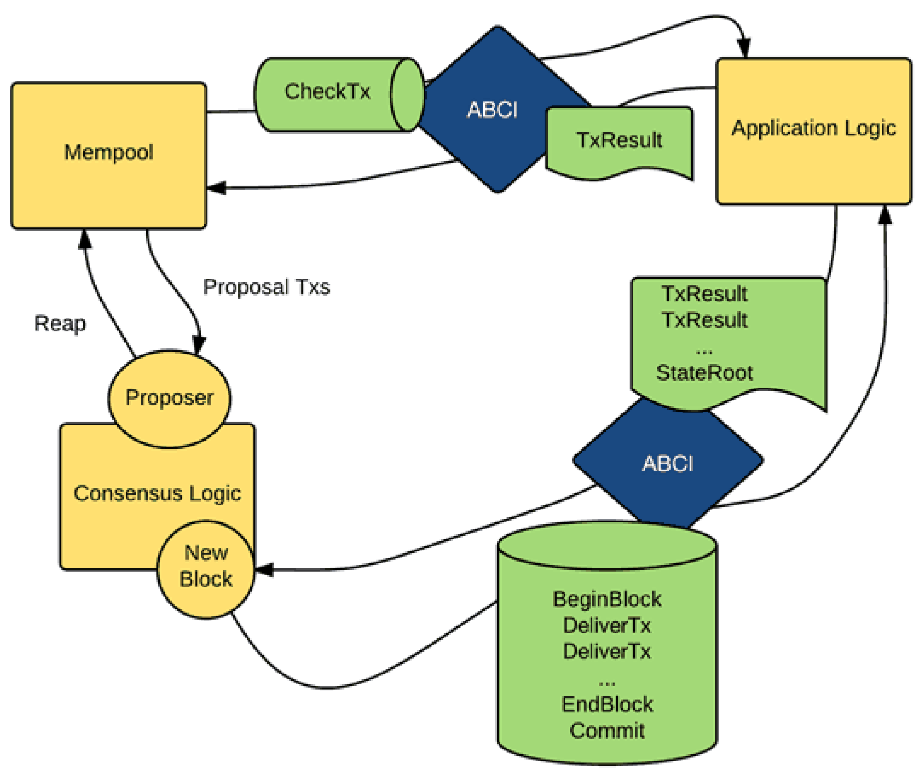
Application Blockchain Interface (ABCI)

ABCI is the interface between the consensus method Tendermint and the application/state machine. ABCI has a variety of methods that are called by

Tendermint in the form of a *Request* with the answer classified as a *Response.* The methods and messages are defined at a low level so applications can be written in any programming language. For ABCI applications to be securely replicated by Tendermint, the connections between them and the application itself must be deterministic in that all nodes will compute identical responses.

ABCI applications can run with the same process as Tendermint or an entirely different consensus engine. If it runs the same process, the ABCI methods are made directly with Go method calls. If they are separate, ABCI creates 4 connections that handle different aspects of the communications such as consensus, mempool, info, and snapshot. The consensus connection is designated for block execution which entails initializing the blockchain, beginning block production, delivering transactions, ending block production, and committing blocks to the chain. The delivering of transactions is where Tendermint submits a request to the ABCI which the ABCI receives and transports the data to the application to update the state. A Response is then sent back to Tendermint where it is subsequently logged without affecting consensus. The mempool is for transactions pending block assignment and there is a check to ensure their validity before they are compartmentalized in a block. The info connection handles data queries from the user. The snapshot connection is for saving and restoring various states of the state-machine (time-machine). In the event of a security breach, snapshot of the state at a specific time can be used to restore the network to a prior, stable state.

Implementing the ABCI in a select programming language, there must be an ABCI server built in that language. Tendermint supports three implementations of the ABCI: In-process, ABCI-socket, and GRPC. In-process is written in Go and can be directly connected with Tendermint easily. GRPC is the easiest approach to develop an autogenerated server in the language (if available) of your choice at the cost of



performance. For those that like to code and desire higher performance, the Tendermint Socket Protocol (TSP or Teaspoon) can be used to implement a custom ABCI. The TSP is an asynchronous socket server that ensures the messages/transaction maintain their order as they go from Tendermint to the application. The connection support and other supporting faculties need to be constructed deterministically to ensure harmony with Tendermint.[xii]

From blokt.com

Performance (TPS)

Transaction throughput is a tricky metric in that its significance can vary dependent on context. For one, transactions can vary in type and in the amount of compute resources they require. Furthermore, conditions that affect a network's ability to process transactions, like the number of validators and network latency, can vary, making TPS capacity variable depending on the circumstances. These and other considerations apply when assessing TPS performance. We thus look at several different ways of measuring TPS to provide a more comprehensive picture of performance. Since it's impossible to pinpoint a single accurate TPS performance number, we include different metrics to illustrate possible performance ranges for the platforms analyzed. We thus look at developer TPS estimates, theoretical TPS based on platform parameters, and on-chain network max TPS results where available.

Developer Estimates

Platform developer estimates tend to be the most widely available data points for platform performance. However, they can be unreliable given developer incentives to paint their platform in the best possible light. We still present them to provide a view into what core developers say their platform can do.

Tendermint under extremely centralized validator conditions boasts 14,000 TPS. This would never happen under real conditions and the estimate of 4,000 TPS was with 64 active nodes. The Cosmos Hub currently has 150 validators, so it is likely that this valuation hyperbolized and not realistic.[iv]

Developer Estimates of Platform Performance			
Platform	TPS	Environment	Certainty Level
Cosmos Hub	4,000	Testnet	Medium
Polkadot Relay Chain	1000	Internal Benchmark	Low
Individual Substrate Chain	700-1500	Internal Benchmark	Low
Polkadot Ecosystem	70,000-150,000	Internal Benchmark	Low

Mainnet Max Results

Cosmos Hub and any of its zones have not substantially been tested and the parameters of each chain vary enough that the data is not meaningful to record.

Finality

Tendermint prioritizes fast finality by allowing validators to perform partially asynchronously, preventing network slowdowns. Cosmos Hub has a small validator set relative to most chains, sacrificing decentralization in favor of speed. This is not necessarily true for all of Cosmos' hubs or zones since each employs a custom validator set. These numbers could begin to skew with larger sets.

Platform Time to Finality			
Platform	Block Time (s)	Confirmations	Time to Finality (s)
Cosmos Hub	7	1	7
Polkadot	6	N/A	12-60

Scaling

Cosmos' bottom-up modular architecture lays a strong foundation for the network's scalability. The blockchain's internal modules delineate tasks within the application which protect against channels getting clogged or overburdened. The applications or zones are connected to a central hub that provides accessibility to every zone through an interaction with the IBC. This hub/zone relationship creates a network of application specific chains coalescing with a central hub that allows for the delegation of tasks per zone. Many hubs working together begins to create the 'Internet of Blockchains' framework. This is sustainable from a liveness perspective; however, a heterogeneous nature of the network brings its security drawbacks.

Each hub or zone has its own independent validator set which runs the risk of centralization and therefore security issues. New zones can be instantiated quickly using the Cosmos suite of tooling but depending on the community involved, investors, or general marketing effort, the initial involvement within the project opens the door to malicious validators easily taking over the network or even having a foothold for a later date. Therefore, fair initial token distributions need to be handled carefully. That being said, these separated chains are responsible for validating itself while it shares its state with the hub. The network can scale infinitely without getting overburdened as long as the new zones have sustainable resources (numerous honest validators and large capital allocations.

Cosmos Scaling Approach Advantages & Disadvantages vs. Polkadot	
Advantages	+ **Heterogeneous Network:** Each zone being independent from others allows for a unique validator set unburdened by myriad transactions running through the entire network. With IBC, every chain can run an application specific chain and relay that data between each other and a centralized hub seamlessly. In Polkadot, there may be consistent congestion in the future since everything runs through the relay chain. + **Limitless Hubs/Zones:** There is no limit to the number of zones and because each validator set is independent, the network does not get congested from more zones. The only caveat is that the hub has a larger ledger to update which is inconsequential at this level, and even if it was, a new hub can be instantiated. Polkadot's shared security and parachain limit of 100 hinders the speed and scalability of the network. + **Customizable Interfaces:** The ABCI can be implemented in any programming language removing any restrictions for blockchain developers with a strong skillset outside of solidity. With soon launch of Evmos, many new IBC & EVM compatible chains can be instantiated to help Ethereum scale or leverage other EVM chains to help scale Cosmos outside of its hub/zone model.
Disadvantag	− **Security:** Cosmos Zones' validator sets are each unique and therefore run the risk of being centralized or secured by small amounts of capital. This can be partially solved with Interchain Security, however, Polkadot remains much more secure since the entire network shares security and validators.

Use Cases

Cosmos has the potential to become a ubiquitous ledger with many use cases. Currently the Cosmos Hub hosts many applications that consists of topics related to DeFi, development, interoperability, security and more. Much of it is clearly still infrastructure, but this can very easily spread to large GameFi products or catered enterprise data bases very soon. The centralization of validators may not be a concern for games and companies looking to keep track of internal resources. This allows them to optimize the chain created with SDK to maximize performance at the loss of security, with the security fully under their control. Transactions that are not that meaningful to non-gamers or users outside a given company won't have to be exposed, and individual zones remain clean and organized. Hubs further promote this segregation of data.

With the IBC and very mutable ABCI, it is possible to create chains developed in any programming language that is interoperable with any chain within the ecosystem. The long-term effects of this cannot be overstated. Payment networks created across the world by developers with very different backgrounds will be able to seamlessly connect with another and conduct business.

Gas & Fees

Cosmos Hub validators receive transaction fees as a reward for validating blocks just like most other blockchains, however, validators are able to accept any token type or combination of types in the Cosmos ecosystem. Validators have full control to set whatever exchange rate they want and which transactions they want within the bounds of the gas limit. The validators are paid out every hour based on their stake in the network, minus a 2% Reserve Tax that goes to the reserve pool. The delegators pay a validator assigned commission rate.[4]

The current average transaction cost for Cosmos Hub and Osmosis is shown below. It is notable that Cosmos Hub has comparable sizing to some of the parachains. Osmosis currently does not charge any fee to swap or transfer tokens in their zone, although, paying gas is optional to speed up the transaction.

Transaction Fee Comparison						
	Native Token Transfer		**Non-Native Token Transfer**		**AMM Swap**	
Platform	Gas Fee	USD Value	Gas Fee	USD Value	Gas Fee	USD Value
Cosmos Hub	0.0002 ATOM	$0.005	0.0002	$0.005	-	-
Osmosis	~0	$0	~0	$0	~0	$0
MoonRiver	0.00004 MOVR	$0.002	0.00005 MOVR	$0.003	0.0004 MOVR	$0.02
MoonBeam	0.002 GLMR	$0.005	0.005 GLMR	$0.01	0.01 GLMR	$0.03
Astar	0.00002 ASTR	$0.000002	0.00005 ASTR	$0.000005	0.0001 ASTR	$0.00001
Polkadot Relay	0.016 DOT	$0.28	-	-	-	-

Ease of Development (i.e. how easy it is for engineers to build)

Cosmos currently offers a large amount of tooling to help developers easily build. WebAssembly (WASM) is a lightweight programming language and execution environment for smart contract development on Cosmos. If the independent developers are dedicated enough to build a custom ABCI, any programming language can be used in the creation of application. However, the most popular languages used on Cosmos are Go, Rust, Solidity and Vyper.

The Cosmos SDK provides a framework to build blockchain applications, and it is further streamlined using Starport. Starport allows developers to build a blockchain with a developer-friendly interface alongside pre-built apps, auto generated boilerplate code, amongst many other functions. Since the Cosmos SDK is open source, there are specific developer SDK chats on Discord as well as a forum to help developers get in direct contact with others building on the SDK.

There is also a suite of unique tools/solutions available within the ecosystem that includes Agoric Swingset, CosmWasm, Ethermint & Evmos, and CosmJS.

Agoric Swingset allows developers to test smart contracts in different blockchain setup environments. CosmWasm is a smart contract module built on the Cosmos-SDK where smart contracts can run on multiple chains using the Inter Blockchain Communication protocol. Ethermint is an Ethereum Virtual Machine (EVM) Proof-of-Stake blockchain which runs on top of Tendermint Core. Through Ethermint, developers are able to use Ethereum ecosystem libraries like web3.js and ether.js, however this idea has expanded and evolved into Evmos (see Novel Applications). Cosmos has also created its own optimized typescript library to power JavaScript client solutions called CosmJS.[xiii]

Tendermint Core has built in tooling to assist with its implementation which includes debugging, benchmarking, testnets and validation of remote signers. *Tm-load-test* is a framework testing tool for load testing Tendermint networks. Tendermint contains many different test networks for monitoring Tendermint's performance where the repository contains *tmtestnet.py,* an execution tool that automatically combines relevant scripts to allow deployment to Tendermint test networks. Tendermint's many debugging commands allow developers to kill live Tendermint processes and archive the consensus state, configuration used, network state and other important data. Tendermint Inspect allows developers to debug inconsistent state and RPC endpoint access.

Interoperability

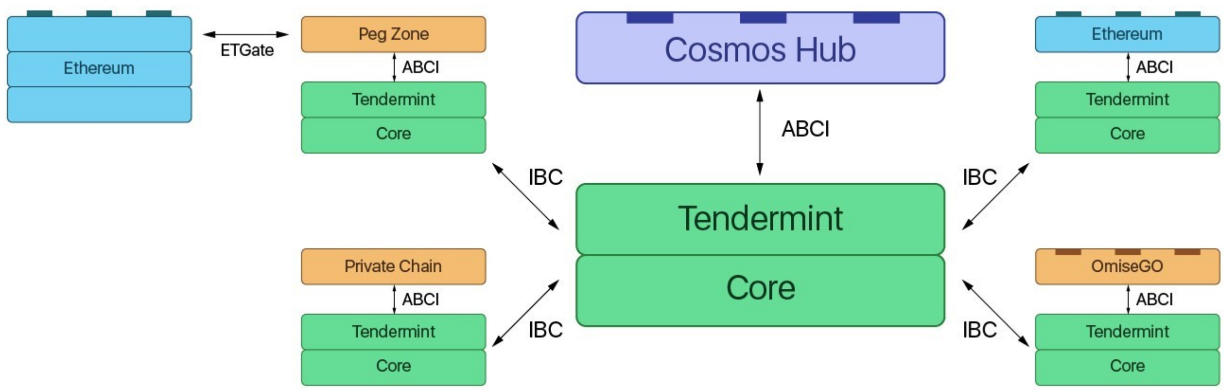
Inter-Blockchain Communication Protocol

IBC is an end-to-end connection-oriented stateful protocol which allows for the communication between blockchains built on the Cosmos SDK. Cosmos advertises itself as 'The Internet of Blockchains' (TCP/IP for blockchains) because it allows for any data payload type to be transferred between chains in a standardized way without compromising security or finality. This transactional data can consist of cross-chain token transfers, cross-chain voting, cross-chain token exchange order, account delegation, settlement information, and eventually inter-chain staking. Currently its adoption lies within the Cosmos Hub and its various zones, but its usage will proliferate to other hubs and their respective zones, thus creating the Internet of Blockchains.

IBC relies on certain relayer processes to securely communicate with other modules on various blockchains. This is done using messaging packets over IBC channels that are created to connect two distinct ports on the local hub. The transport, authorization, and organization layer (TAO) is separated from the application where IBC is able to define how data is sent and interpreted across blockchains. However, this does not include how the data should be structured across blockchains. The transport layer specifically allows for communicational access of data packets between chains, whereas the application defines how the data packets are interpreted between chains.[3]

Functionally the IBC relays data between chains, but it also abstracts a meta layer of account management denoted as Interchain Accounts, which tackles the issue of composability between Cosmos zones and interchain transactions. Interchain Accounts allows IBC-enabled blockchains to control accounts on separate chains, giving users the ability to access all IBC-enabled chains from one Cosmos Hub account. Utilizing an interchain account, it is possible to use one chain to indirectly execute an application on another native chain, leveraging the Cosmos Hub as the intermediary.

Cosmos is also attempting to assist and scale other chains outside of its local ecosystem. As an improvement to probabilistic finality chains, Cosmos implemented Peg-Zones which allow for near instantaneous finality through IBC. Peg-Zones help establish another layer of finality while also tracking and relaying the state of the



target blockchain.

From [Medium Post]

Gravity Bridge

Gravity Bridge is an alternative approach to IBC with a focus on connecting to EVM compatible blockchains. Gravity Bridge is a zone where its validators are used to sign transactions for token management instead of a more centralized multi-sig. Through IBC, any zone within the Cosmos Hub can being liquid Ethereum assets into the Cosmos ecosystem and vice versa. It is also being used to transfer NFT tokens between Ethereum and Cosmos with an Cosmos NFT platform called Stargaze.

Gravity Bridge employs the popular method of bridging assets between chains by relying on an Ethereum smart contract and its application instantiated by the Cosmos SDK. In order to transfer assets, they are locked on either blockchain and subsequently minted on the other side. This concept is used by many other bridges like Wormhole, Avalanche Bridge, and Multichain. Currently the Gravity bridge allows the transference of a few assets, but it provides a functional door between Cosmos and other blockchains.[xiv]

Differences in Approach

Cosmos and Polkadot have both developed sophisticated frameworks and solutions to solve the scalability and interoperability problems that plague the industry. However, there are stark differences in their approaches symbolized through Cosmos' hub/zone framework, and Polkadot's parachains.

Polkadot's improved interoperability solution, Cross-Consensus Message Passing Protocol (XCMP) communicates among homogenous shards that share validators and always relay back to the base layer. IBC is similar in principle, but it communicates among heterogeneous chains with different validator sets. IBC utilizes a hub that is similar to Polkadot's Relay Chain. The key difference between the two is that the interoperability of Polkadot is baked into the network, with all interchain communication being routed through the Relay Chain. Polkadot's Relay Chain shares its validator set amongst all parachains that bolsters security at the cost of finality and throughput. Cosmos' implementation of the IBC is optional in that the IBC module needs to be instantiated on a zone before creating a connection to other zones via ports and channels. The hub acts as a port city, creating channels between zones on demand. Even though Cosmos zones have different validator sets, the channel creation removes the need for trust in the other zone since there is implied trust in the hub.

Polkadot also has bridge parachains that work similarly to Cosmos peg-zones to track the state of other blockchain networks. Cosmos utilizes its peg-zones via the Gravity Bridge to connect with Ethereum. Gravity bridge is active and highly optimized, batching its transactions to reduce individual user gas costs of up to 96%.[xiv] For Polkadot, Interlay is currently the only active interoperable parachain and it solely interacts with the Bitcoin Blockchain. Snowbridge is still in development with plans to play the role of Cosmos' Gravity Bridge for Polkadot.

Adoption Risks

Cosmos' largest attack vector lies in the centralization of newly instantiated zones or hubs. The freedom of choice for an incoming validator set includes risks of malicious/thoughtless validator preselection or a clandestine launch allowing certain actors to hold the greatest strength in the network. This may not be done maliciously, but a strong community backing a project drives participation and an even distribution of wealth. Without that, localized network security is certainly a risk.

References

[i] Cosmos Network Docs: Blockchain Architecture
[ii] Starport Docs
[iii] Cosmos Docs: IBC
[iv] Cosmos Whitepaper
[v] Cosmos Docs: SDK
[vi] Cosmos Governance Overview
[vii] Tendermint Core Docs
[viii] Cosmos Blog: Interchain Security
[ix] Cosmos Forum: Magenta
[x] Cosmos Forum: Mulberry
[xi] Cosmos Forum: Jackfruit
[xii] Tendermint ABCI Docs
[xiii] Cosmos Network Tools
[xiv] Cosmos Blog: Gravity Bridge