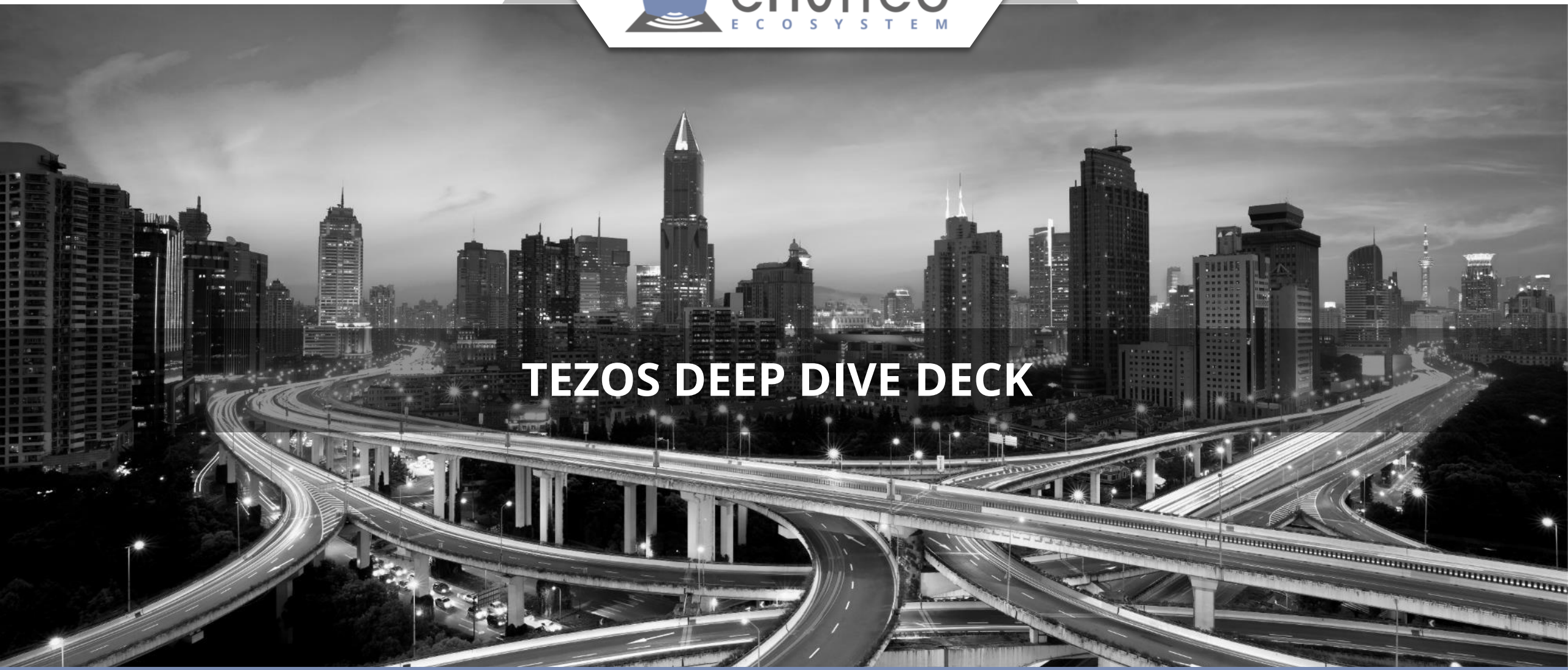
asc(s
Automotive Solution Center for Simulation

envited ECOSYSTEM



TEZOS DEEP DIVE DECK

envited

SUBJECT AREAS

framework for trustfull and traceable data exchange and virtual proof

distribution of high quality data from multiple sources

research and development of new processes and methods

strenghten quantity and quality of experts

PATRON
Automotive Solution Center for Simulation e.V.
non-profit registered association

ECOSYSTEM

- traceability of certificated test data along the supply chain
- support of virtual proof for homologation (ISO26262)
- enables global markets by reducing imbalance of business partners (OEMs <-> Start-ups)

Tezos – the Blockchain Solution to Your Real-World Problem!

Published by asc(s e.V. on behalf of the Tezos Foundation

created by Niko Hildebrand, UNITY AG on behalf of the Tezos Foundation

YOU

Let us tell you something about yourself…

You are
technologically affine
…

*Your superiors **manage** things,*
but they never really get to the bottom of them…

You are well-respected and
your opinion counts
in your organization

Maybe you work
*in the **"old economy"**…*

Your superiors may have heard
*of blockchain during the **hype**…*

You probably have at least
*a **basic understanding** of what a blockchain is…*

...and what your problem might be...

*Your organization operates in an **environment**, that is characterized by...*

*...a **diverse ecosystem**...*

*...**diverging incentives**...*

*...a need for **coopetition**...*

*...a long and fragmented **multi-tier supply chain**...*

*...strict **regulatory requirements**...*

*You face **business challenges**, that are characterized by...*

*...a lack of **trust**...*

*...expensive **intermediaries**...*

*...**certification** requirements...*

*...an obligation to provide **proof** of sth. – e.g. conducted validation...*

*...**auditability**...*

*...lacking market **transparency**...*

*...a need for **privacy**...*

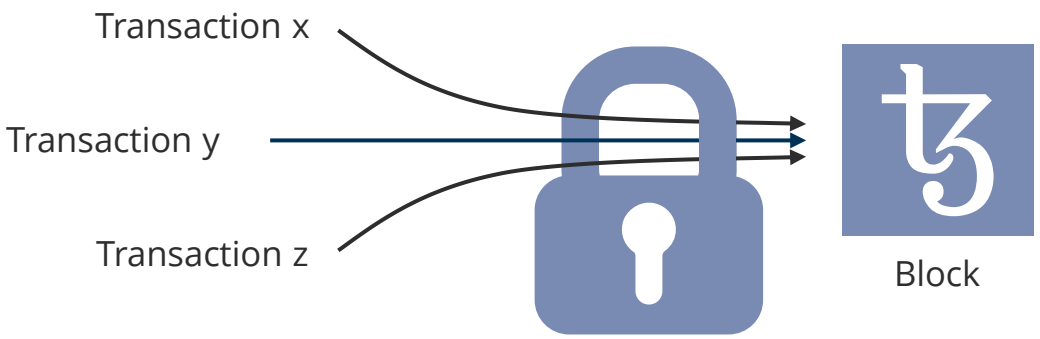
*...the need for tamper-proof **immutable documentation**...*

YOUR PROBLEM

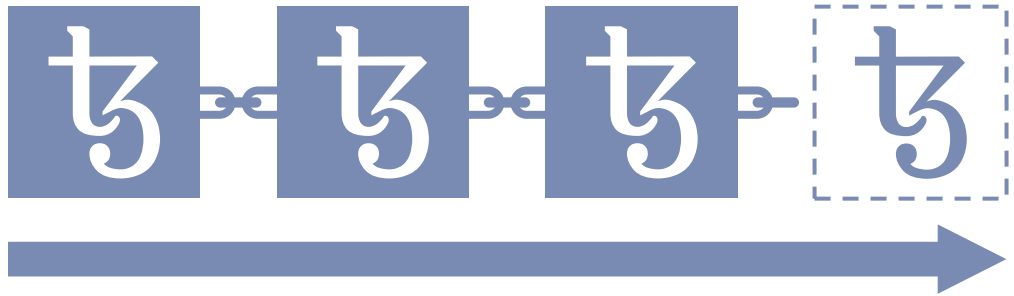
WHY BLOCKCHAIN?

Just a quick reminder – what exactly is a blockchain?

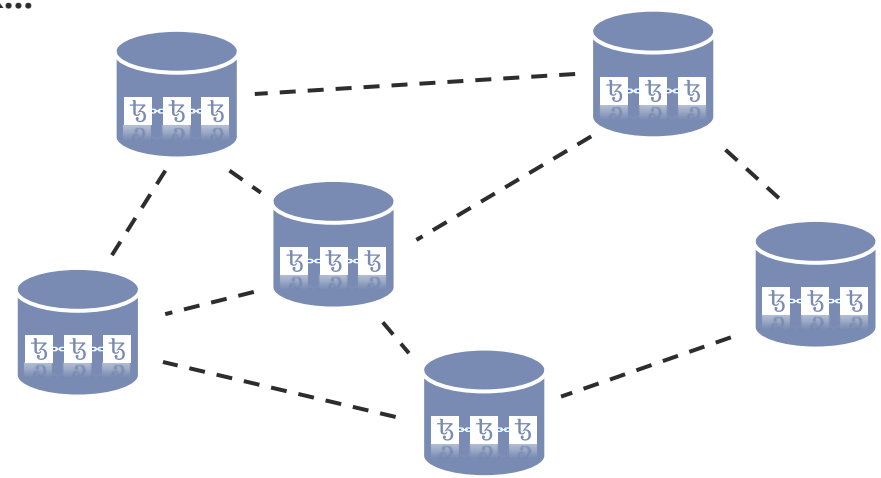
A blockchain is a data structure that groups data (e.g. transactions) into immutable containers called blocks…



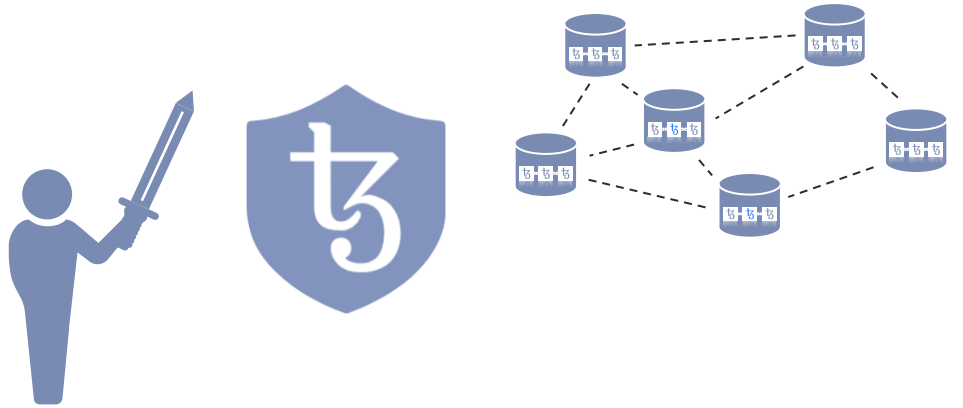
…chains them together in an order-preserving way that only allows appending (but not deleting or editing)…



…is replicated limitless times in a distributed (peer-to-peer) network…



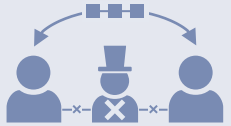
…and maintained by a protocol that aligns participants' incentives in a way that provides protection against fraud and malicious attacks!

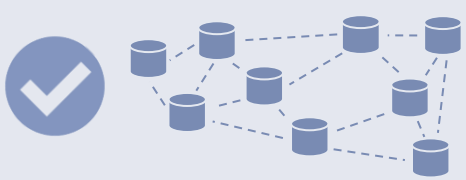


Why blockchain could be a building block for your problem's solution...

Blockchains are **shared** and **immutable** data stores that provide **trust** in **trustless environments**!

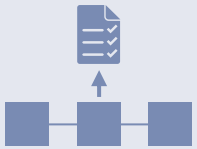
They allow to **save costs** and **increase process efficiency** through **disintermediation**.





They are operated by **distributed Peer-to-Peer networks**
which makes them very **reliable** as there is **no single point of failure**.

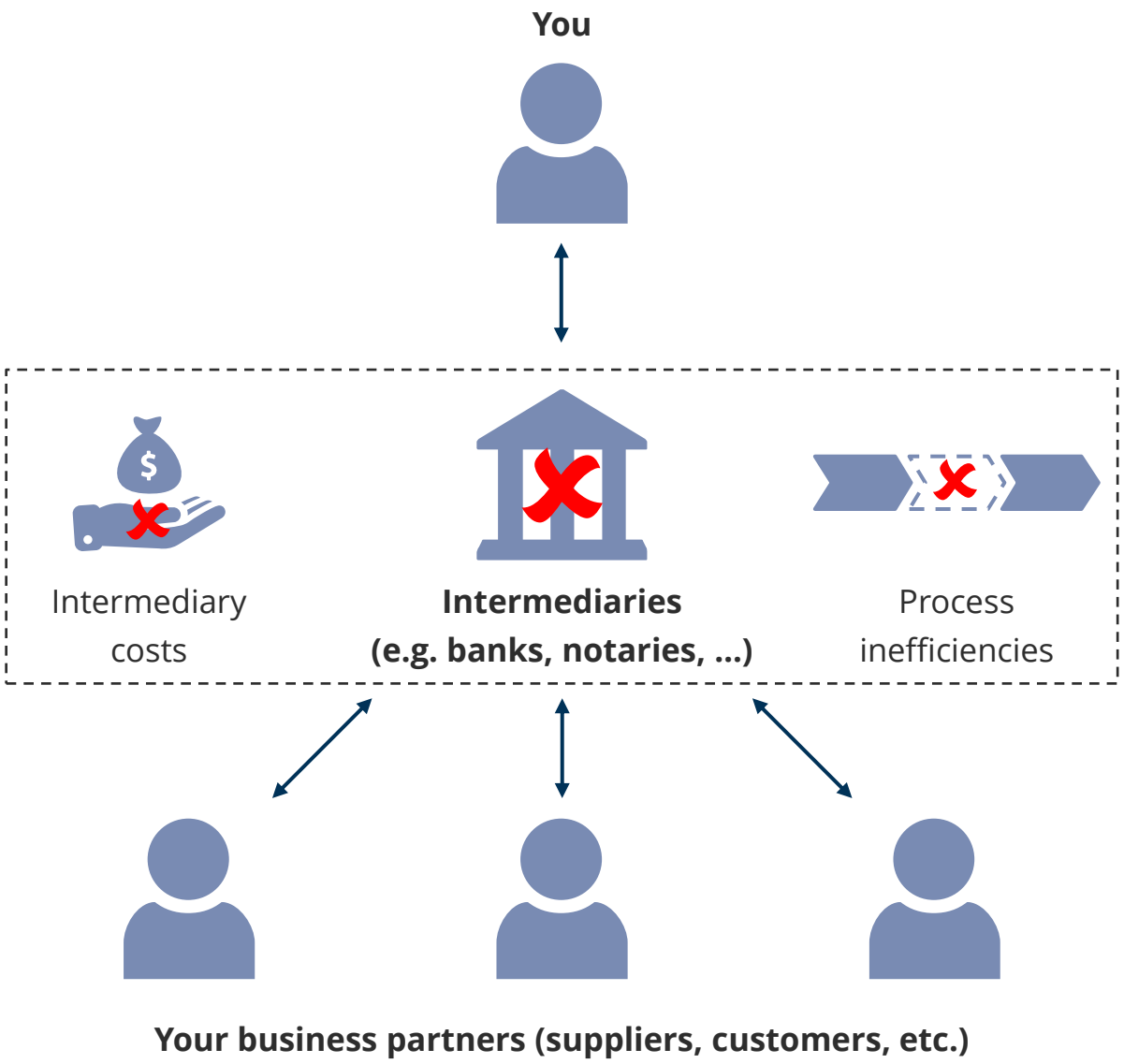
They can be used to provide **transparent documentation**
with **selective privacy** enabling **traceability** and **auditability**.



Their combination of immutability and **accessibility** enables **revocable certificates**.

Why blockchain could be a building block for your problem‘s solution: Disintermediation

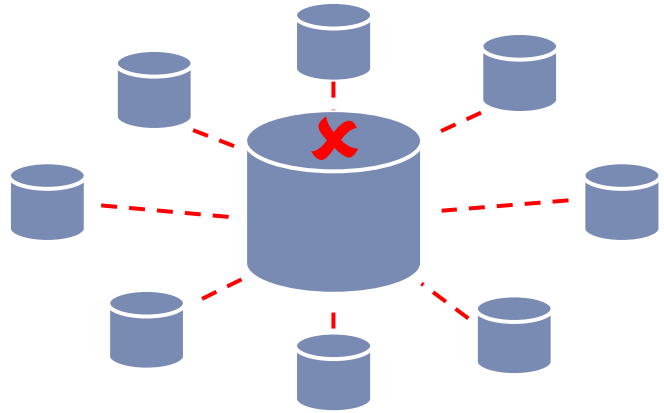
- **Transactions** require two or more parties to fulfill their part of the deal.
- In **physical transactions**, little **trust** is needed since goods and/or money are exchanged **simultaneously**.
- In contrast, **nonphysical transactions** are theoretically prone to **fraud** as the exchange of goods and/or money happens **sequentially**.
- The traditional solution to this problem is the use of a **trusted third party** – a so called **intermediary** – who operates between the transacting parties:
  - If the other party doesn’t fulfill their part of the deal, the intermediary does not release your contribution to the transaction to that party.
  - All transacting parties can trust the intermediary because he has a strong economic incentive to act honestly as his entire business model depends on his **reputation**
- Intermediaries cause additional **process costs** (as they need to be paid for their services) as well as **process inefficiencies** in form of additional process steps and times.
- Blockchain technology allows transacting parties to **directly interact** with each other by **shifting the required trust** from the intermediary to the technology – or rather the network maintaining the blockchain – and thus saves costs and allows to realize process potentials. The removal of intermediaries is called **disintermediation**.



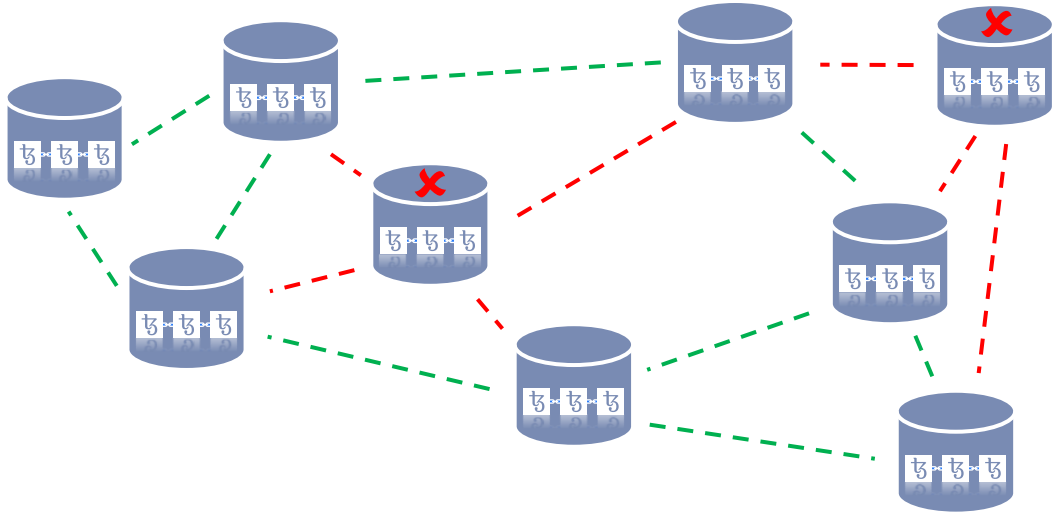
Why blockchain could be a building block for your problem's solution: No single point of failure

- In **centralized networks**, participants are not directly linked to one another but connect via a central network resource that serves as a **data hub**.
- You may know this pattern from **logistics**, where such set-ups are called **hub-and-spoke networks** and the hub serves as a turnover point where goods from various origins are **consolidated** before jointly transported to their destinations.
- While comparatively efficient in various ways, centralized networks have a fatal **weakness**: if the hub fails, the whole network breaks down, which is why there is a **single point of failure**.
- In contrast, **distributed networks** are characterized by direct point-to-point (or **peer-to-peer**) connections between network participants.
- There is no direct connection from every participant to every other participant, but since every participant maintains **multiple connections**, data interchanged between two arbitrary points can be relayed via a multitude of routes.
- Distributed networks are thus much **more resilient** against system breakdowns and data loss as they can even compensate the simultaneous failure of multiple nodes.
- As blockchains are maintained by a distributed (peer-to-peer) network where every peer holds a **complete copy of the blockchain's current state**, the system has no single point of failure.

Centralized networks with a single point of failure:

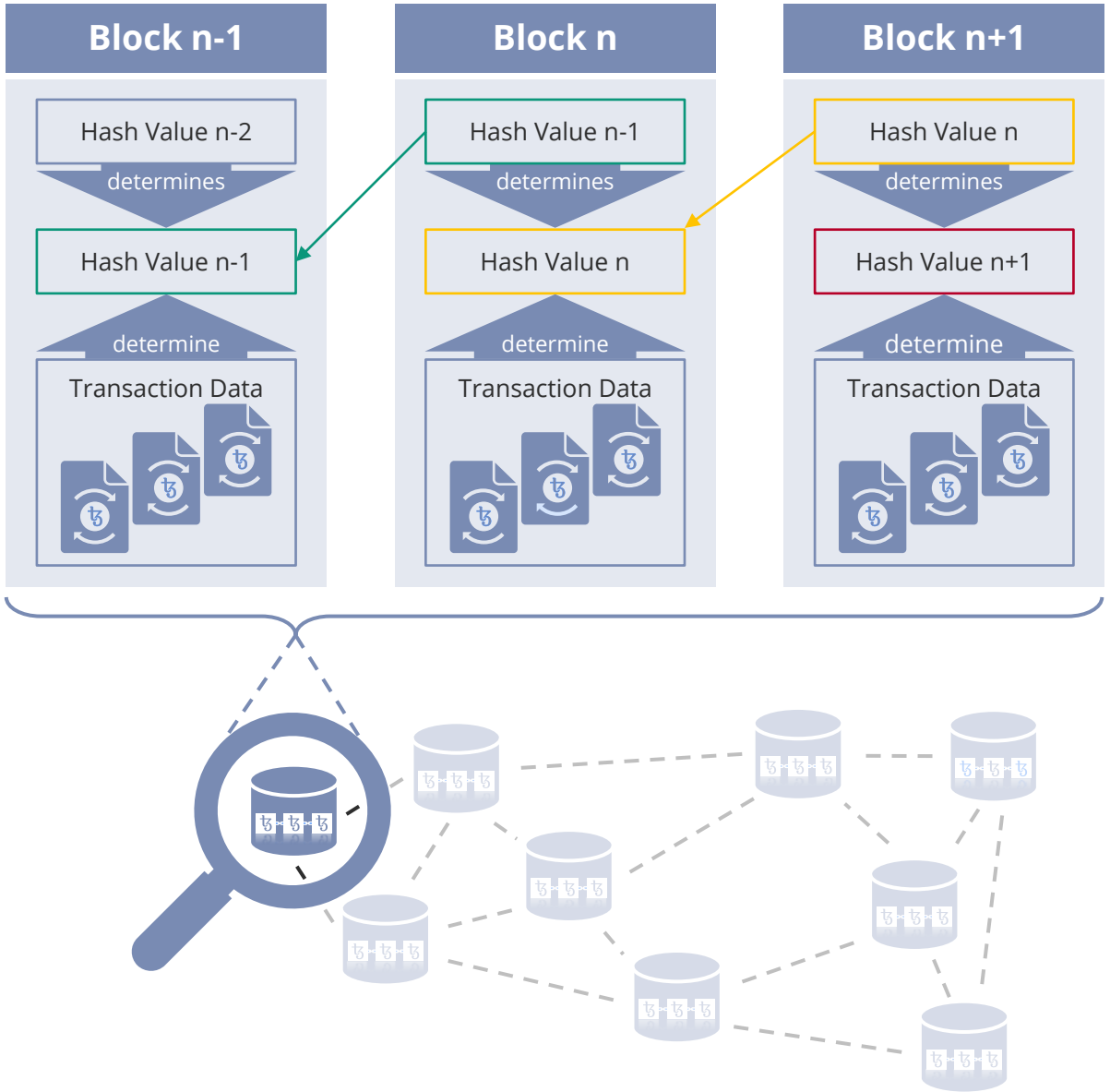


Distributed networks with no single point of failure:



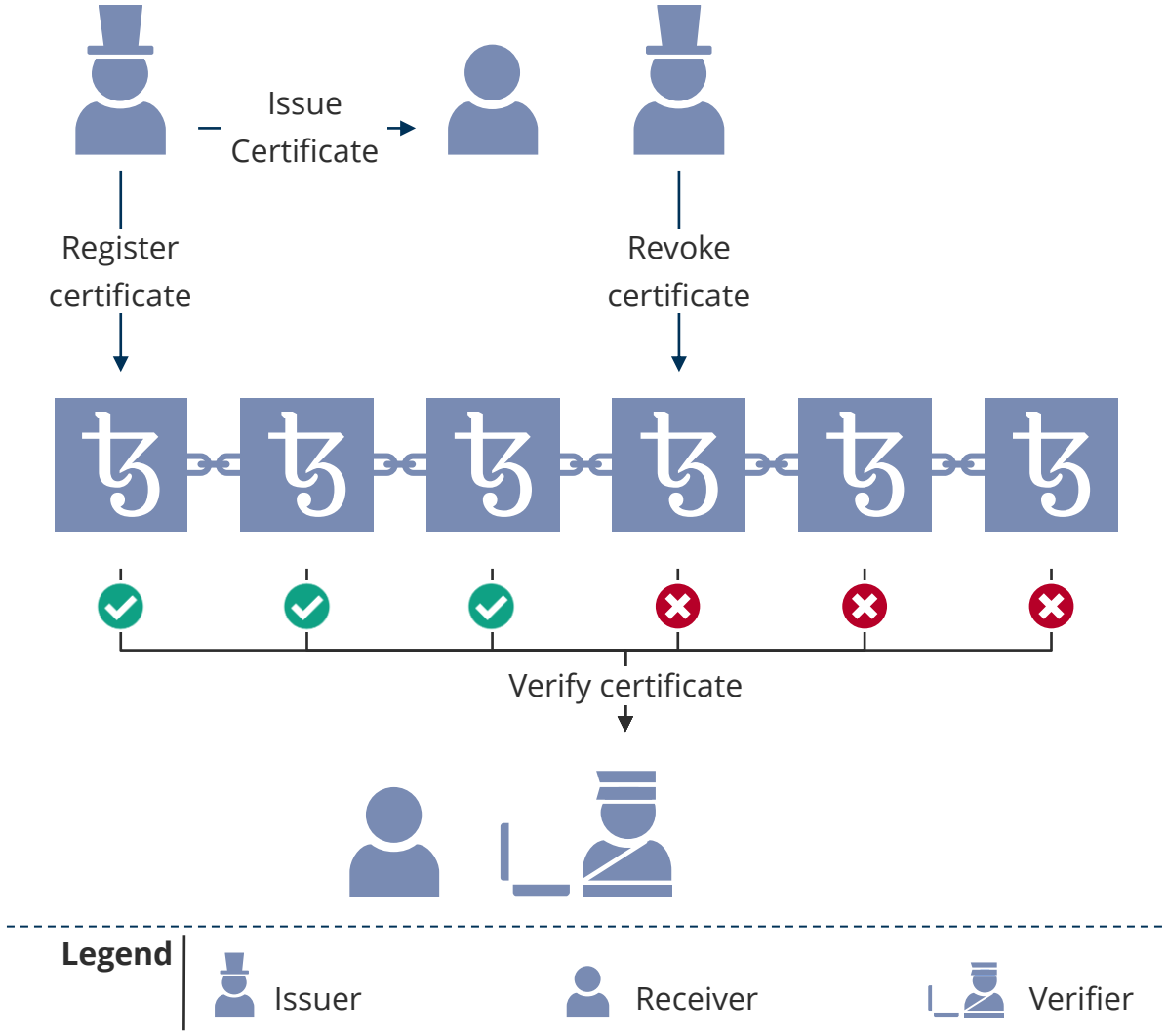
Why blockchain could be a building block for your problem's solution: Immutability/Traceability

- As the name indicates, a blockchain is a **data structure** that essentially **groups data into blocks** (e.g. transaction data) which are **chained together** in a way that preserves **chronological order** and **prevents editing** (i.e. manipulating) any data that has entered the blockchain:
  - Blocks are **timestamped** and new blocks of data are always **appended** at the end ("head") of the chain.
  - A **hash function** creates a concise representation of the block's data called **hash value** often described as the **fingerprint** of the data as it identifies the data and changes drastically if the input is modified in the slightest way.
  - The **hash value of the previous block** is always included in the calculation of the current block's hash value, thereby **linking** the blocks. If data in a previous block were manipulated, changing its hash representation and thus **breaking the chain**.
- **Replicating** this special kind of database across the nodes of a **Peer-2-Peer-Network** (P2P-Network) and finding **consensus** on its "right" state in an attack resistant way makes the blockchain a practically **immutable ledger**.
- Blockchain solutions are therefore destined for applications that require immutability for **traceability/auditability** reasons.
- These are typically found in environments with **strict regulatory requirements** and **complex multi-agent ecosystems**.



Why blockchain could be a building block for your problem's solution: Revocable certificates

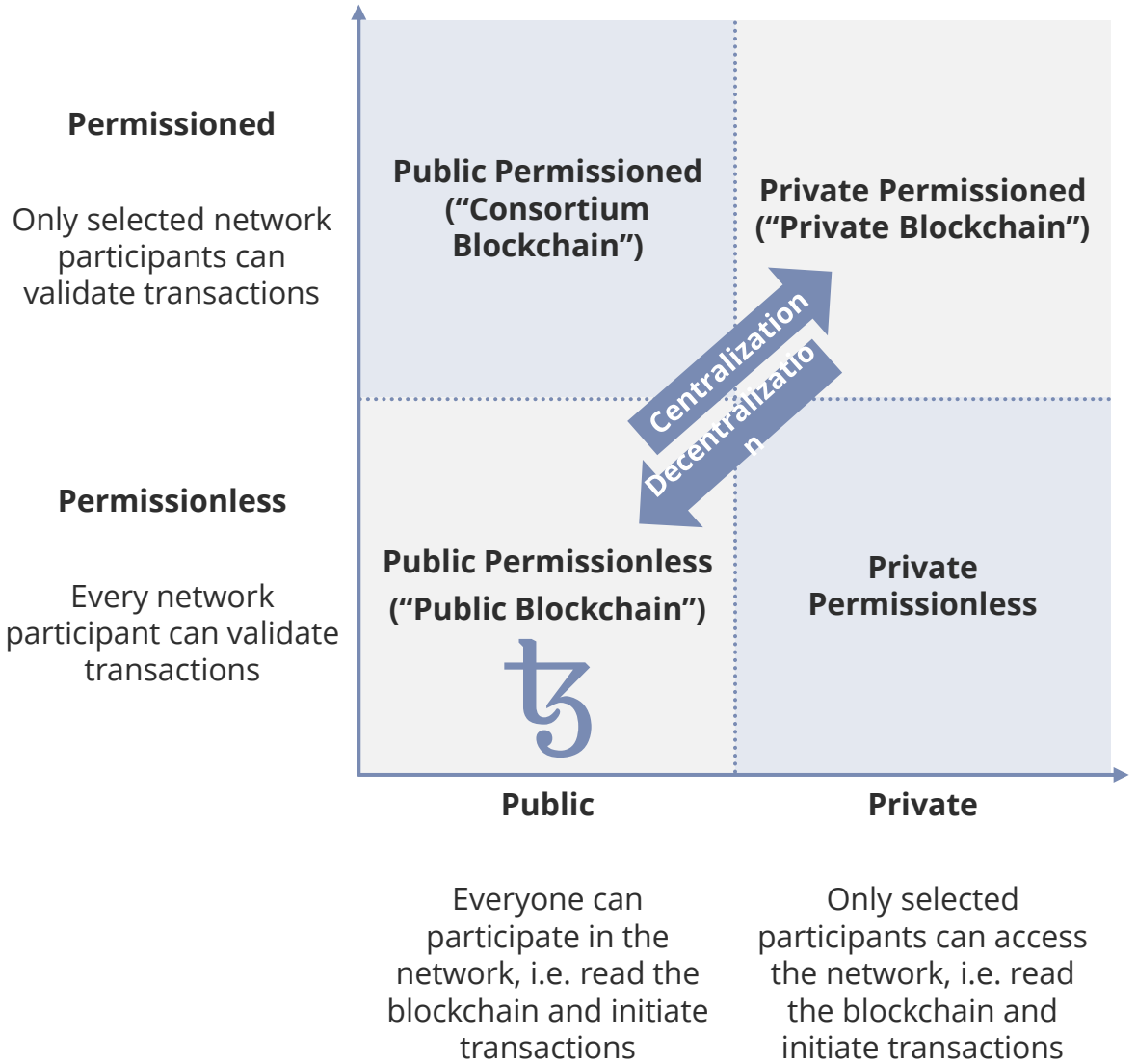
- As an immutable, independent and public ledger, a blockchain enables **revocable certificates/credentials**.
- For certificates to have a value, they must be **trustworthy**.
- In particular, it must be (almost) **impossible to forge or edit** the certificate **even for its issuer**.
- If a credential changes, the existing certificate has to be **revoked** (i.e. declared invalid) and – if applicable – a new certificate has to be issued.
- The blockchain's immutability makes certificates **forgery resistant** and – in case of public blockchains – its accessibility enables it to serve as a **revocation registry**.
- To avoid making **sensitive information** publicly available, the certificate issuer can hand out the certificate to its recipient and just **register a hash-representation** (a representation, that is unambiguously linked to the original data but is neither human readable nor allows a reproduction of the original data) on the blockchain.
- The recipient can present his human readable version of the certificate to any verifier, who can produce the hash-representation and match it with the one found on the blockchain to **verify its authenticity and validity**.
- To revoke a certificate, the issuer **references** the hash and adds the information that it's invalid – any future verification then fails.



WHY PUBLIC BLOCKCHAIN?

Just a quick reminder – what exactly is a PUBLIC blockchain?

- Although every blockchain protocol may have its unique features, there are four **archetypes** that allow a **basic classification** along two binary dimensions:
  - The **public vs. private dimension** determines, whether the network is in general open for everyone (→ public) and thus who can **read the blockchain** and **initiate transactions** or whether its access in restricted (→ private).
  - The **permissioned vs. permissionless dimension** determines, if every network participant (as allowed by the public vs. private dimension) can take part in the **validation of transactions** (→ permissionless) or if transaction validation is restricted to a selected subset (→ permissioned).
- Public permissionless types are also called "**public blockchains**", private permissioned set-ups are called "**private blockchains**" and public permissioned combinations are known as "**consortium blockchains**".
- While all types may have their **use cases** (e.g. private permissionless for voting), private and permissioned blockchains are more centralized (in the sense that they are less decentralized) und thus provide a lower level of security against various attack vectors while being "less immutable".
- **Tezos** is a public and permissionless blockchain that is open to everyone and only requires bakers (i.e. validators) to hold a minimum stake (i.e. amount of tokens).



Why public blockchains are the future...

PRIVATE   PUBLIC

The benefits of closed networks are limited – today's internet applications wouldn't be possible if we only had intranets!

The **fixed costs of maintaining your own network** are higher than **(variably) paying for using an existing** one, overall costs for numerous small networks are higher than for one large network!

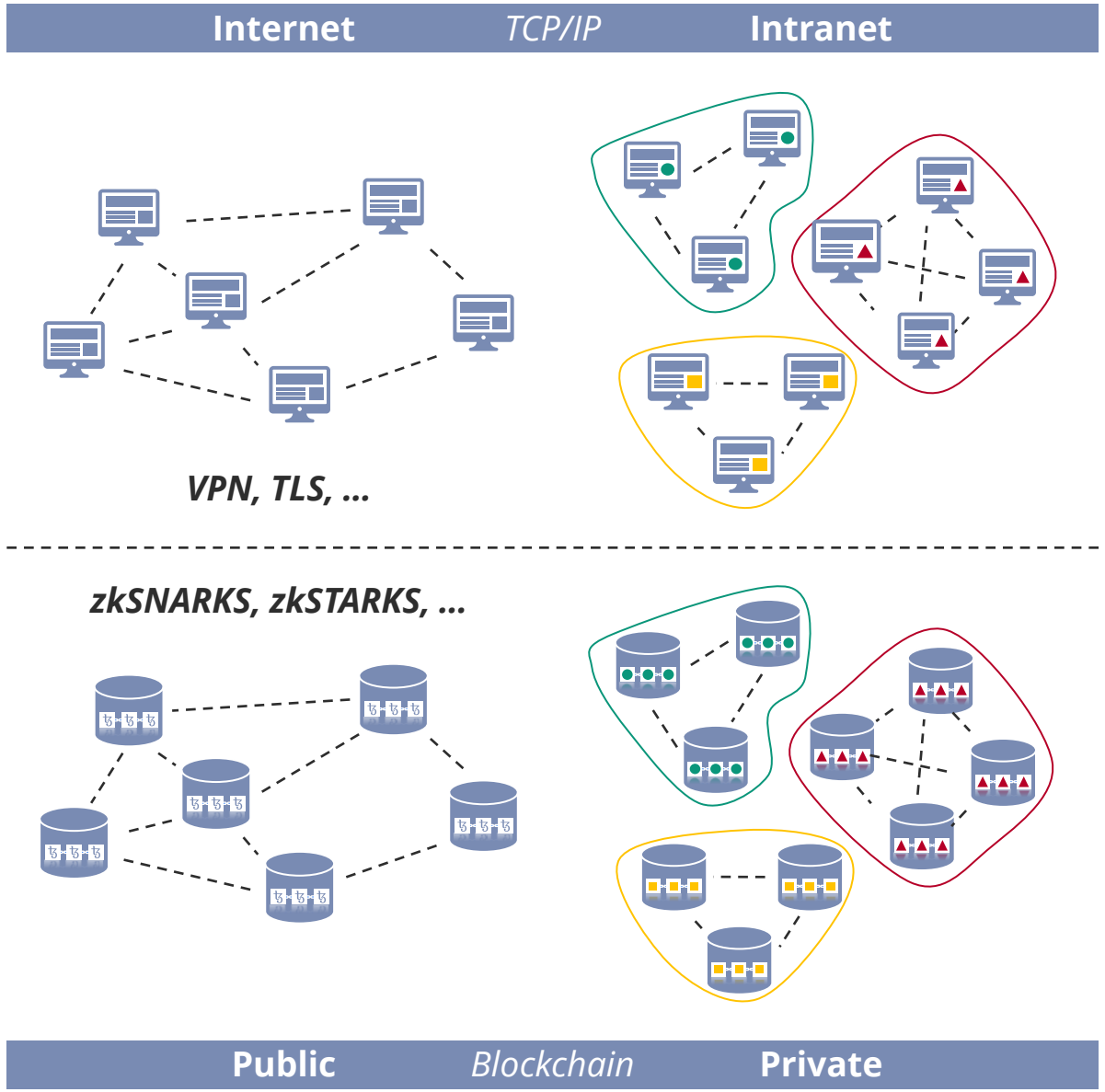
Public blockchains **tap into the wisdom of the crowds** and their **intrinsic motivation**, while private blockchains are solely developed by paid and thus extrinsic motivated people.

Public blockchains provide much **higher immutability/security against manipulation** through stronger decentralization!

Current USPs of private blockchains (like privacy, scalability) will erode with further technological developments that are already under way (e.g. zkSNARKS, Layer 2)

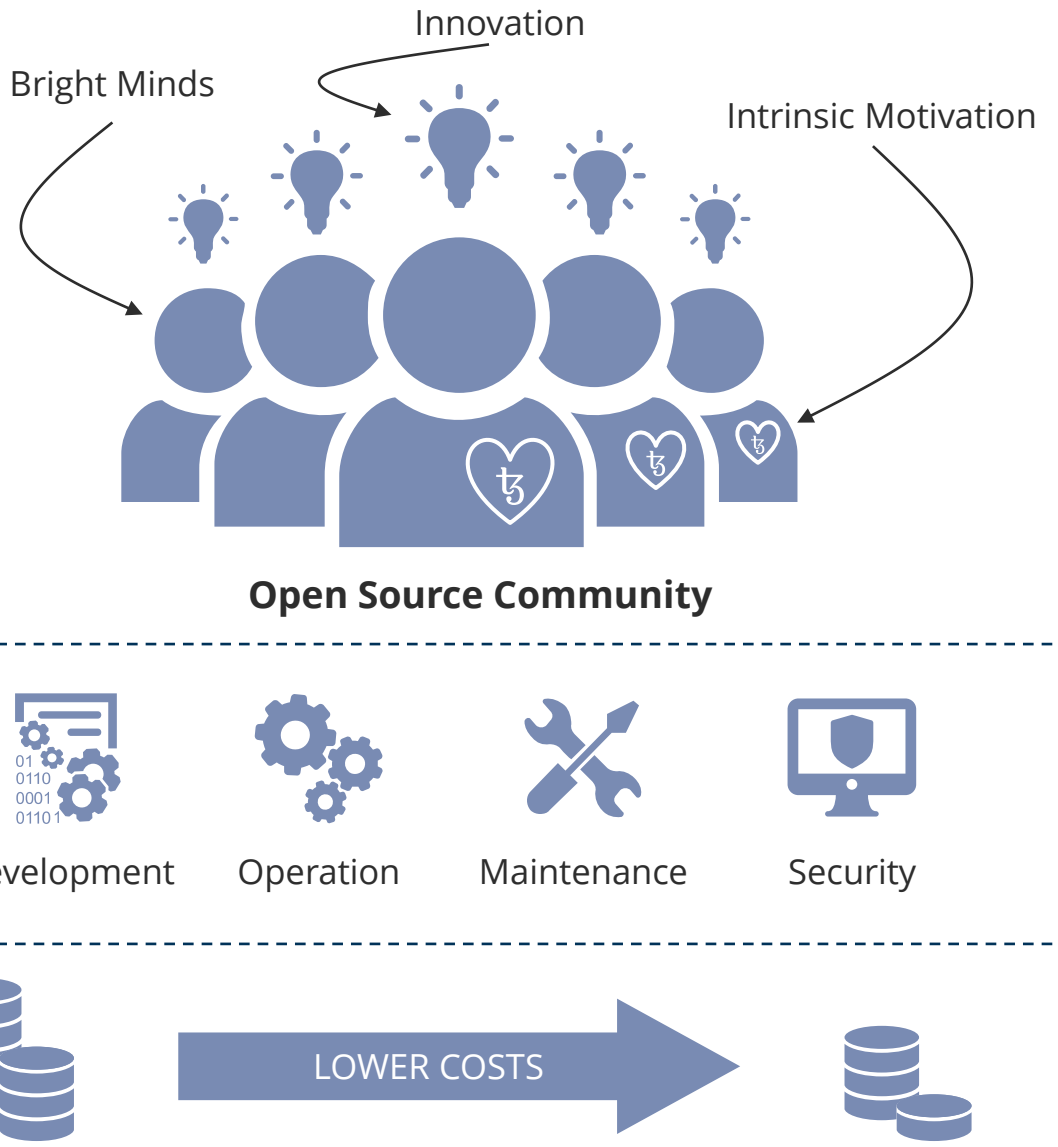
Why public blockchains are the future: The internet analogy

- In economic theory, there is a big chapter about **network effects**. It covers everything from **decreasing marginal costs** through **economies of scale** to **increased individual utility** with growing network size.
- Imagine a telephone network that consists of only two endpoints. Its users' **benefits would be very limited** as they could only call the other endpoint but no one else (for selected use cases – say the "red telephones" between the Pentagon and the Kremlin during the Cuba Crisis – there could still be significant utility).
- Now imagine a protocol such as TCP/IP that – in contrast to telephones, which are limited to synchronous verbal communication – allows a **multitude of applications** such as websites, e-mail, tube video sites, IP telephony, etc.) and the effect multiplies.
- You can employ the protocol in a private environment which gets you an **intranet**, or in a public network which we call the **internet**. From which use can you extract the most value?
- While private enclaves are still necessary e.g. for company internal applications, they now tend to be built as parts of the internet that are shielded from the public through **additional privacy and security mechanisms** such as **VPNs**, **TLS**, etc. with only critical infrastructure remaining truly self-sufficient networks.
- With blockchains, it is exactly the same: a private blockchain may be adequate for a very specific use case, but **public blockchains will generate much more value in the long run**.



Why public blockchains are the future: The power of open source technology

- In addition to “classical network effects“ such as increasing marginal utility and decreasing marginal costs with a growing network, public blockchains unleash **the power of open source technology**.
- They are developed and maintained by **vibrant open source communities** that consist of some of the brightest individuals in their fields.
- Using a public blockchain thus allows you to:
  - Capture the enormous potential and **innovation capabilities** of **intrinsically motivated** communities
  - **Distribute** the burden of **development and maintenance costs** on multiple shoulders
  - Profit from **greater software security** because the openly available codebase is not only reviewed by the community itself but also vetted by third parties (like yourself) who wish to build their applications on top of it
  - **Concentrate** on your use case, the development of the corresponding **application** and its integration with the blockchain via **interfaces**.
- An important economical aspect, that is not applicable for open source technology in general, but also comes with public blockchains, is that the **costs for operating the blockchain** are carried by the network: **you pay for its use instead of its operation!**



Why public blockchains are the future: Developments in scaling and privacy technologies

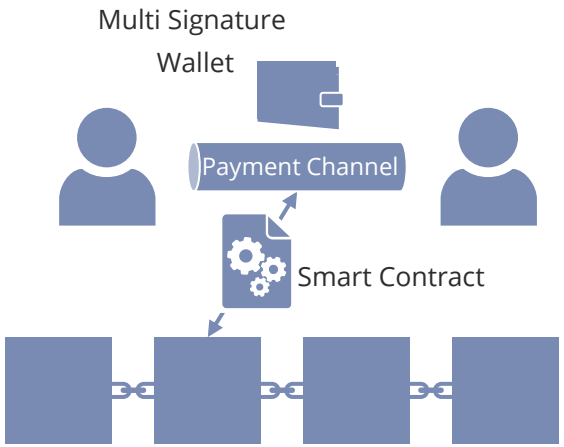
SCALING TECHNOLOGIES

- **Scalability** is one of the major points of concern of public blockchains.
- However, several so-called **layer 2** solutions (because they add a layer on top of the root blockchain) are currently being developed and tested, in order to solve scalability problems.
- One such concept are **payment channels** (as devised for the **Bitcoin Lightning Network**) between two transacting parties.
- **Ethereum Plasma** (which is also the basis for **Tezos Marigold**) on the other hand establishes a **tree of secondary (child) chains** which are essentially smaller copies of the main (parent) chain.

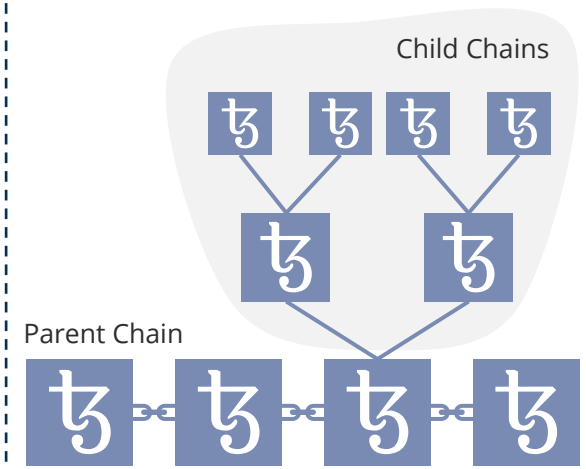
PRIVACY TECHNOLOGIES

- Developments in privacy technologies diminish **concerns about confidentiality** on public blockchains.
- **Zero-knowledge proofs** allow to prove a fact without revealing the fact itself, e.g. that a valid transaction has occurred without revealing the transaction (and its details) itself.
- Current protocols are **zkSNARKs** (zero-knowledge succinct non-interactive arguments of knowledge) which require a confidential initial set-up and are used in Zcash or **zkSTARKS** (zero-knowledge succinct transparent arguments of knowledge).

Payment Channels



Tree of Secondary Chains



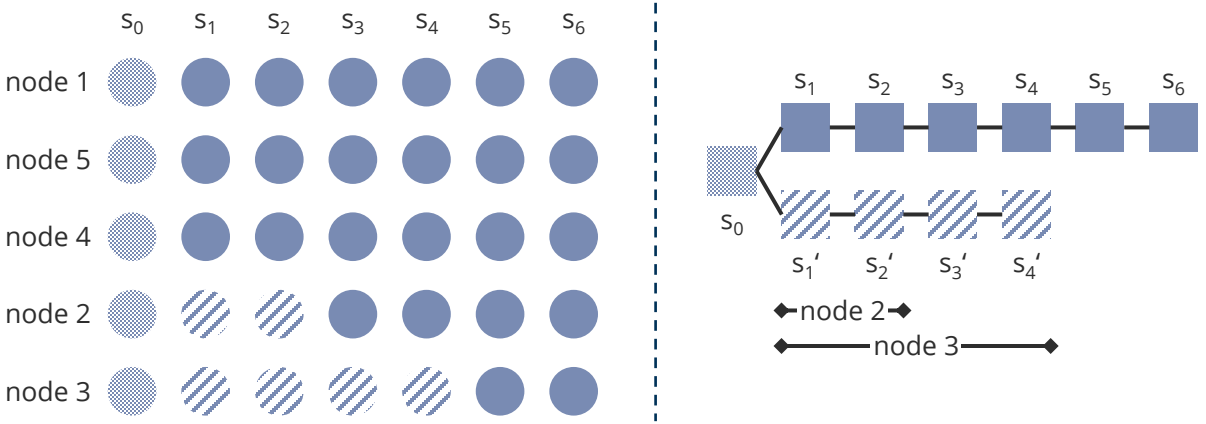
The Proof…

- *…is **complete:** every true claim will convince an honest verifier*
- *…is **sound**: a false claim will not convince an honest verifier*
- *…reveals **zero knowledge**: the proof does not leak the secret*

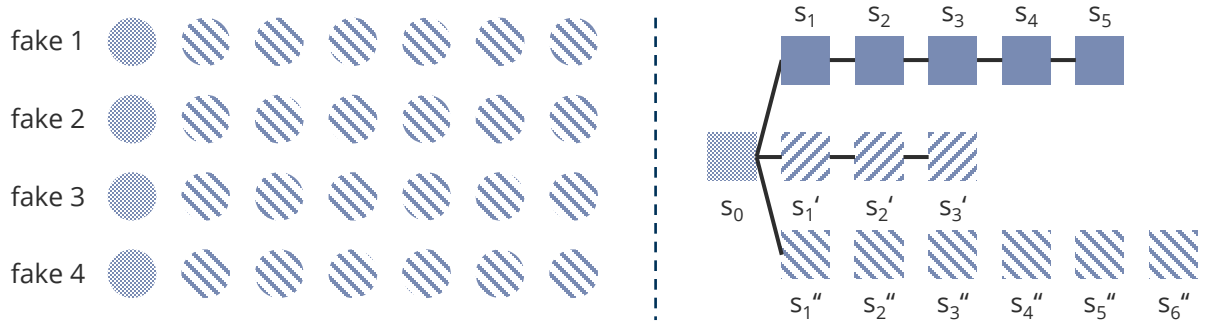
An excursion to algorithm theory: Why public blockchains require a coin/token

- In order to agree upon “one truth” in the distributed network, a **consensus protocol** is required.
- A consensus algorithm needs to possess the following **properties**:
  - **Termination (finality):** the algorithm terminates
  - **Agreement:** all nodes agree on the same value
  - **Validity:** the agreed upon value makes sense
- In **fault-tolerant distributed networks**, these properties cannot be achieved simultaneously (FLP impossibility).
- By **abandoning the termination requirement** (finality becomes probabilistic), consensus protocols for blockchains become **semi-algorithms**, resulting in the possibility of multiple “current state” versions of the blockchain across the network that evolve over time.
- In a **Sybil attack** an attacker exploits this by creating fake nodes that convince the network of a wrong version of the “current state”.
- To prevent this, **Sybil control mechanisms** such as **Proof-of-Work (PoW)**, **Proof-of-Stake (PoS)**, … (PoX) utilize economic incentives to prevent malicious behavior. This is achieved by the use of a coin (crypto currency) by:
  - **rewarding honest behavior** (rewards are generated through coin inflation)
  - **making non-compliant behavior expensive** (work/coin collateral).

Due to **non-termination**, blockchain states evolve across the network:



In a **Sybil attack**, fake nodes manipulate the „truth“:



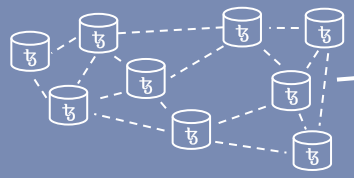
Sybil control mechanisms (PoX) prevent this through **economic incentivisation** with a coin:

WHY TEZOS?

What is Tezos? – A definition by Arthur Breitman, co-founder of Tezos

"Tezos is a **technology**...

...implemented in a **software project**



...which allows participation in a **peer-to-peer network**

...that produces a **blockchain**

...which maintains a **decentralized ledger**

...instantiating a **cryptocurrency."**

Arthur Breitman

Why Tezos is the right choice when building your blockchain solution...

Tezos is a **public and permissionless blockchain** and public blockchains are **the future**!

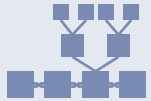
Tezos is **upgradable** through a **proven on-chain governance mechanism** and thus **built to last**!

Tezos employs **Liquid Proof-of-Stake** which is more **scalable** than Proof-of-Work and **does not consume much energy**!

Tezos is **secure** and uses the purpose-built functional language **Michelson** which allows **formal verification of smart contracts**!

Tezos is driven by a **vibrant and active community** with a lot of **academic backing**!

Powerful new features like **zkSNARKs for privacy and scalability** and a **Layer 2 solution for further scalability** are under way!



Some KPIs: Tezos in numbers...

June 2018 active since

> 880,000 created blocks	> 400 active validators
> 440,000 funded accounts	~ 5.5% average inflation
> $ 1.2 B market capitalization	3 successful amendments
~ 40 transactions per second	> 15 active projects

Key Feature: The amendment process/on-chain governance makes Tezos future-proof

The intrinsic dilemma of blockchains so far:

∞ a structure built to last for eternity **VS.** a very young and immature technology 

”everything is set in stone“

“technology still advances in quantum leaps“

So far, every new blockchain project tried to solve one or two “problems“ of previous blockchains.

But nobody is omniscient and there are still major advances in virtually every aspect of the technology.

So by the time a new blockchain goes live, it is already obsolete!

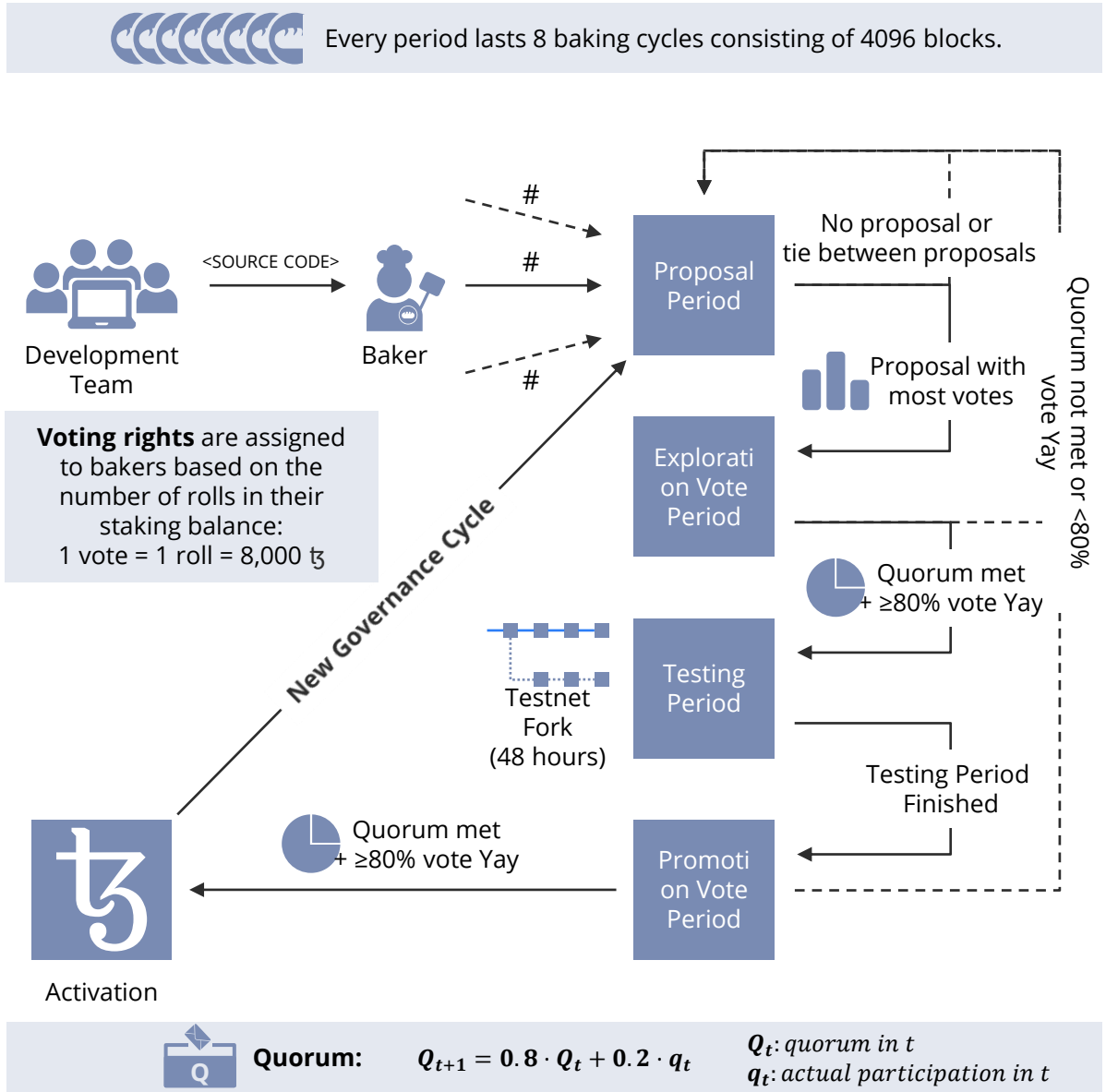
Changes to “classical” blockchains can only be achieved by forks.

Forks do have their *raison d'être*, but are very hard to coordinate in a truly decentralized ecosystem and do not favor decisions based on options’ merits.

Tezos solves this dilemma with a **built-in and proven governance mechanism** that allows **on-chain coordination** leading to **controlled upgrades to the protocol**. The Tezos blockchain can thus **evolve over time** and **adapt the best technological developments from the entire ecosystem**!

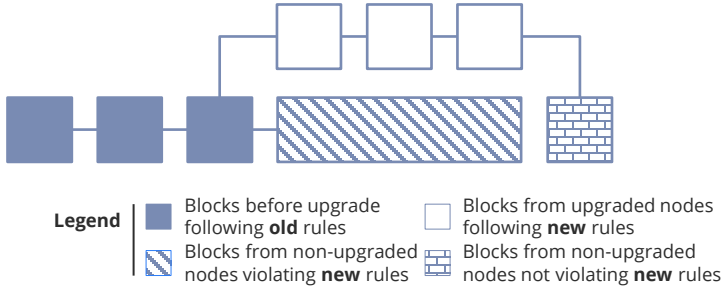
How the Tezos on-chain governance/amendment process works

- The Tezos **on-chain governance** is implemented in an **amendment process**, that consists of **four major periods**, each of which lasts **eight baking cycles** of 4,096 blocks (so 32,768 blocks per period).
- Depending on the votes, each period may end by **forwarding the process to the next period** or by **reverting to the process' outset**.
- During the **Proposal Period**, bakers can submit up to 20 amendment proposals by injecting the **hash** of the amendment's **source code**.
- If there is no proposal or a tie by the end of the period, the process reverts to its start. Otherwise, the proposal with the majority of votes proceeds to the **Exploration Vote Period.**
- Bakers can then vote, whether they wish the previously selected proposal to proceed to the **Testing Period**, which happens, if both a dynamically determined **quorum** which is calculated as a function of the previous quorum is met and a **supermajority greater than 80%** is reached. Otherwise the process is reset.
- During the **Testing Period**, a **testnet fork** which follows the amended protocol is branched and maintained parallel to the main chain for 48 hours.
- The **Promotion Vote Period** follows without another vote but is concluded by the final vote of whether the amendment should be adopted. If the quorum is met and a supermajority reached, the amendment is **activated** on the main chain. Either way, the process starts again with the next **Proposal Period**.



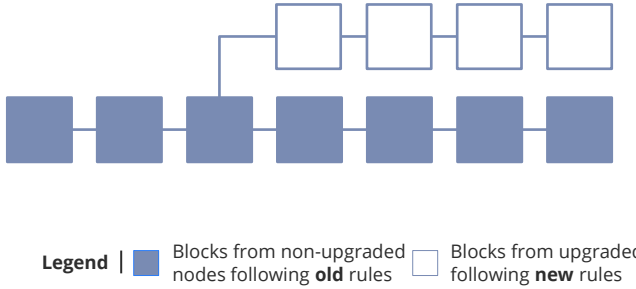
What is the difference between forks and the on-chain governance?

Soft Fork



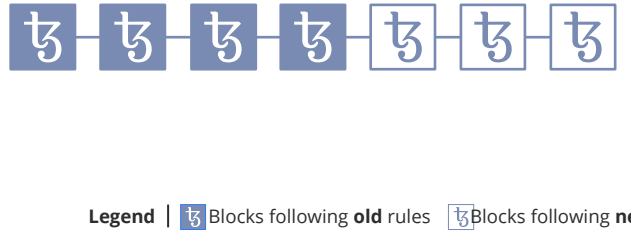
- A **soft fork** is a **backward compatible** change of the protocol adopted by a willing subset of nodes.
- Changes in the protocol **restrict the protocol rules**, so “old” nodes will accept blocks adhering to the new rules.
- Only a subset of blocks created by “old” nodes will be accepted by the network, as some will break the now stricter rules.

Hard Fork



- A **hard fork** is a change of the protocol that is **not backwards compatible**.
- Changes in the protocol **allow blocks that were not allowed before**, so “old” nodes will **not** accept blocks adhering to the new rules, effectively splitting the network.
- This **duplicates** the coins, **dividing their value** between the “old” and “new” currency, and nodes will switch to the version they think will be adopted by the majority.

On-Chain Governance



- With the **on-chain governance**, a new version of the protocol that has previously been tested in a testnet fork, is **activated by all nodes**, replacing the previous protocol.
- With the controlled governance mechanism, protocol options are **evaluated based on merits** rather than **herd behavior**.
- A “duplication” of currency is avoided, thereby preserving its value.

There are scenarios/situations in which **forks still make sense** (e.g. emergency bugfixes), but with on-chain governance, they are **not *necessary*** for protocol evolution, making them very **rare**.

Key Feature: Liquid Proof-of-Stake makes Tezos secure without wasting energy

Bitcoin's intrinsic dilemma:

network security **VS.** energy consumption

"Bitcoin mining is needed to prevent 51%-attacks"

"Bitcoin mining consumes more energy than Switzerland"

If an attacker controlled more than half of the network he would be able to "convince" the network of his "truth".

He could thus reverse transactions by rewriting the blockchain history and **double spend** his money.

This type of attack is called **51%-attack** or **Sybil attack*** as the attacker creates a lot of fake nodes to gain control.

Bitcoin seeks to prevent 51%-attacks by making them very costly.

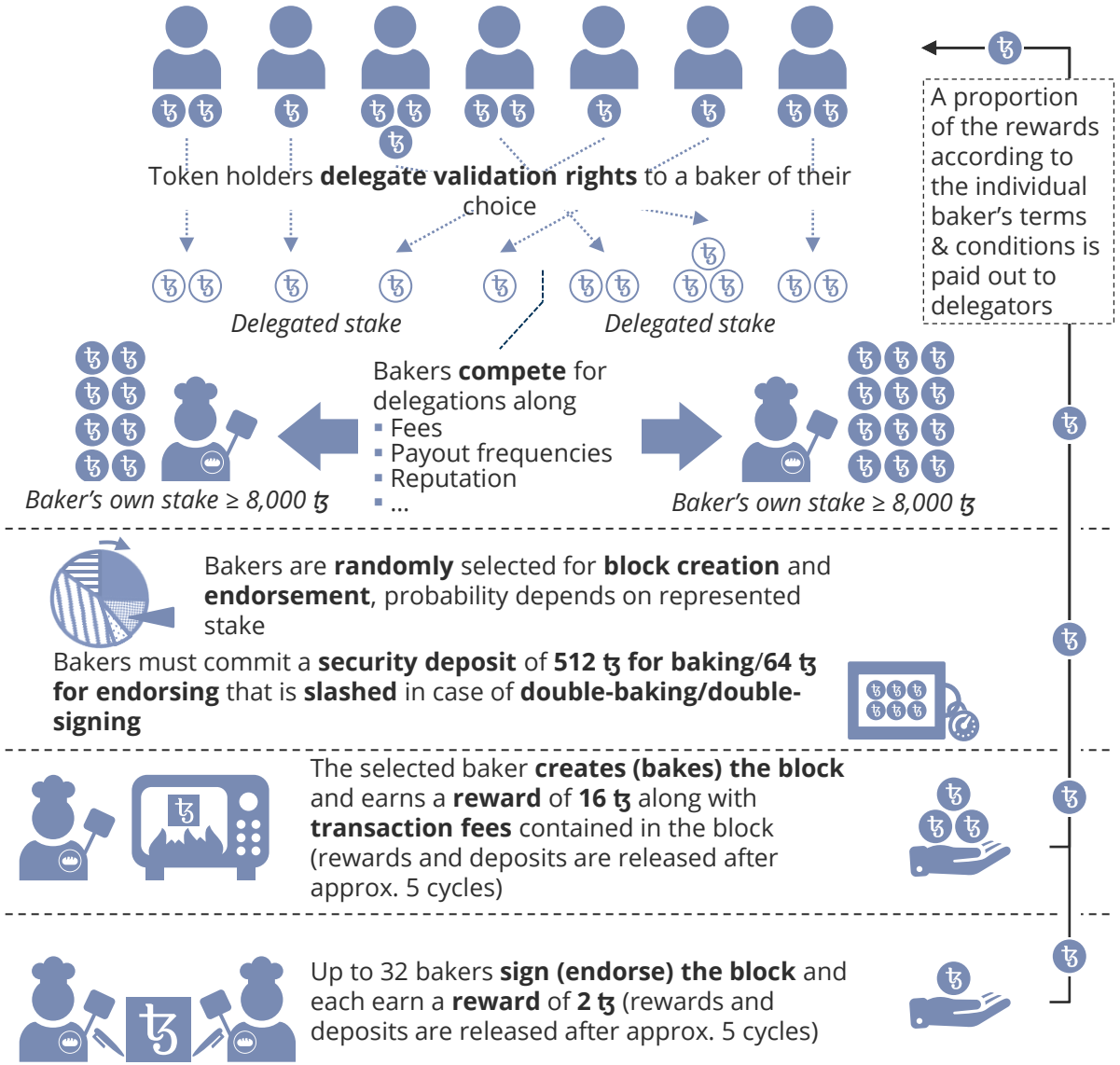
This is achieved through the Sybil control mechanism Proof-of-Work (PoW) that requires a Bitcoin miner (i.e. validator) to prove that he invested a lot of work by solving a cryptographic puzzle which consumes a lot of computing power an consequently energy.

Tezos solves this dilemma with a different kind of **Sybil control mechanism** called **Liquid Proof-of-Stake**. Like in PoW, Bakers (i.e. validators) are **extrinsically incentivized** through rewards but energy consumption is avoided by a security deposit, that can be slashed.

* Named after Sybil Dorsett who suffered from dissociative personality disorder

How Liquid Proof-of-Stake (LPoS) works

- The top design priority for **Liquid Proof-of-Stake (LPoS)** is security by true decentralization.
- While a worrying degree of concentration can be observed with mining pools for Bitcoin's **Proof-of-Work** (PoW) and **delegated Proof-of-Stake** (dPoS) as utilized in EOS and Lisk operates with a fixed and limited validator set, Tezos' LPoS strives for **low entry barriers for validators** (called bakers in Tezos).
- Baking requires downloading the baking/endorsing node, holding a **roll** (8,000 ꜩ) as well as **modest computing power** and a **reliable internet connection**.
- Delegation is **optional** for token holders and bakers **compete** for delegations with the proportion of earnings they share with their delegators, their reputation and individual terms & conditions.
- The amount of delegations a baker can accept is limited by his self-bond.
- Bakers are **randomly selected** for block creation and endorsement, with a **probability** depending on the **represented stake**. They commit a **security deposit** (512 ꜩ for block creation, 64 ꜩ for endorsement) which can be **slashed** in case of misbehavior.
- After a block is created for a **reward of 16 ꜩ** plus **transaction fees** contained in the block it is **endorsed by up to 32 bakers** for a respective reward of **2 ꜩ**. Rewards and deposits are released after approx. **5 cycles** and delegators are paid their share.



Tez – the Tezos coin and its functions

CANNOT BE DELEGATED

FINANCIAL OWNERSHIP

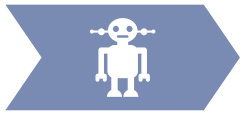
CAN BE DELEGATED

VALIDATION

VOTING

Key Feature: Formally verifiable smart contracts

The smart contract dilemma:



process automation **vs.** process integrity

”smart contracts can automate inter-party processes“

“is the smart contract doing what it should?“

Smart contracts are simple **computer programs** that can automatically be executed, when certain conditions are met.

They are **deployed on the blockchain** and thus both accessible and immutable.

Smart contracts can be used to automate the execution of the conditions of legal contracts and thus **automate inter-party processes**.

However, when engaging in a smart contract, there are two problems:

First, smart contracts are programmed by humans and humans make errors, so is the smart contract doing what it should be?
Second, the smart contract’s compiled version deployed on the blockchain is not human readable, so is it doing what its author claims?

Tezos solves this dilemma with smart contracts in the custom-made, **formally verifiable language Michelson** and a **certified compiler**. The first **minimizes the probability of errors** as formal correctness can be proven, the latter allows to **analyze the human-comprehensible version**, compiling it oneself using the certified compiler and then comparing the result to the deployed version.

The main design goals for the Michelson virtual machine were readability, security and efficiency

- **Virtual machines** that allow the **execution of smart contracts** are **attack vectors** for blockchains.
- Bugs caused by smart contracts that are known from other blockchains are:
  - Overflow (Multiple)
  - Reentrance bugs (Ethereum DAO hack)
  - Absence of libraries (Parity)
  - Combination of imperfect features (Parity)
  - Honeypots
- All these bugs were possible due to **design failures**. The Michelson VM was **custom-made** to avoid bugs and allow **formal verification** (mathematical proof).
- Next to **security**, the Michelson **design goals** were:
  - **Readability**: an expressive representation of the smart contract on the blockchain.
  - **Efficiency**: allowing fast contract execution and making the calculation of gas costs as easy as possible.
- Michelson is a **statically typed stack language** without variables but with **high-level primitives** (arbitrary length integers, maps, sets, lambdas and crypto primitives: hash, check signature).

	EVM	WASM	Michelson
Properties	256 integers	32/64 integers	Infinite precision integers
	No data structures	No data structures	Persistent sets, maps, lists
	Side effects	Side effects	No side effects
	Purpose made	Standard	Purpose made
Platform	Ethereum	Dfinity, EOS	Tezos

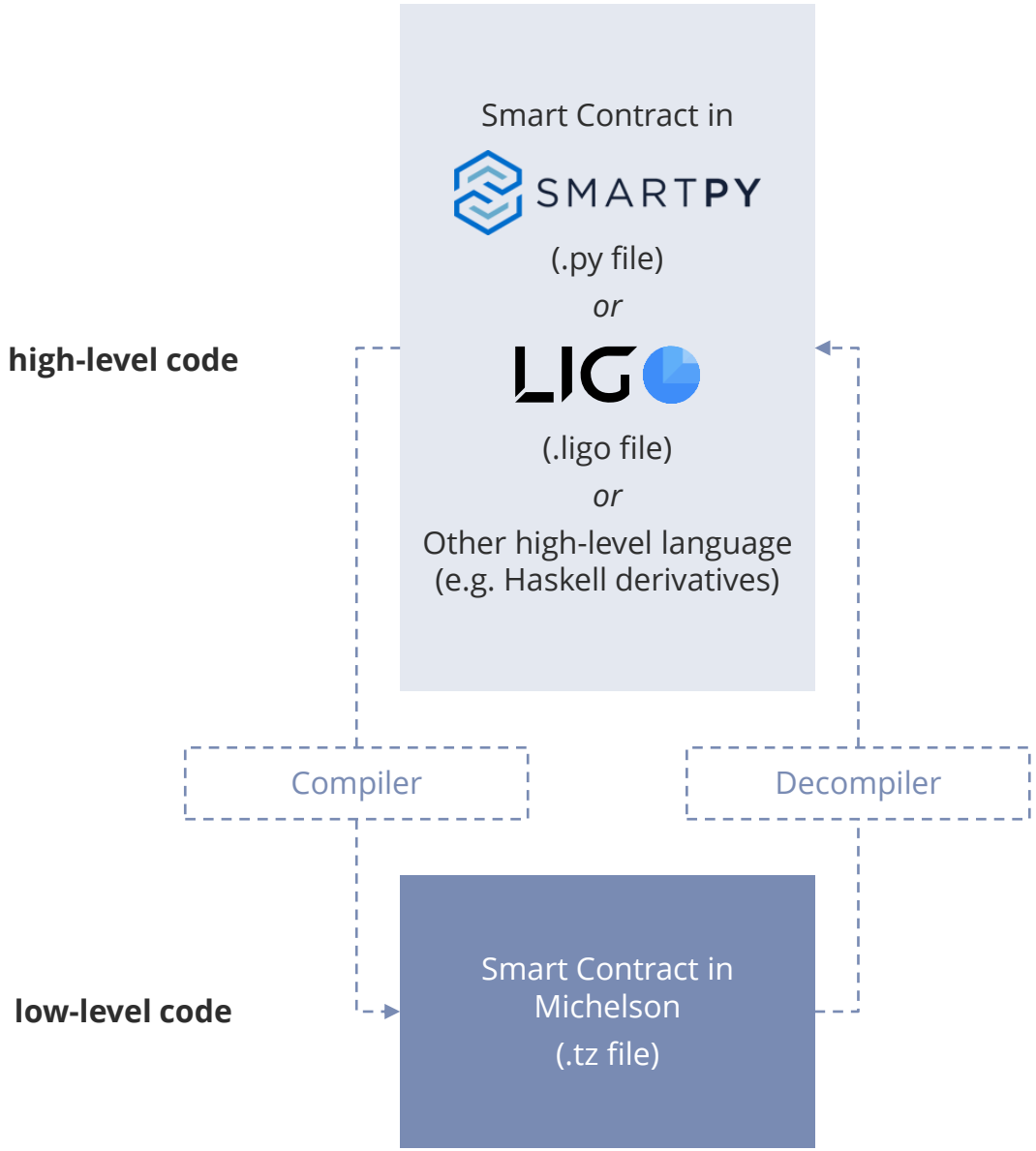
"Business logic, not protein folding."

Arthur Breitman

(about the purpose of smart contracts in Tezos)

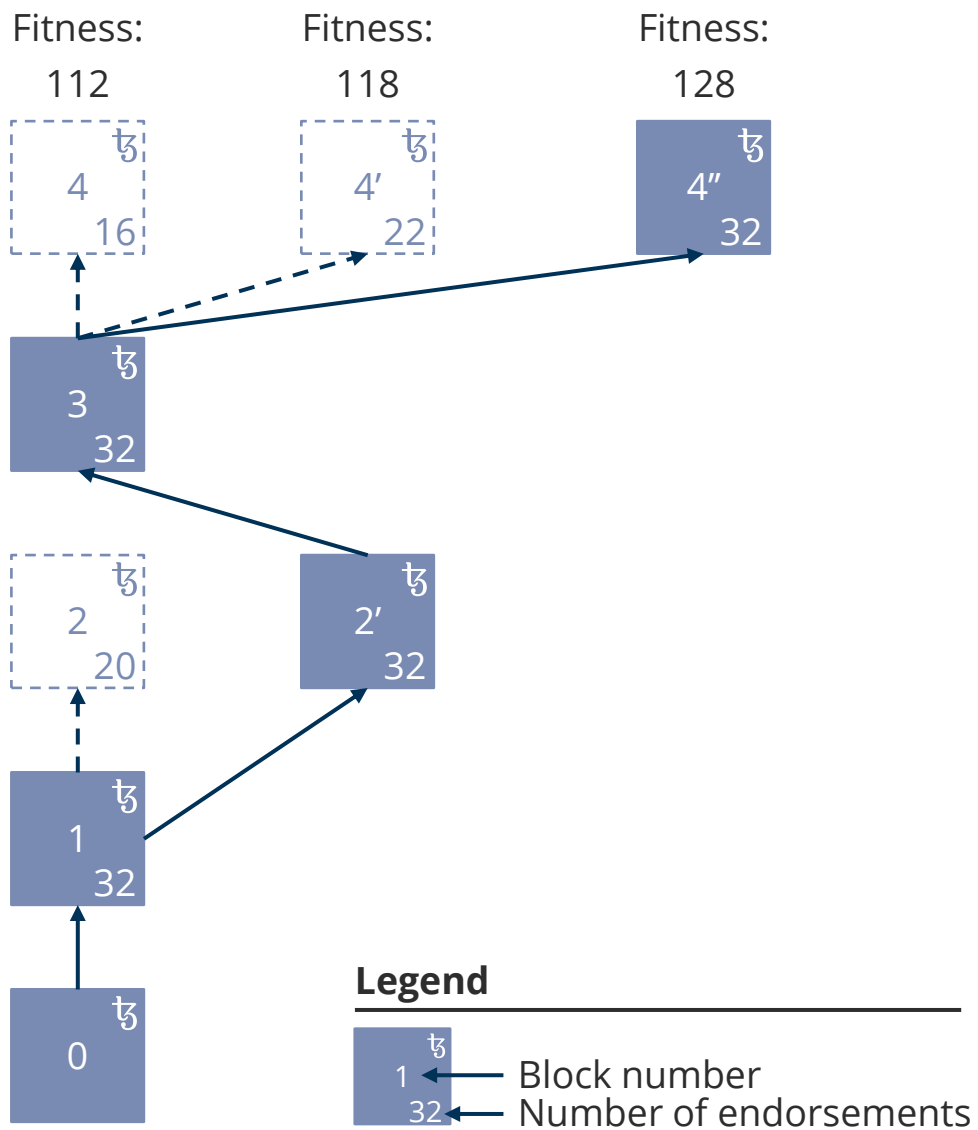
Smart contracts in high-level languages can be compiled to Michelson

- Michelson is a **compromise** between the design goal of efficiency with respect to gas accounting that would suggest an **assembly-like language** on the one hand and security and readability that would suggest a **high-level functional language** on the other hand.
- There are high-level languages such as **SmartPy**, **LIGO** (and more, e.g. Haskell derivatives) that can be **compiled to Michelson**.
- This allows to program smart contracts in a way that comes **more natural to most programmers**, e.g. LIGO offers different syntaxes that are designed to resemble Pascal (PascaLIGO), Ocaml (CameLIGO) and ReasonML (ReasonLIGO).
- It also allows people who want to engage with a smart contract deployed on Tezos to **read and understand what the contract does** (its "terms").
- To **verify**, that the deployed contract on the chain and the high-level language contract offered for inspection are indeed **equivalent**, a **certified compiler** can be used: it guarantees that the third party gets the same result when compiling the high-level code and can compare it to the low-level code on the chain.



The Tezos consensus algorithm: Emmy+

- Because they are closely entwined, **Sybil control mechanisms** like **Proof-of-Work (PoW)** or **Proof-of-Stake (PoS)** are often confused with **consensus algorithms**. However, they are merely **mechanisms to protect the consensus protocol** against **Sybil attacks**.
- The consensus algorithm is needed for the network to **agree upon a "common version of the truth"** (i.e. the right chain). It is mainly characterized by the question of **how the right chain is determined**.
- There are two main types of consensus protocols:
  - **Nakamoto Consensus:** the longest/heaviest/fittest chain is the canonical one.
  - **Byzantine Fault Tolerant Consensus:** the latest block with more than 2/3 of the validator set's signatures is appended.
- Bitcoin's consensus protocol uses Nakamoto Consensus with the longest chain criterion as the longest chain has consumed the most work – which shows how consensus protocol and Sybil control mechanism are mutually dependent.
- The **Tezos consensus protocol** is called **Emmy** and since the Babylon amendment the refined version **Emmy+** is in place.
- Emmy+ uses Nakamoto Consensus with the **fittest chain criterion** where **fitness is determined by the number of endorsements** (signatures from endorsing validators) contained in the chain. Up to 32 bakers (validators) can endorse a block.



Tezos utilizes two types of consensus for different purposes

Consensus on chain state

Achieved through consensus mechanism Emmy+

Necessary preconditions:

- "Healthy" network, i.e. diverse validator set
- Low entry barriers for new bakers
- Sybil control mechanism (LPoS)

Implications:

- Secure consensus on canonical chain

Consensus on technology evolution

Achieved through on-chain governance/ amendment process ("change management")

Necessary preconditions:

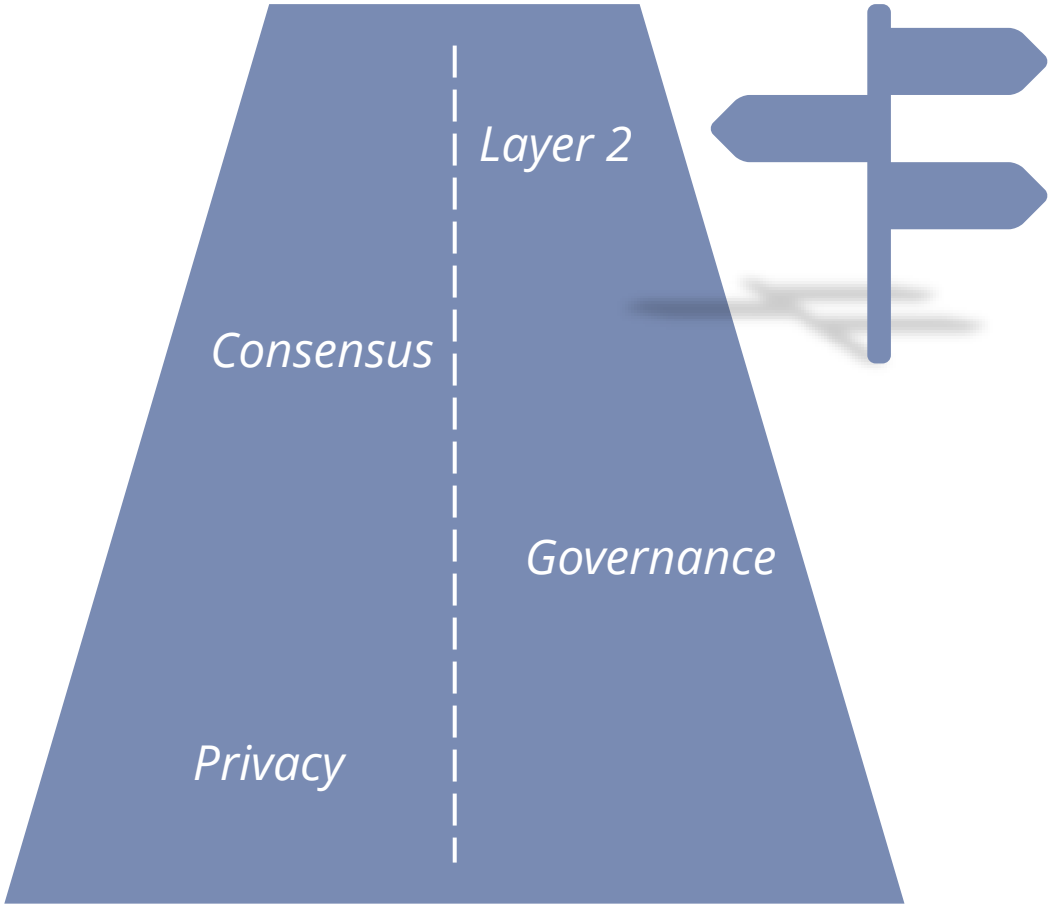
- Tools
- Discussion forums
- Transparency
- Communication
- participation

Implications:

- Community can vote on development that you do not favor
- Drastic reduction of hard fork probability
- No „asset duplication problem" through hard forks

Is there a development roadmap for Tezos?

- For people or organizations thinking of **building applications on top of Tezos**, it is of course of interest **which developments can be expected in the future**.
- As Tezos is a **truly decentralized** project, there is **no central governing body** that decides on its development, so there is no **roadmap** as such.
- Developments are implemented through the **amendment process** and thus **governed by the community** through **on-chain governance**.
- However, no development simply materializes out of nowhere. All the greater developments are discussed in the community.
- Some of the developments that can be expected in the future are:
  - Privacy preserving transactions through zkSNARKS
  - Introduction of an alternative consensus algorithm (Tenderbake, Avalanche)
  - Scalability through Sharding
  - Scalability through Layer 2 solutions (Marigold)
  - Improvements to randomness (Publicly Verifiable Secret Sharing – PVSS, Verifiable Delay Function – VDF)
  - Improvements to the amendment process itself like
  - Constitutionalism and Futarchy



The self-amendment process works: 3 successful amendments and counting...

- The self-amendment process through Tezos' on-chain governance **works**!
- So far, there have been **3 amendments** that went live after the proposals successfully ran through the voting process.
- One proposal that got voted to the Exploration Period, **Brest A**, did not make it to the Testing Period, reverting the process to Proposal.
- A **period** within the amendment process takes 8 baking **cycles** with a cycle consisting of 4,096 blocks. As blocktime varies slightly but has a lower bound of 1 minute, a period takes at least 22,76 days or **roughly 3 weeks**.
- Accordingly **a full iteration of the amendment process** running through *Proposal*, *Exploration*, *Testing* and *Promotion Period* takes at least 91,02 days or roughly **13 weeks / 3 months**.
- Tezos Agora (agora like the central festival, assembly and market place of cities in ancient Greece) allows to:
  - **Browse all periods that occurred to far** with respect to proposals, their description, voting outcomes, etc.
  - Discuss current and future proposals in the Tezos Agora Forum.
- The Tezos community follows the **convention** of naming amendment proposals with **city names** in **alphabetical order**. With the last successfully activated proposal being Carthage, the next proposal should thus be named after a city starting with "D".

A thens

05/30/2019

Participation: 84.35 %
In Favor: 99.89 %

Increase gas limit per block, reduce roll size from 10,000 ꜩ to 8,000 ꜩ

B abylon

10/18/2019

Participation: 83.46 %
In Favor: 84.53 %

Emmy+, delegable tz1 addresses, Michelson upgrades, hardened governance

C arthage

03/05/2020

Participation: 72.05 %
In Favor: 99.61 %

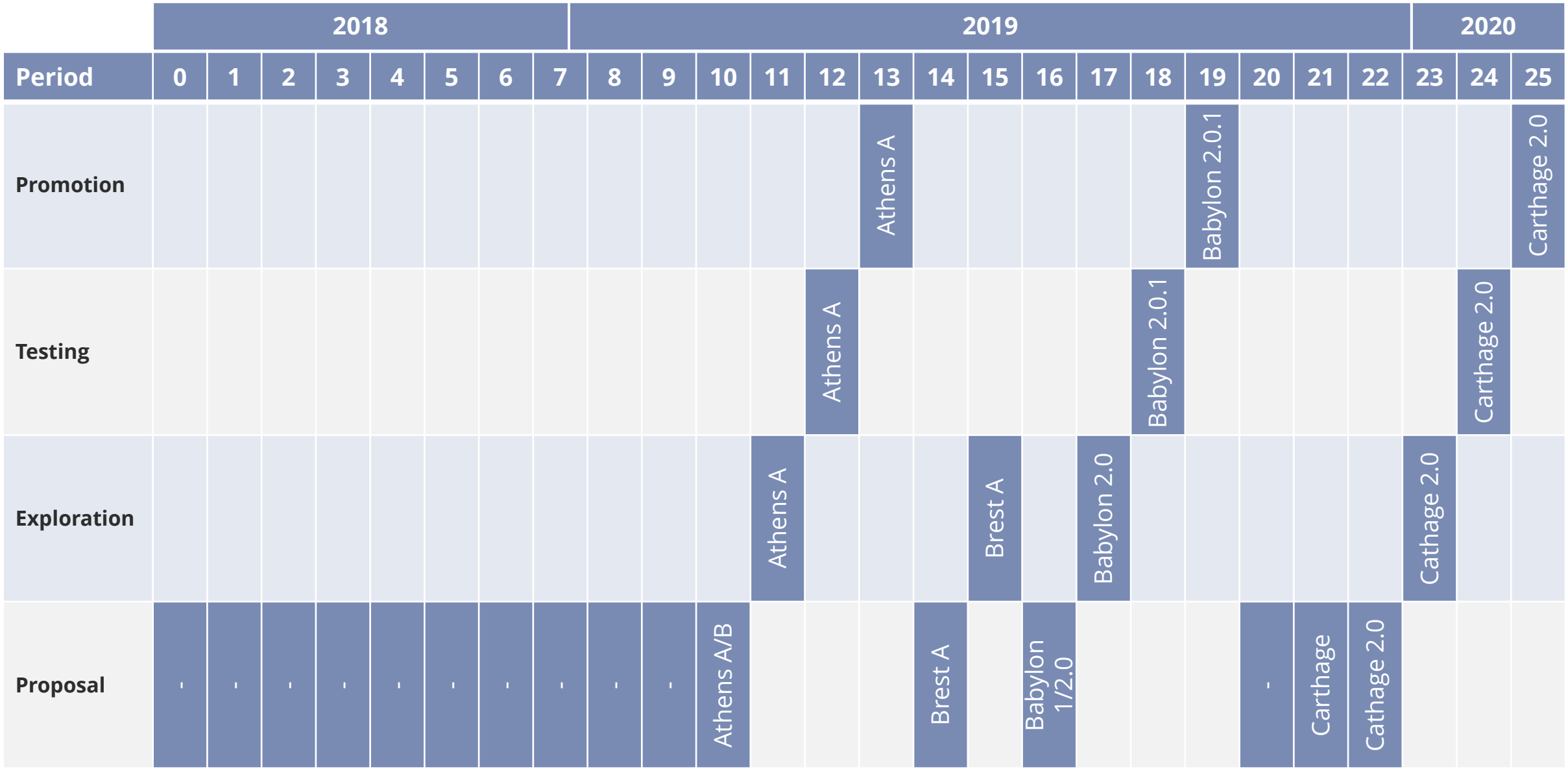
Increase gas limit per block and operation, improve formula for baking and endorsing rewards

D ...?

Legend

- *Date of Activation*
- *KPIs from Promotion Period*
- *Description of Amendment*

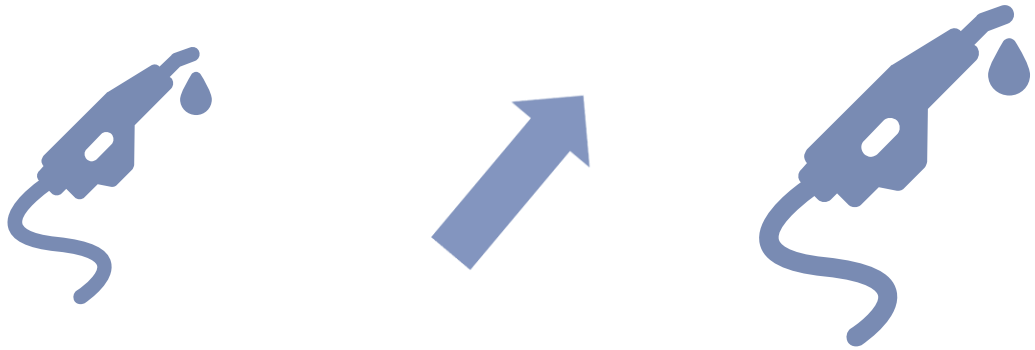
An overview of Tezos' self-amendment history so far...



Athens – the first amendment to the Tezos protocol

- Athens is the **first successful amendment** to the Tezos protocol through the on-chain governance mechanism.
- The proposal was developed and brought forward by **Nomadic Labs** in the 10th period and activated on 05/30/2019.
- Its main goal was to introduce a sensible yet simple amendment in order to **prove the on-chain governance mechanism's viability**.
- With Athens A and Athens B, two different proposals entered the **Proposal Period**, Athens B proposing a subset of Athens A's changes:
  - **Athens A:** Increase the gas limit per block and reduce the roll size from 10,000 ꜩ to 8,000 ꜩ.
  - **Athens B:** Increase the gas limit per block.
- **Gas** is a measure for the computation power needed to validate a block, so the increase of the limit allows more computation steps.
- A **roll** is the minimum amount of ꜩ a Baker must hold in order to bake (i.e. validate blocks).
- With a majority of 70.3% of votes, Athens A proceeded to the **Exploration Period**, got voted to the **Testing Period** and was finally activated on **05/30/2019** after receiving 99.89% of the votes at a participation rate of 84.35% in the **Promotion Period**.
- Nomadic Labs included a symbolic **invoice** of 100 ꜩ in their proposal as an example for funding proposal development.

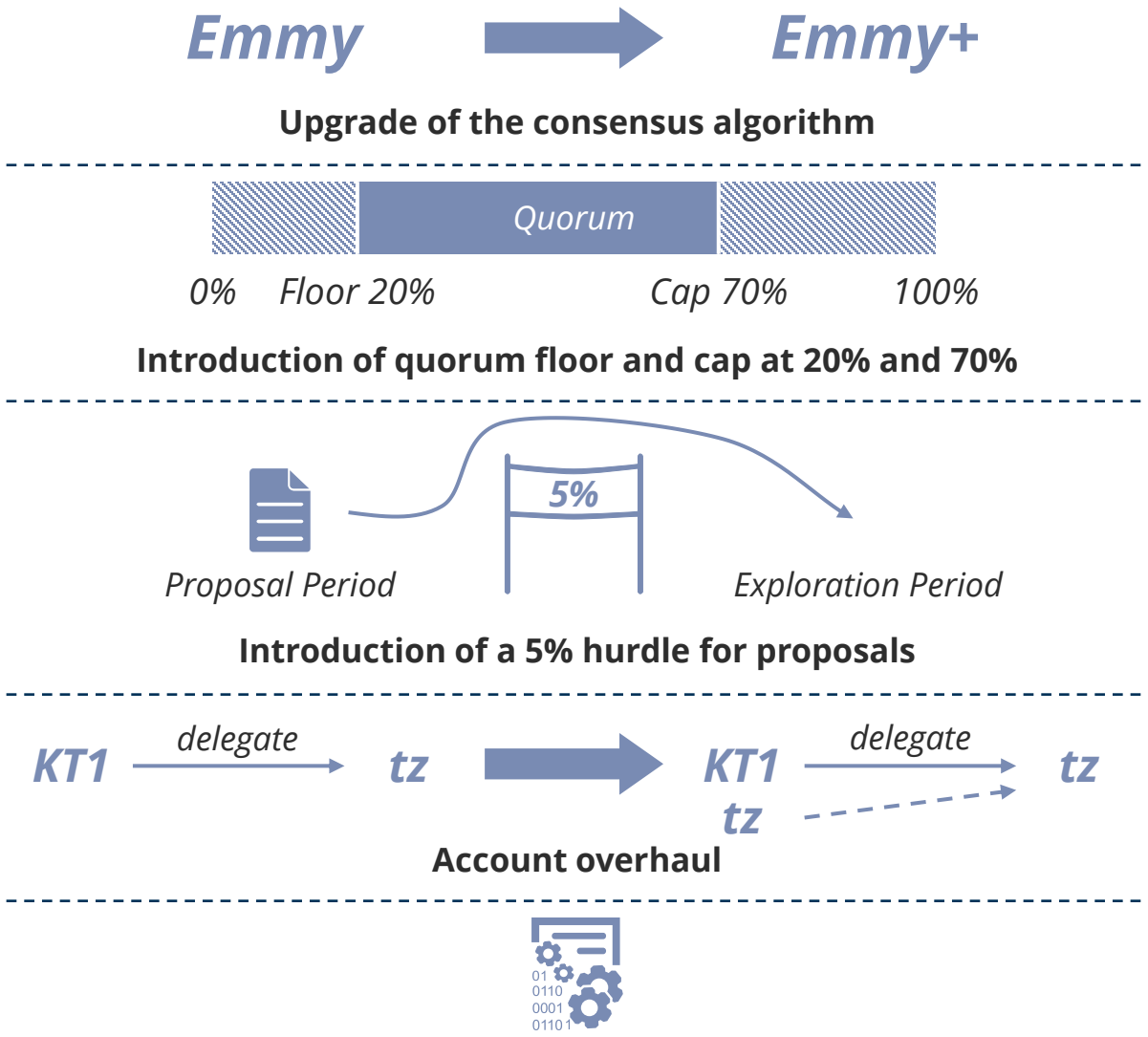
Reduction of the roll size from 10,000 ꜩ to 8,000 ꜩ



Increase of the gas limit

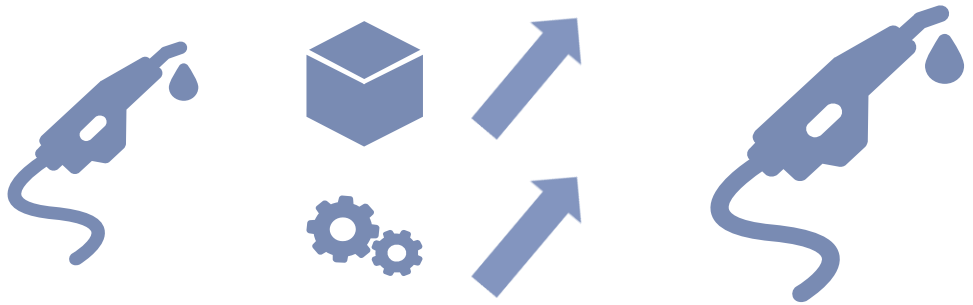
Babylon – the second amendment to the Tezos protocol

- Babylon is the **second successful amendment** to the Tezos protocol through the on-chain governance mechanism.
- It was the first amendment that introduced a **significant set of new features** and thus proved, that the amendment process not only worked (as shown by Athens) but that **amendments to large parts of the codebase** are feasible.
- Babylon is thus a **cornerstone** for Tezos to become **a blockchain that evolves over time** and **adapts the best technologies** from the entire ecosystem.
- It was jointly developed by [Nomadic Labs] and [Cryptium Labs] with contributions from **Marigold** and invoiced with 500 ꜩ.
- Babylon brought the following changes:
  - An **upgrade of the consensus algorithm** Emmy to the more robust version **Emmy+**.
  - A **quorum floor** was set at **20%** and a **quorum cap** at **70%**.
  - A proposal now requires a **minimum of 5% support** to proceed to the Exploration Period.
  - Introduction of a clear distinction between **delegable tz1, tz2 and tz3 accounts** and **KT1 accounts for smart contracts**.
  - **New Michelson features** such as the possibility for multiple **multiple big_maps** and **entrypoints** to assist smart contract developers and designers of higher-level languages.



Carthage – the third amendment to the Tezos protocol

- Carthage is the **third successful amendment** to the Tezos protocol through the on-chain governance mechanism and was **activated on 03/05/2020**.
- It was jointly developed by **Nomadic Labs** and **Cryptium Labs** and did not contain an invoice.
- The proposal was nicknamed the **housekeeping proposal** as it focused on code clean-up, optimizations and minor fixes instead of introducing significant new features.
- Noteworthy changes brought in with Carthage are:
  - Increase of the **gas limit per operation** from 800,000 to 1,040,000.
  - Increase of the **gas limit per block** from 8,000,000 to 10,400,000.
  - Adaption of the **formula for calculating baking and endorsing rewards** to be linear in the number of endorsements (replacing a step function) and to be more resistant to certain types of attacks.
  - Minor **improvements to Michelson**.

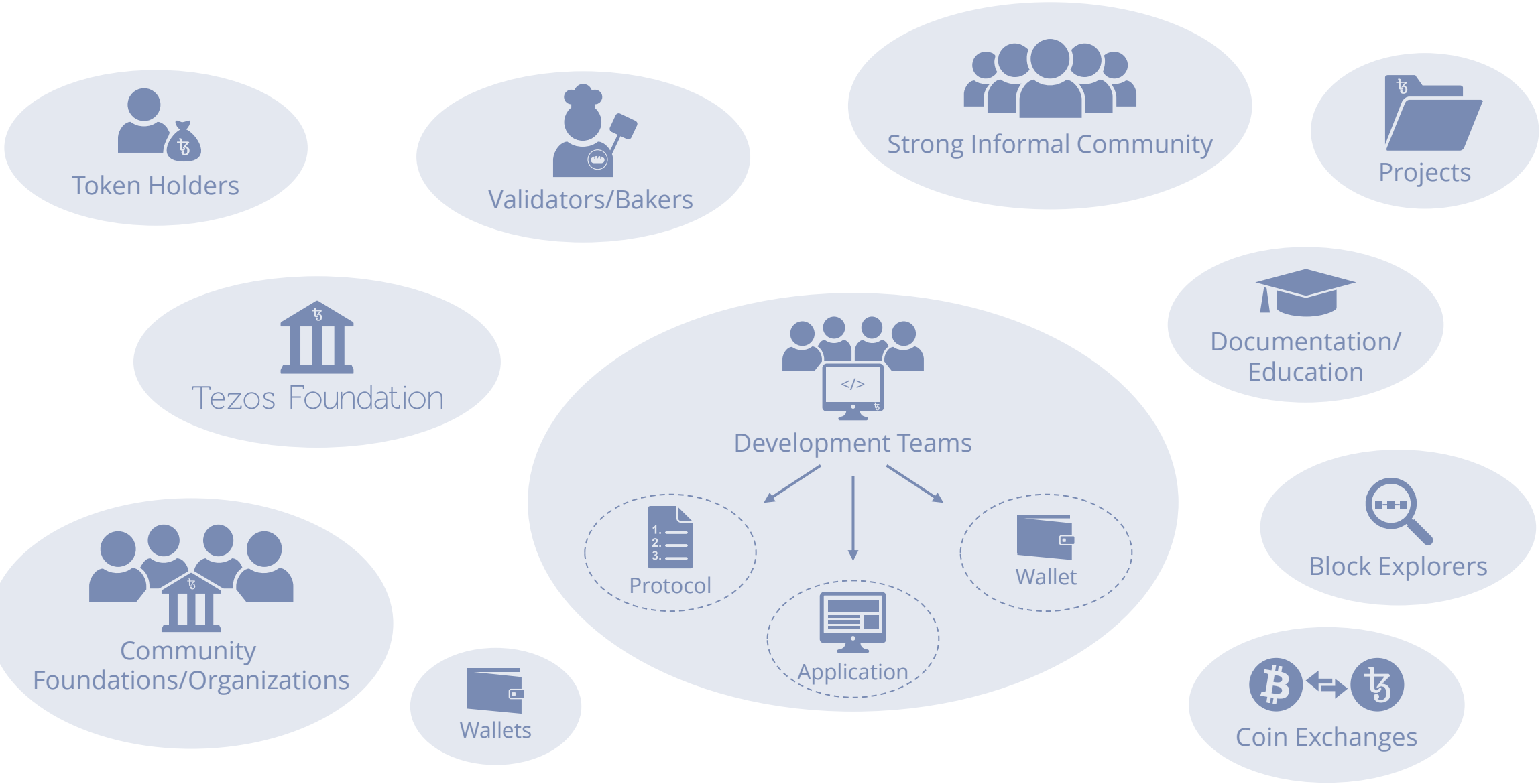


Increase of the gas limit per operation to 1,040,000
Increase of the gas limit per block to 10,400,000

Adaption of the reward calculation formula

Improvements to Michelson

The Tezos ecosystem unraveled: the different roles and actors that constitute Tezos



What is the Tezos foundation and what is its mission?

Our Vision

We seek to **empower** persons and entities from all over the world to create **a robust and decentralized digital commonwealth**.

Our Mission

We believe Tezos will **fuel social, political, and economic innovation** on a global scale. Our core mission is to **support the Tezos protocol and ecosystem** in service of this goal.

Our Strategy

We **deploy resources** to help **facilitate the advancement of the Tezos protocol** and **growth of the Tezos ecosystem**.

Tezos is a distributed, peer-to-peer, permissionless network. **No single entity owns, manages, or controls "Tezos."** Understanding this paradigm is fundamental to understanding Tezos.

Tezos Foundation

As highlighted in the **Tezos position paper**, the success of any decentralized network is determined by the efforts of a **robust, diverse, and flourishing community**.

Tezos' potential rests in the hands of its **community**, and we have no doubt that the Tezos community is **among the strongest and most exceptional** in the cryptocurrency ecosystem.

A brief history of Tezos and the Tezos foundation

To support the advancement of the Tezos ecosystem, the foundation awards grants

Categories of initiatives that may be eligible for funding through grants by the Tezos Foundation (as announced for the most recent RFP):

- Applications built using Tezos smart contracts
- Tools for Tezos smart contract development
- Educational/Training Resources Covering Tezos
- Projects focusing on using Tezos in new markets
- Marketing and other initiatives to help increase awareness of Tezos and its ecosystem
- Tooling around Tez as money
- Projects which are uniquely possible on Tezos
- Other proposals for projects targeting categories not listed above that may benefit the Tezos ecosystem

A list of the grants awarded in the most recent request for proposals can be reviewed here.

GRANTMAKING PHILOSOPHY

*As the **steward of the funds** gathered during the donation period, we **support** groups in the Tezos ecosystem that actively **work to advance the project** in a variety of ways. **Grants** offer a strategic way to support other stakeholders and community members, such as **educational and research institutions**, **developers**, and **enthusiasts** from all over the world as they work to advance the project.*

Tezos Foundation

Bitcoin ≠ Blockchain

As Bitcoin was the first blockchain in the world, Bitcoin and blockchain are often confused with one another. However, they are not the same! Blockchain is a technological concept and an umbrella term for any technology following that concept. The relationship between Bitcoin and blockchain is thus better characterized as: Bitcoin ⊆ Blockchain!

Blockchain ≠ Tremendous Energy Consumption

Bitcoin uses a Proof-of-Work mechanism to prevent Sybil attacks that does consume a lot of energy. With Bitcoin still being the most prominent blockchain and media reporting comparisons of Bitcoin's energy consumption with that of countries, the impression that blockchain requires a lot of energy stuck. However, there are other Sybil control mechanisms such as Proof-of-Stake that do not require a significant amount of energy. The Tezos protocol uses Liquid Proof-of-Stake which is low energy!

(Liquid) Proof-of-Stake ≠ Consensus Algorithm

Neither Liquid Proof-of-Stake nor any other Proof-of-Stake mechanism (nor Proof-of-Work for that matter) constitutes a consensus algorithm. The PoX family is a group of mechanisms designed to protect the network and its consensus algorithm against Sybil attacks (i.e. an attacker creating a lot of fake nodes to gain control over the network). They can thus be referred to as Sybil control mechanisms and yes – they are closely entwined with consensus algorithms, but they are not the same thing!

Liquid Proof-of-Stake ≠ Delegated Proof-of-Stake

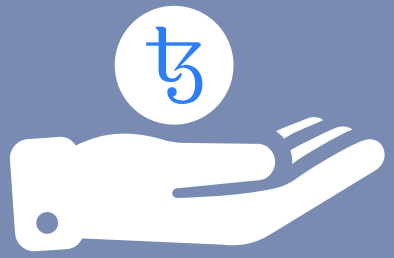
As delegation is an integral part of Liquid Proof-of-Stake (LPoS), it is often confused with Delegated Proof-of-Stake (dPoS) as known from EOS and Lisk. However, they are different concepts, with very different grades of decentralization and – respectively – network security. While the number of validators is limited to 21 in EOS and 101 in Lisk and more static, in Tezos' LPoS it is only bounded by the maximum number of rolls depending on the total supply of tez and more dynamic.

Smart Contracts = ¬Smart ∧ ¬Legal Contracts

Smart contracts are neither legally binding nor are they smart on their own accord. Smart contracts are relatively simple computer programs that are deployed on the blockchain, allow the automated execution of (inter-party) processes and thus the automated execution of contract conditions.

You Don’t Need to Use the Token in Your App

Public blockchains require a cryptocurrency to provide economic incentives for its validators to maintain the network. But just because a blockchain has a token, it does not require you to actively use that token in the application you build on top of the blockchain. You will have to pay for using it by the means of the cryptocurrency but it does not have to play a role in your use case!



Blockchain has real world applications

As blockchain is a very young yet non-trivial technology, there are a lot of failed or stuck blockchain projects and blockchain has been accused of just being a technology searching for its problem. However, blockchain does have its real world applications it's rather a matter of understanding the technology, its strengths and weaknesses and selecting those applications it is actually well-suited for!

Dipl.-Ing. Alexander F. Walser
Managing Director

Automotive Solution Center for Simulation e.V.
Curiestraße 2 | 70563 Stuttgart | Germany
Phone: +49 (0) 711 699659-21 | Fax: +49 (0) 711 699659-29
Email: hello@envited.market
Web: www.asc-s.de | www.envited.market

Get in touch with us!