

블록체인 간접 체험

Tezos Wallet Application (Built-in)

```
Aug 19 21:33:34 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.chain_1: Update current head to BMQD78RZdBWJSCMuH5anyPkZJrLj61fai9bvRcaZE2GzPi6QKYT (fitness 00::0000000001104652), same branch
Aug 19 21:33:34 - validator.chain_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.block: Block BLxNjiejt3NJKxXxCj36igEiTaqnhDyi4QWF7r1tapghH17H6MT successfully validated
Aug 19 21:33:34 - validator.block: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: switching to new head BLxNjiejt3NJKxXxCj36igEiTaqnhDyi4QWF7r1tapghH17H6MT
Aug 19 21:33:34 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:34 - validator.chain_1: Update current head to BLxNjiejt3NJKxXxCj36igEiTaqnhDyi4QWF7r1tapghH17H6MT (fitness 00::0000000001104672), same branch
Aug 19 21:33:34 - validator.chain_1: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:34-00:00
Aug 19 21:33:35 - validator.block: Block BKpBPJjkRMtfhPN1RthqCD2kprzsgVsBhgA6kv7Ah4CaBkJvZZ successfully validated
Aug 19 21:33:35 - validator.block: Pushed: 2019-08-19T12:33:34-00:00, Treated: 2019-08-19T12:33:34-00:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:35 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: switching to new head BKpBPJjkRMtfhPN1RthqCD2kprzsgVsBhgA6kv7Ah4CaBkJvZZ
Aug 19 21:33:35 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:35-00:00, Treated: 2019-08-19T12:33:35-00:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:35 - validator.chain_1: Update current head to BKpBPJjkRMtfhPN1RthqCD2kprzsgVsBhgA6kv7Ah4CaBkJvZZ (fitness 00::0000000001104693), same branch
Aug 19 21:33:35 - validator.chain_1: Pushed: 2019-08-19T12:33:35-00:00, Treated: 2019-08-19T12:33:35-00:00, Completed: 2019-08-19T12:33:35-00:00
Aug 19 21:33:36 - validator.block: Block BKx6gUL9EsUsJdeRRaZhsMabqDt8cs3pC7ReNTD3YX6Y4oDgFu8 successfully validated
Aug 19 21:33:36 - validator.block: Pushed: 2019-08-19T12:33:35-00:00, Treated: 2019-08-19T12:33:35-00:00, Completed: 2019-08-19T12:33:36-00:00
Aug 19 21:33:36 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: switching to new head BKx6gUL9EsUsJdeRRaZhsMabqDt8cs3pC7ReNTD3YX6Y4oDgFu8
Aug 19 21:33:36 - prevalidator.NetXdQprcVkp.a.Pt24m4xiPbLD_1: Pushed: 2019-08-19T12:33:36-00:00, Treated:
```

Tezos Wallet Application (TezBox)

The screenshot shows the TezBox wallet application interface. At the top, there's a header bar with three colored dots (red, yellow, green) on the left, the title "TezBox - Wallet Application" in the center, and a gear icon and a lock icon on the right.

The main area has a header "TEZBOX" with a logo on the left and settings/lock icons on the right. On the left, a sidebar titled "My Accounts" shows a "MAIN" account with address `tzlabppcukzsstG9r5p...` and a "Add Account" button. On the right, there's a circular profile picture, the word "MAIN", the balance "0.00tz", a "VIEW ON TZSCAN" button, and a QR code.

Below this, there are tabs for "Transactions", "Send", "Delegate", and "Options". The "Transactions" tab is active, showing a list of recent transactions:

Date	To	Amount	Details
08-07-2019 09:25	To tzITJwUVAWC55m...	727.999tz	⋮
17-06-2019 01:09	To tzITJwUVAWC55m...	5244.734tz	⋮
16-06-2019 16:12	To tzlUmKj7oFBeKuKi...	4397.000tz	⋮
16-06-2019 15:37	To tzlUmKj7oFBeKuKi...	1.000tz	⋮
15-06-2019 15:47	From tzIUQj6dMKpFn...	4199.997tz	⋮

At the bottom left, there's a footer with network information: "NetXdQprcvkpaWU", "level 57199", "Connected to Mainnet_004", and a URL "https://mainnet.tezrpc.me". There are also links for "Disclaimer", "Terms", and "Privacy".

Tezos Block Explorer (TzScan)

The Tezos Block Explorer

MAINNET ZERONET ALPHANET

Sponsored OCamlPro Need a development on Tezos ?

Cycle 139 | Block 571228 | 2m 08s

Search by address : Transaction / Block / Account / Level GO

tz Circulating Supply 804.44 M_{tz}

Market Cap 972 M\$

\$ Tezos USD Price 1.21 \$ ↓-1.79%

฿ Tezos BTC Price 0.00011326 BTC

\$ Volume 24h 625.8 K\$ ↓-0.12%

Block 571228 (Cycle 139)
569345 45% Est. 1d 12h 53m 573440

Latest Baker tz1RCFbB9GpA...

Period 17 : Exploration
43% 60% 100%
~ 12d 21h 56m left Threshold 72.9% Threshold 80%

Sponsored OCaml PRO Need a development on Tezos ?

Blocks

Age	Level	Baker	#Ops	Volume	Endorsement Rate	Block Prio. 0 Baked
2m 09s	571228	tz1RCFbB9GpALps...	29	1.14 K _{tz}	98%	96%
3m 09s	571227	Foundation Baker 5	25	0 _{tz}	Transactions	Originations
4m 09s	571226	tz1KfEsrtDaA1sX...	27	171 _{tz}	4 2 6 8	Delegations
					4 0	Activations
					4 3	1 6

Last 24h

Transactions	Originations	Delegations	Activations
4 2 6 8	4 0	4 3	1 6

테조스 플랫폼

Tezos , **the last** crypto currency



Tezos: A Self-Amending Crypto-Ledger Position Paper (2014)

Conclusion

We've presented issues with the existing cryptocurrencies and offered Tezos as a solution. While the irony of preventing the fragmentation of cryptocurrencies by releasing a new one does not escape us, **Tezos truly aims to be the *last* cryptocurrency.**

No matter what innovations other protocols produce, it will be possible for Tezos stakeholders to adopt these innovations. Furthermore, the ability to solve collective action problems and easily implement protocols in OCaml will make Tezos one of the most reactive cryptocurrency.

Tezos , the last crypto currency

Problems

The Protocol Fork Problem

Shortcomings of Proof-of-Work

Smart Contracts

Correctness

Contents

1	Motivation	2
1.1	The Protocol Fork Problem	3
1.1.1	Keeping Up With Innovation	3
1.1.2	Economics of Forks	4
1.2	Shortcomings of Proof-of-Work	5
1.2.1	Mining Power Concentration	5
1.2.2	Bad incentives	6
1.2.3	Cost	7
1.2.4	Control	8
1.3	Smart Contracts	8
1.4	Correctness	9

2	Abstract Blockchains	10
2.1	Three Protocols	10
2.1.1	Network Protocol	10
2.1.2	Transaction Protocol	11
2.1.3	Consensus Protocol	11
2.2	Network Shell	11

3	Proof-of-Stake	12
3.1	Is Proof-of-Stake Impossible?	12
3.2	Mitigations	13
3.2.1	Checkpoints	13
3.2.2	Statistical Detection	13
3.3	The Nothing-At-Stake Problem	14
3.4	Threat Models	14

4	Potential Developments	15
4.1	Privacy Preserving Transactions	15
4.1.1	Ring Signatures	15
4.1.2	Non Interactive Zero-knowledge Proofs of Knowledge	15
4.2	Amendment Rules	15
4.2.1	Constitutionalism	15
4.2.2	Futarchy	16
4.3	Solving Collective Action Problems	16

Tezos , **the last** crypto currency

Problems

The Protocol Fork Problem

Shortcomings of Proof-of-Work

Smart Contracts

Correctness

Solutions

Self-Amending & On-chain governance

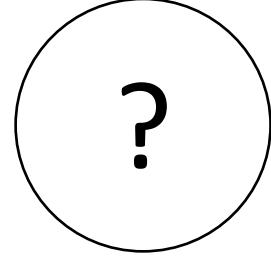
LPoS

Smart Contracts

Formal Verification

Abstract Blockchain

Tezos , **the last** crypto currency



1st Gen. Bitcoin

First ever Cryptocurrency
and blockchain

2nd Gen. Ethereum

Turing complete
Smart contract

Who's the next?

Scalability
Inter-operability
Sustainability

Motivations
Protocol fork
Proof of Work
Smart Contract
Correctness



Abstract blockchain

Self-Amending
On-chain governance
LPoS
Smart contract
Formal Verification
Abstract Blockchain

PoS 합의 알고리즘

Why PoS? No more PoW

Inefficient

PoW의 지나친 에너지 소모와 환경 파괴
Hash puzzle 경쟁으로 인한 중복 계산

Centralization

ASIC과 마이닝 풀의 등장

Mis-alignment of interest

채굴자의 동기는 채굴 보상
(ex. 자동 해시 파워 이동 서비스)

Not scalable

Hash puzzle 경쟁으로 인한 중복 계산
Not solved by just shorter block interval (more stale blocks)

But why PoW is still the King?

Time Tested

Old-school blockchains (Bitcoin, Litecoin, Ethereum) are using PoW
Bitcoin network has never been hacked

Secure

Real proof by burning physical resources
Hard to attack
(Dis)Incentive – Block reward and computing power is a kind of stake

Fair

Very fair guaranteed by cryptography

What is PoS?

PROOF

블록에 포함된 트랜잭션과 해당 블록이 적법하다는 증거 또는 증명

STAKE

특정 퍼블릭 키 해시(주소)가 보유한 지분의 상대적 가치

특정 주소의 staking 토큰 수 / 네트워크 전체 staking 토큰 수

PROOF OF STAKE

지분의 비율에 따라 선택된 주체가 새 블록을 생성

FOLLOW-THE-SATOSHI (Follow-the-coin)

새롭게 생성(mined, minted)된 토큰의 **모든 최소 단위에** 고유한 시리얼 넘버를 부여

시리얼 넘버 한 개를 (무작위로) 선택하고, 이 토큰을 보유한 주소가 블록을 생성함

더 많은 토큰을 보유할수록, 블록 생성 확률이 높음

PoS also should be **fair** and **secure**

Fair & Secure

블록 생성의 기회가 지분의 비율에 따라 공정하게 주어져야 함

스케줄 생성에 개인이 큰 영향을 끼쳐서는 안 됨

이미 정해진 스케줄이 변경되어서도 안 됨

악의적인 행동을 효과적으로 막을 수 있어야 함 (Nothing at stake, Long range attack)

PoS also should be **fair** and **secure**

Validator selection rule

Disincentive for malicious behavior

Incentive for honest behavior

PoS also should be **fair** and **secure**

Validator selection rule

Disincentive for malicious behavior

Incentive for honest behavior

PoS of Tezos, A mix of several ideas

Validator
Selection Rule

Dis-incentive for
Malicious Behavior

Incentive for
Honest Behavior

Multiple snapshots
Random seed

Safety deposit
Plenitude rule

Block reward
Accussing reward

PoS of Tezos, A mix of several ideas

Validator
Selection Rule

Dis-incentive for
Malicious Behavior

Incentive for
Honest Behavior

Multiple snapshots
Random seed

Safety deposit
Plenitude rule

Block reward
Accussing reward

용어 정리

Baking

Block producing , Staking

Baker

Block producer, validator, miner, forger

Delegation

Only staking right, NOT ownership

Cycle

Period defined in protocol

4,096 Blocks (1 block = 1 min)

Validator selection rule = Roll snapshot + Random seed

Follow the roll

1 roll = 8,000 xtz (베이커 보유분 + 위임 받은 수량)

베이킹 파워가 를 단위로 내림(Rounded down)

8,000 ~ 15,999 xtz = 1 roll

FTS 알고리즘 측면에서는 효율적

Roll snapshot을 통해 베이킹 파워를 측정

Tezos's PoS

PoS with Optional Delegation

Validator selection rule = Roll snapshot + Random seed

Random seed

베이커들이 결정

예측이 사실상 불가능한 무작위 숫자열

Validator selection rule = Roll snapshot + Random seed

Roll snapshot

지분을 얼마나 보유하고 있는가

+

Random seed

예측이 사실상 불가능한 무작위 숫자열



Baking rights

블록 B는 베이커 A가 생성한다

Validator selection rule = Roll snapshot + Random seed

Cycle

1 Cycle = 4,096 블록 = 4,096분 = 2일 20시간 16분 (best case)

사이클 단위로 베이킹 스케줄이 **미리** 결정됨



Baking rights in cycle 1

Block 4,097: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)

Block 4,098: 베이커(Satoshi, Charles, Dan, Vitalik, Kwon, ...)

Block 4,099: 베이커(Justin, Kate, Arthur, Satoshi, Satoshi, ...)

...

Block 8,192: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)

Validator selection rule = Roll snapshot + Random seed

Block 8,192: 베이커(Alice, Diego, Arthur, Charlie, Satoshi, ...)



가장 높은 우선순위의 베이커는 이전 블록 생성 후 **1분** 후 베이킹 가능

2번째 우선순위의 베이커는 이전 블록 생성 후 **2분** 후 베이킹 가능

...

16분이 지난 경우, **보증금 없이** 베이킹 가능

체인별 블록 생성 시간을 통해서 체인의 score를 가늠할 수 있음

Validator selection rule = Roll snapshot + Random seed

Roll snapshot in **Cycle N**

지분을 얼마나 보유하고 있는가

+

Random seed in **Cycle N+1**

예측이 사실상 불가능한 무작위 숫자열



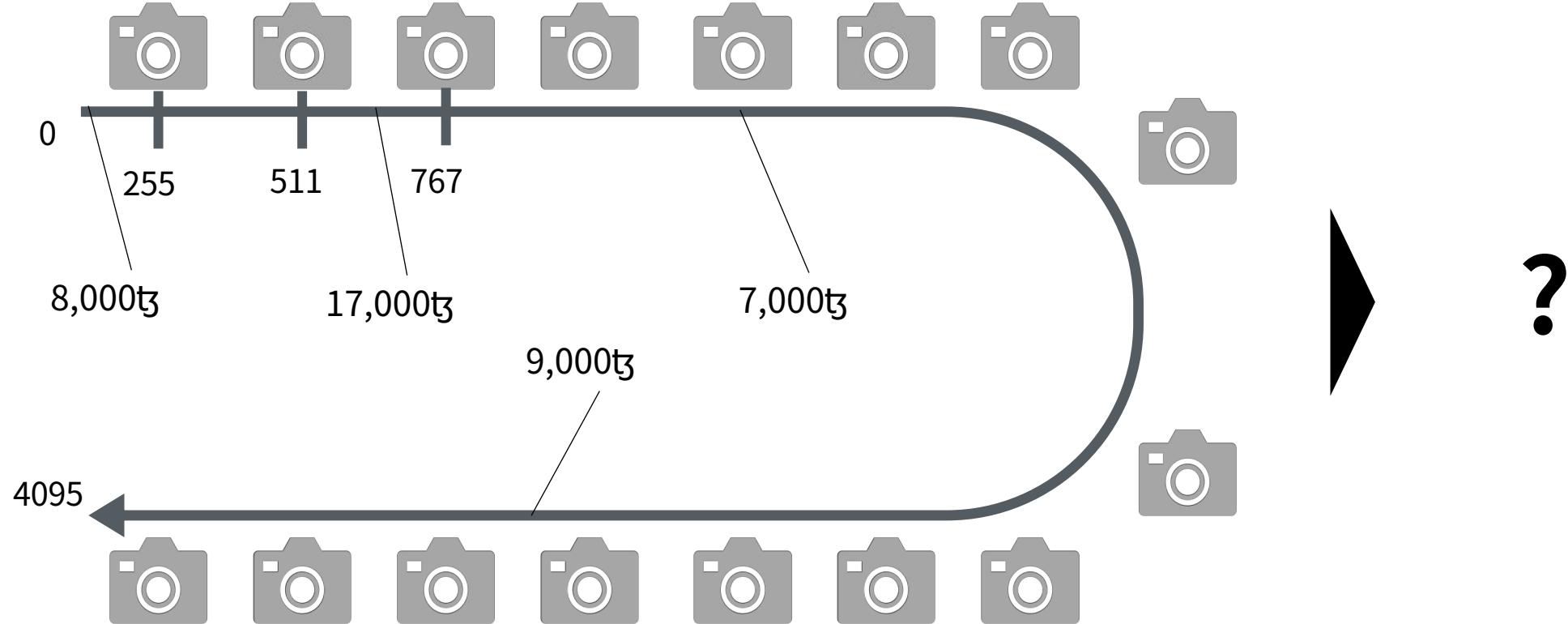
Baking rights in **Cycle N+7**

블록 B는 베이커 A가 생성한다

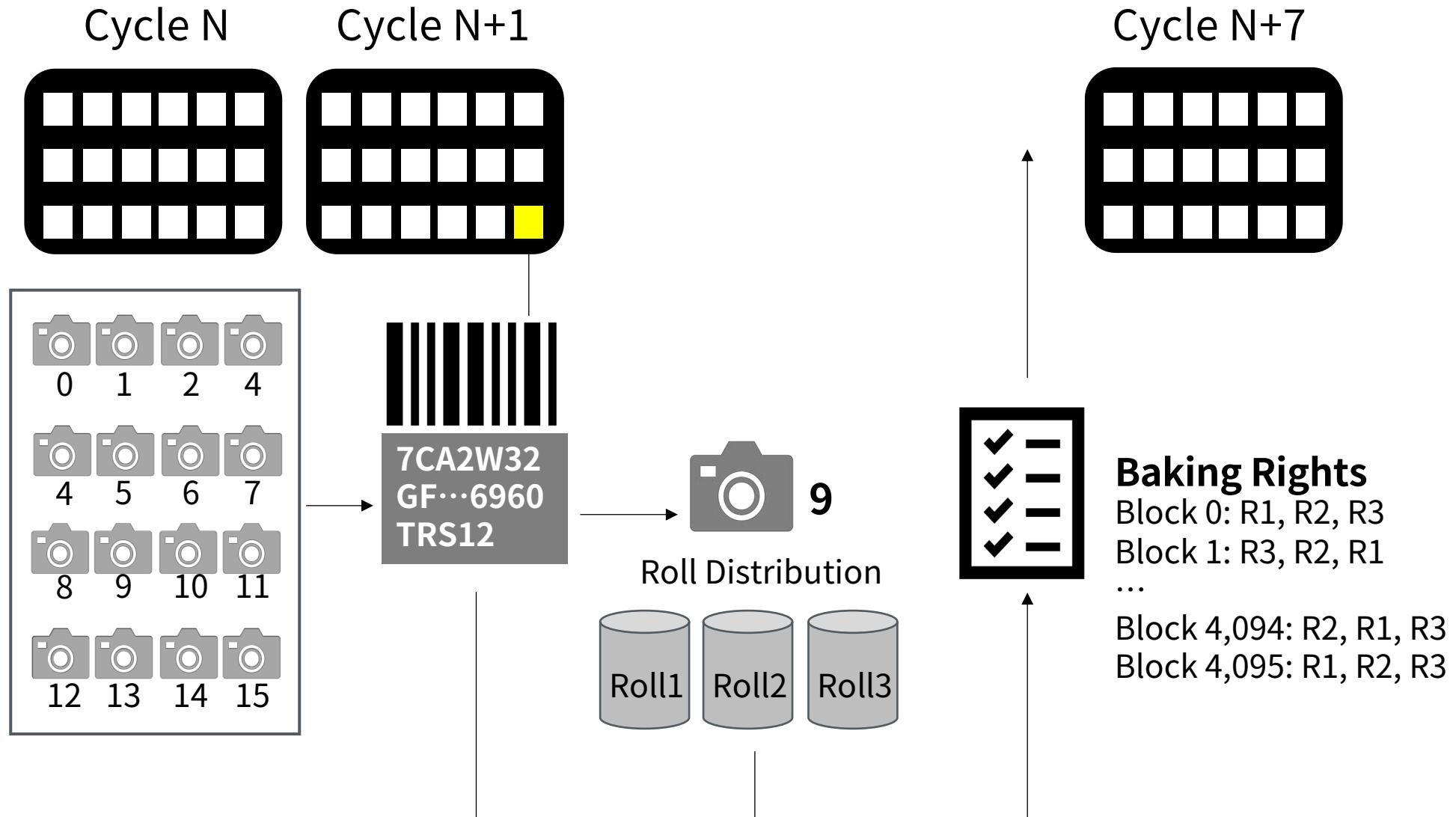
Validator selection rule = Roll snapshot + Random seed

Multiple snapshots (16 in a cycle) and random selection

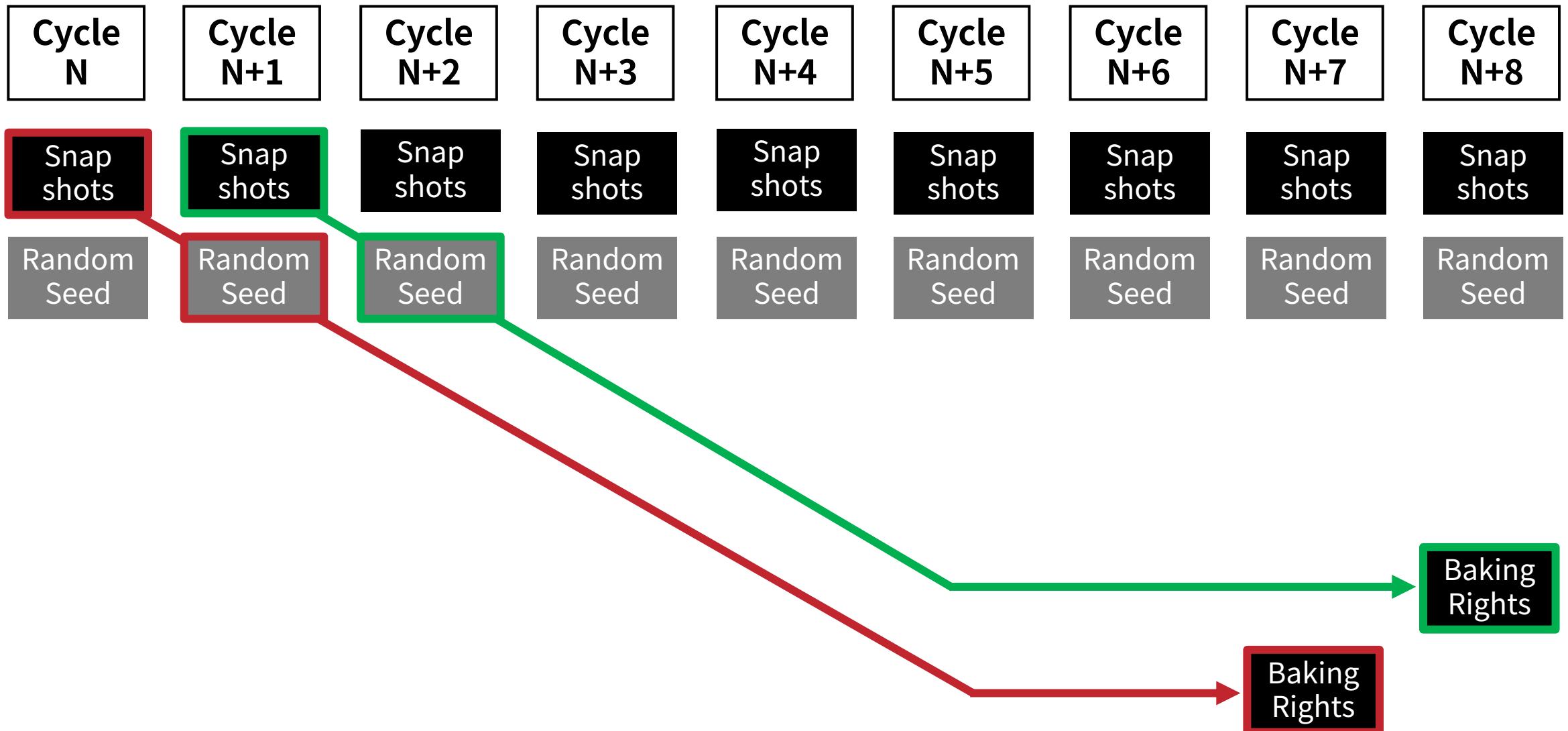
Proof of **STAKING**



Validator selection rule = Roll snapshot + Random seed



Validator selection rule = Roll snapshot + Random seed



Validator selection rule = Roll snapshot + Random seed

Random seed

베이커는 미리 정해진 베이킹 스케줄에 따라 블록B를 생성

임의의 숫자를 암호화(**hash commitment**)하여 블록B에 포함

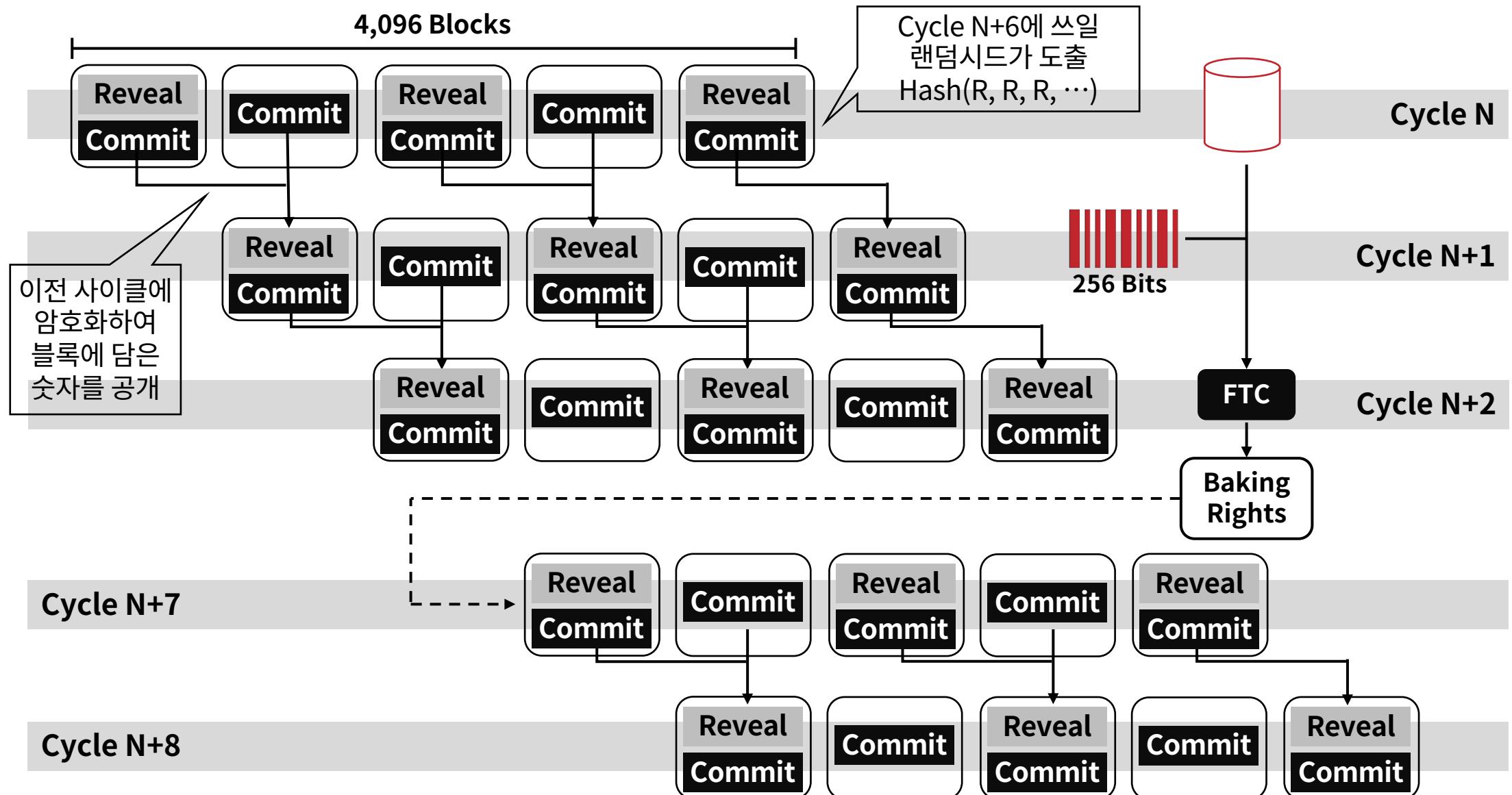
이 숫자는 다음 사이클에 공개(**reveal**)해야 함

숫자를 공개하지 않은 경우, 블록B의 보상과 수수료 몰수

매 사이클 마지막에 공개된 숫자를 모두 결합, 해싱하여 랜덤 시드 생성

공개는 32블록마다 자유롭게 가능

Validator selection rule = Roll snapshot + Random seed



PoS of Tezos, A mix of several ideas

Validator
Selection Rule

Dis-incentive for
Malicious Behavior

Incentive for
Honest Behavior

Multiple snapshots
Random seed

**Safety deposit
Plenitude rule**

Block reward
Accussing reward

Disincentive = Deposit + Endorsement

Endorsement

블록당 32명의 Endorser가 미리 정해짐 (Roll snapshot + Random seed)

이전 블록에 투표(endorse, notarize)

$\mathcal{B}lock_i$ 가 베이킹 된 후, 인도서들은 $\mathcal{B}lock_i$ 에 대해 서명을 제출함

$\mathcal{B}lock_i$ 베이커가 이 투표를 블록에 포함

체인의 score(fitness) = 인도스먼트의 합

인도성이 블록에 담기고, Canonical chain이 되어야 보상

보상 = 최대 $2 \frac{\epsilon}{dT}$ ($2 / dT$)

Disincentive = Deposit + Endorsement

Safety Deposit

베이킹, 인도싱에 요구되는 조건

5사이클 동안 동결(locked, frozen)

악의적인 행동 시 몰수(forfeit, slashed)

Baking	Number	Deposits	Rewards
Baking	1	512 tS	16 tS
Endorsing	Up to 32	64 tS	2/dT tS

PoS with Optional Delegation

Chain-based PoS consensus with a use of endorsements to speed-up confirmation times and reduce selfish baking

PoS of Tezos, A mix of several ideas

Validator
Selection Rule

Dis-incentive for
Malicious Behavior

Incentive for
Honest Behavior

Multiple snapshots
Random seed

Safety deposit
Plenitude rule

Block reward
Accussing reward

Incentive

Baking reward

Steal: 후순위 베이커가 베이킹 하는 경우

Endorsing reward

최대 $2 \frac{t_3}{dT}$ (2 / dT)

Accusing reward

악의적인 행동을 신고

악의적인 행동이 적발된 베이커는 Safety deposit 몰수

몰수된 토큰의 절반은 소각(proof of burn), 절반은 신고자에게 지급

합의 알고리즘 심화

합의, 분산 시스템의 목적

Consensus protocol

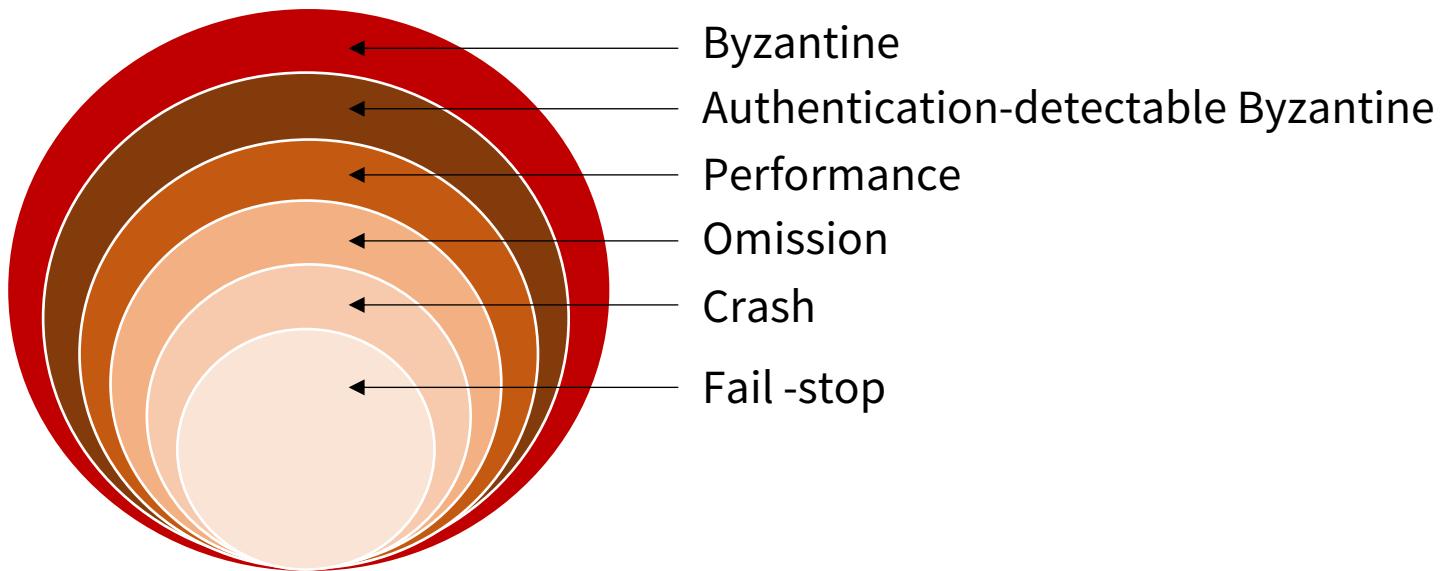
The mechanism where consensus is built around the a single chain

Byzantine fault tolerance

The characteristic of distributed systems which tolerates failures called Byzantine faults.

The Byzantine Generals Problem (Lamport, 1982)

No solution with fewer than $3m + 1$ generals can cope with m traitors. (OM Algorithm)



Safety vs. Liveness*

A **safety** property is one which states that **something will not happen**.

- A program with the correct input cannot stop if it does not produce the correct output.

A **liveness** property is one which states that **something must happen**.

- A program will **terminate** if its input is correct.

* Proving the Correctness of Multiprocess Programs (Lamport, 1977)

FLP Result*

No completely **asynchronous** consensus protocol can tolerate even **a single** unannounced process death (**crash failure**)

- both **termination** and **agreement** cannot be satisfied.

* Impossibility of Distributed Consensus with One Faulty Process (Fischer, Lynch & Patterson, 1985)

합의, 분산 시스템의 목적

Paxos protocols*

Safety is guaranteed in a **closed network** with a **quorum rule** and a multi-phase learning process.

* The Part-Time Parliament (Lamport, 1989)

합의, 분산 시스템의 목적

Practical BFT

Both safety and liveness are guaranteed but use an assumption of partial synchrony.

합의, 분산 시스템의 목적

Safety vs. Liveness

FLP Result

Paxos protocols

Practical BFT

Safety vs. Liveness

BFT-based Consensus

Classical

Safety over liveness

Deterministic finality



Nakamoto Consensus

Chain-based

Liveness over safety

Probabilistic finality



블록체인 거버넌스

What is governance?

Governance

All processes of governing

Interaction, Decision-making, Enforcement, ...

조직 또는 집단의 지배구조

정치적 이해관계자들의 다원적, 협력적 통치 방식 (정치학)

다양한 행위자들이 문제를 해결하는 새로운 국정운영방식 (행정학)

What is governance?

Blockchain network: 하나의 독립된 사회 또는 조직

Blockchain protocol: 네트워크에 참여한 노드들이 연결되는 방식

**Blockchain protocol is
a form of governance**

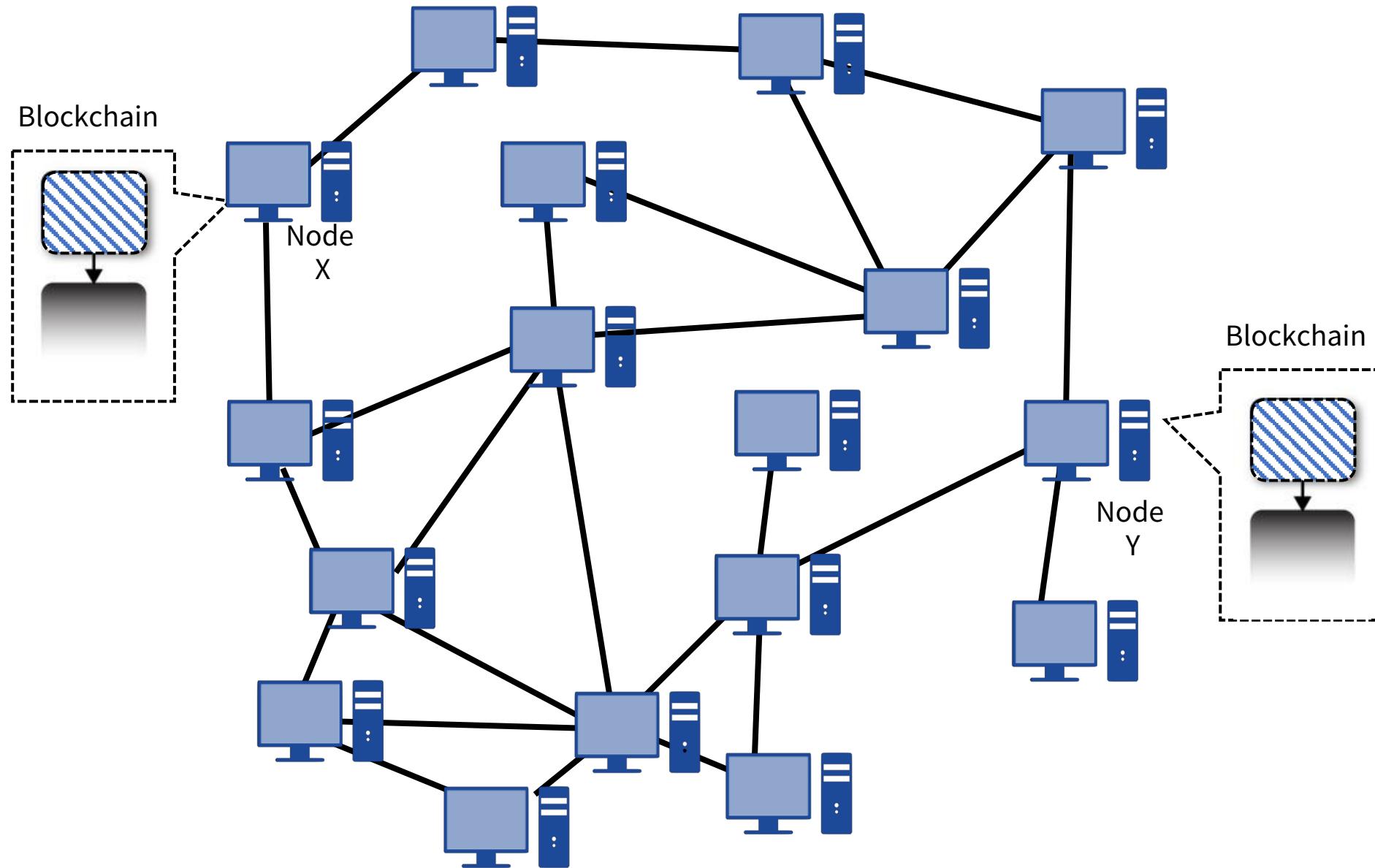
What is governance?

Blockchain Governance

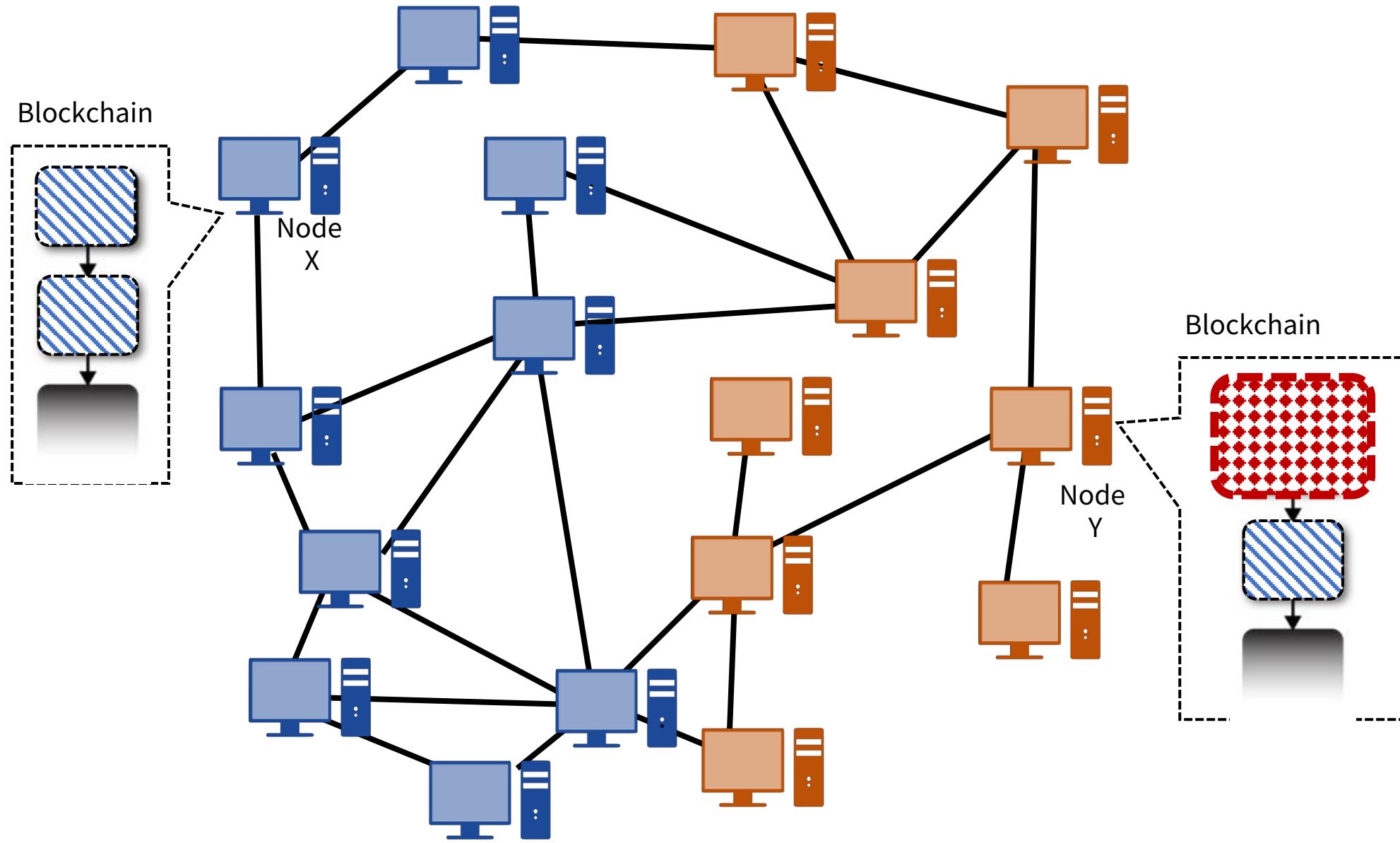
합의 알고리즘을 포함하여 네트워크에 참여한 노드들이 공유하는 규칙의 집합체
블록체인 전체 생태계에서 이루어진 모든 의사결정 방식의 체계

새로운 규칙의 생성, 변경, 폐기를 포함

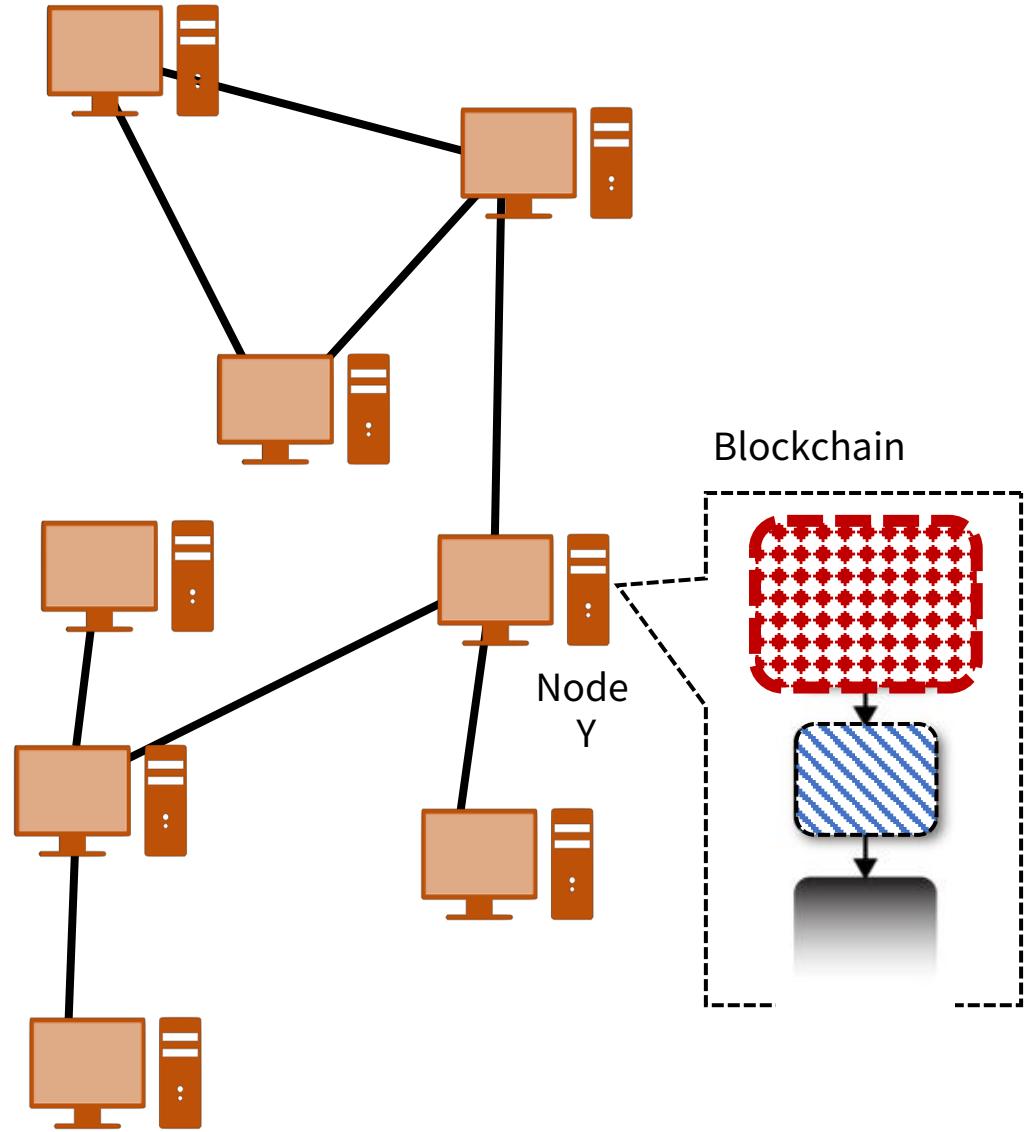
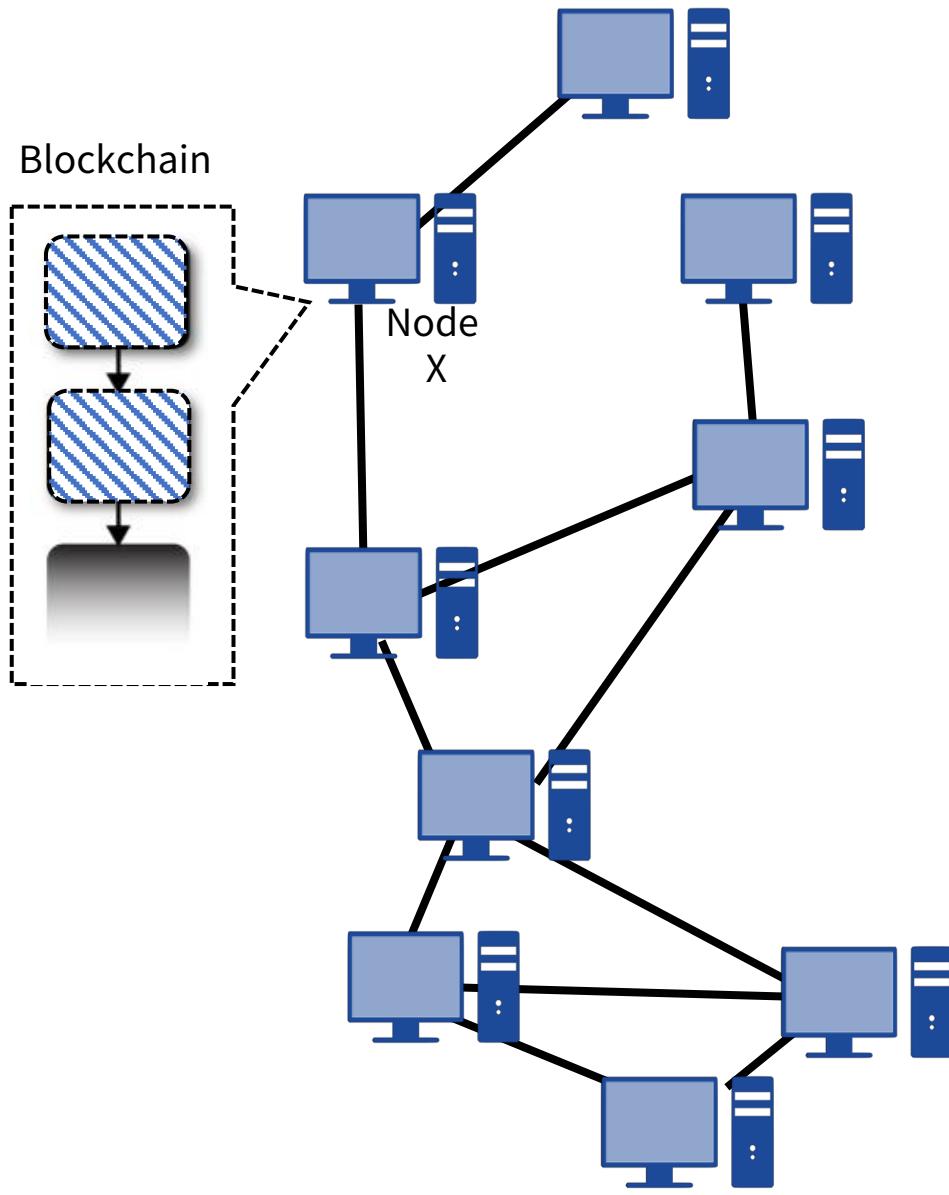
프로토콜의 변경



프로토콜의 변경



프로토콜의 변경



What is governance?

Procedure & Coordination

Proposal (Who propose, What propose, **Incentives**)

Discussion (**Information** gap)

Voting (1 head – 1 voting, staked-based voting, **Incentives**)

Operator's favor vs. User's favor

Updating the software at the same time

Migration cost

What is governance?

Off-chain vs. On-chain

블록체인 거버넌스가 어디에서 이루어지고 있는가

기존 방식(Off-chain)보다 더 나은 방식은 없을까

다른 건 기록하면서 왜 거버넌스는 기록하지 않지

Off-chain governance: Bitcoin Improvement Proposal (ref #10)

bitcoin / bips

Watch 625 Star 3,547 Fork 1,962

Code Pull requests 93 Projects 0 Wiki Security Insights

Bitcoin Improvement Proposals <http://bitcoin.org>

1,888 commits 5 branches 0 releases 196 contributors

Number	Layer	Title	Owner	Type	Status
1		BIP Purpose and Guidelines	Amir Taaki	Process	Replaced
2		BIP process, revised	Luke Dashjr	Process	Active
32	Applications	Hierarchical Deterministic Wallets	Pieter Wuille	Informational	Final
38	Applications	Passphrase-protected private key	Mike Caldwell, Aaron Voisine	Standard	Draft
39	Applications	Mnemonic code for generating deterministic keys	Marek Palatinus, Pavol Rusnak, Aaron Voisine, Sean Bowe	Standard	Proposed

Off-chain governance: Ethereum Improvement Proposal (ref #11)

The Ethereum Improvement Proposal repository <http://eips.ethereum.org/>

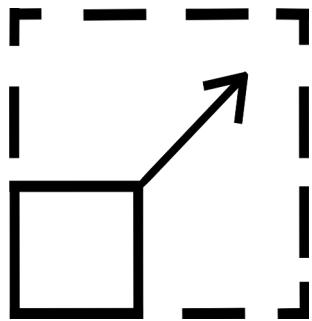
The screenshot shows the GitHub repository for Ethereum Improvement Proposals (EIPs). At the top, there's a navigation bar with links for 'Code', 'Issues 369', 'Pull requests 112', 'Projects 0', 'Security', and 'Insights'. To the right are buttons for 'Watch' (816), 'Star' (4,565), 'Fork' (1,505), and 'Issues' (369). Below the navigation is a summary bar with metrics: 1,702 commits (red), 2 branches (dark red), 0 releases (yellow), and 203 contributors (brown). A green bar is partially visible on the right. The main content area displays a table of EIPs:

Number	Title	Author
2	Homestead Hard-fork Changes	Vitalik Buterin
7	DELEGATECALL	Vitalik Buterin
20	ERC-20 Token Standard	Fabian Vogelsteller, Vitalik Buterin
137	Ethereum Domain Name Service - Specification	Nick Johnson
140	REVERT instruction	Alex Beregszaszi, Nikolai Mushegian
141	Designated invalid EVM instruction	Alex Beregszaszi

Challenges in blockchain (feat 2.0)

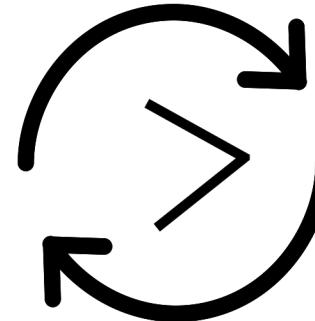
Scalability

Storage
Process time
Validation time
Propagation time
of miners



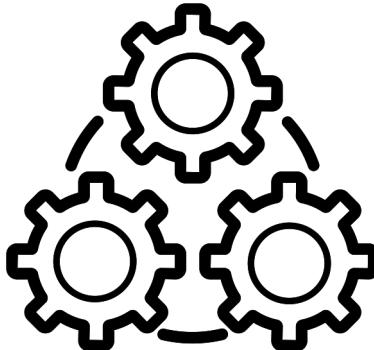
Sustainability

Resolving conflicts
Adopting new innovations



Interoperability

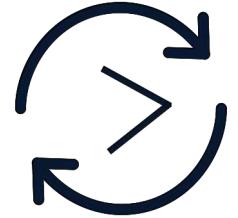
Communication between
different protocol



Security

Core security
Smart contract security

바보야, 문제는 **거버넌스**야



Sustainability

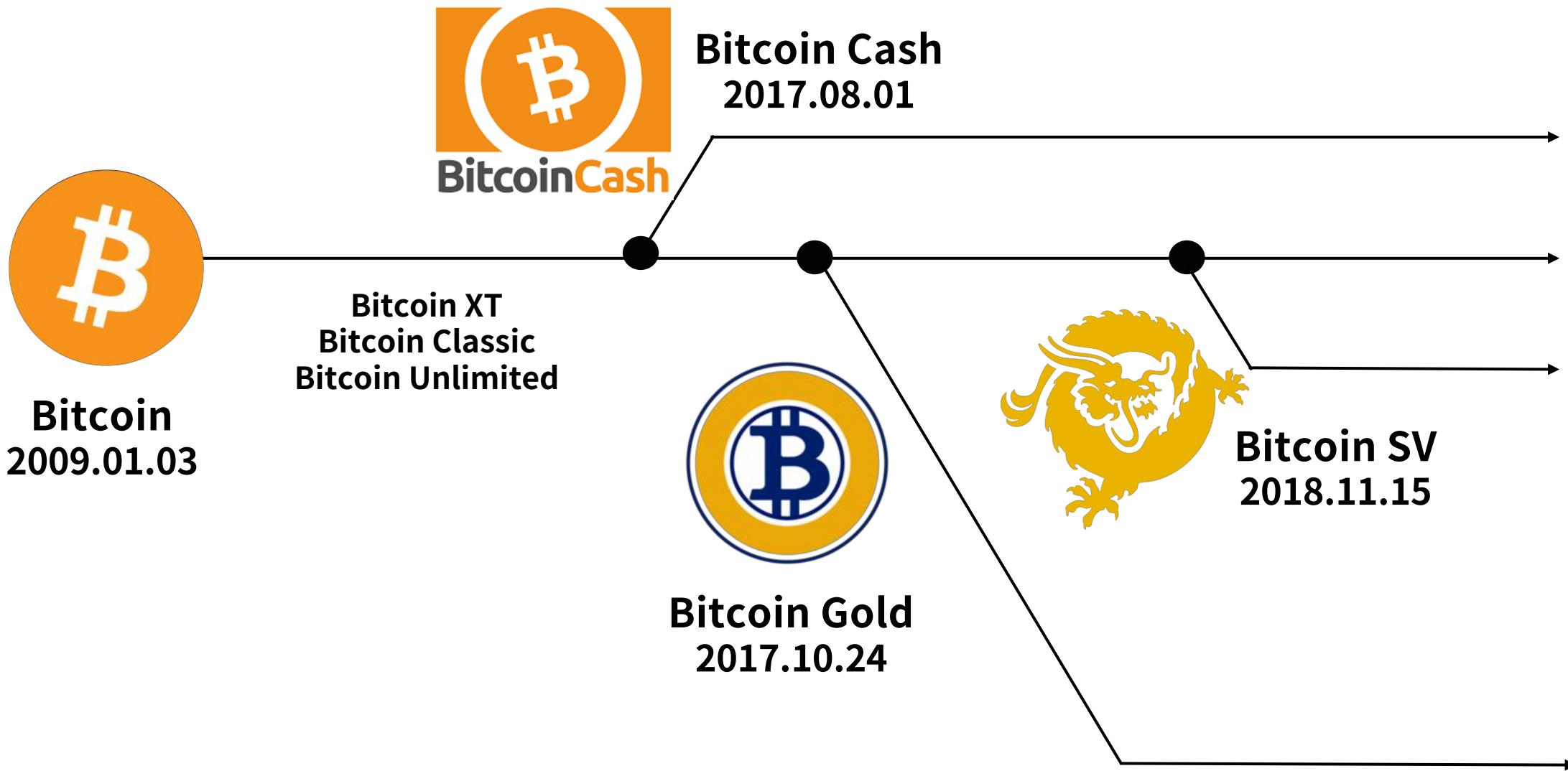
어떻게 (프로토콜의) 갈등을 해결할 것인가

새로운 혁신들을 어떻게 받아들일 것인가

어떤 혁신을 받아들일 것인가

재단(Foundation)과 개발팀(Dev team)에 의존하는 게 맞는가

Hard fork case: Bitcoin



Hard fork case: Ethereum



Ethereum
Classic
2015.07.30

DAO Attack
2016.06.17
\$50M Hacked



Ethereum
2016.07.21

Hard fork case: Ethereum

Name	On Roadmap	Date	Block
Frontier	Yes	2015.07.31	1
Frontier Thawing	Yes	2015.09.08	200000
Homestead	Yes	2016.03.15	1150000
DAO Fork	No	2016.07.21	1920000
EIP-150 Hard Fork	No	2016.10.19	2463000
Spurious Dragon	No	2016.11.23	2675000
Byzantium	Yes	2017.10.16	4370000

Off-chain vs. On-chain

Off-chain governance

Old school blockchains (Bitcoin, Ethereum, forks of them)

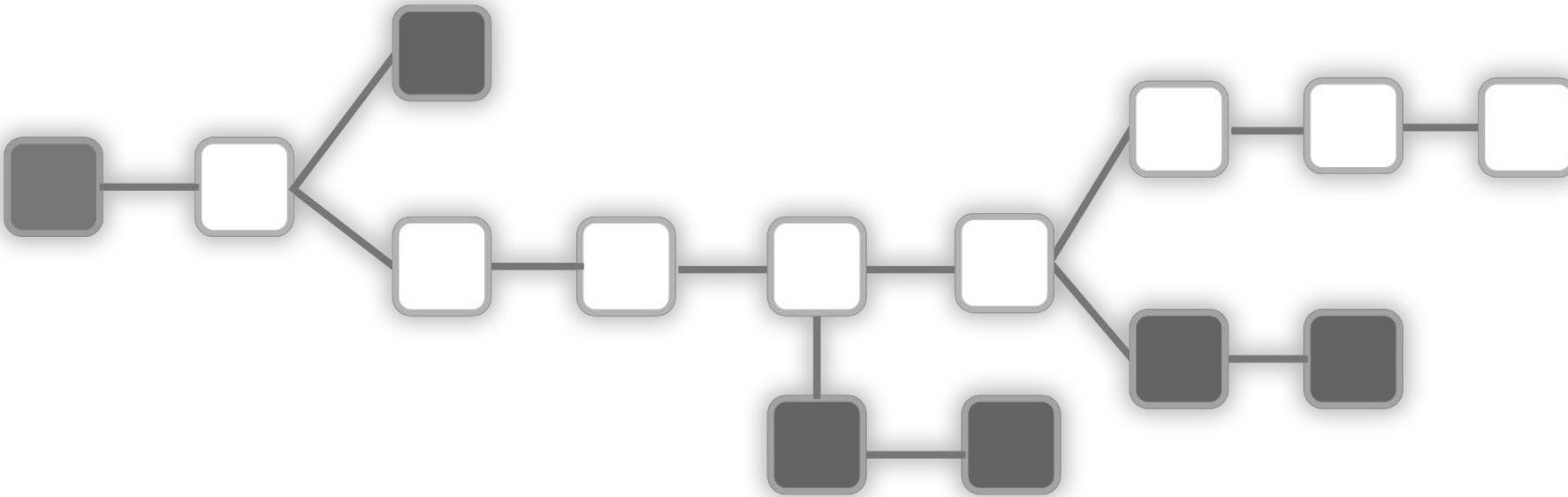
Power balancing between core developers, miners, users, business

Online forum, Improvement proposals(BIP, EIP)

(better than legacy but) **Centralized** (public lack the technical knowledge)

Opt-in (flexibility not to choose, easily fork by open-source)

Abstract Blockchain

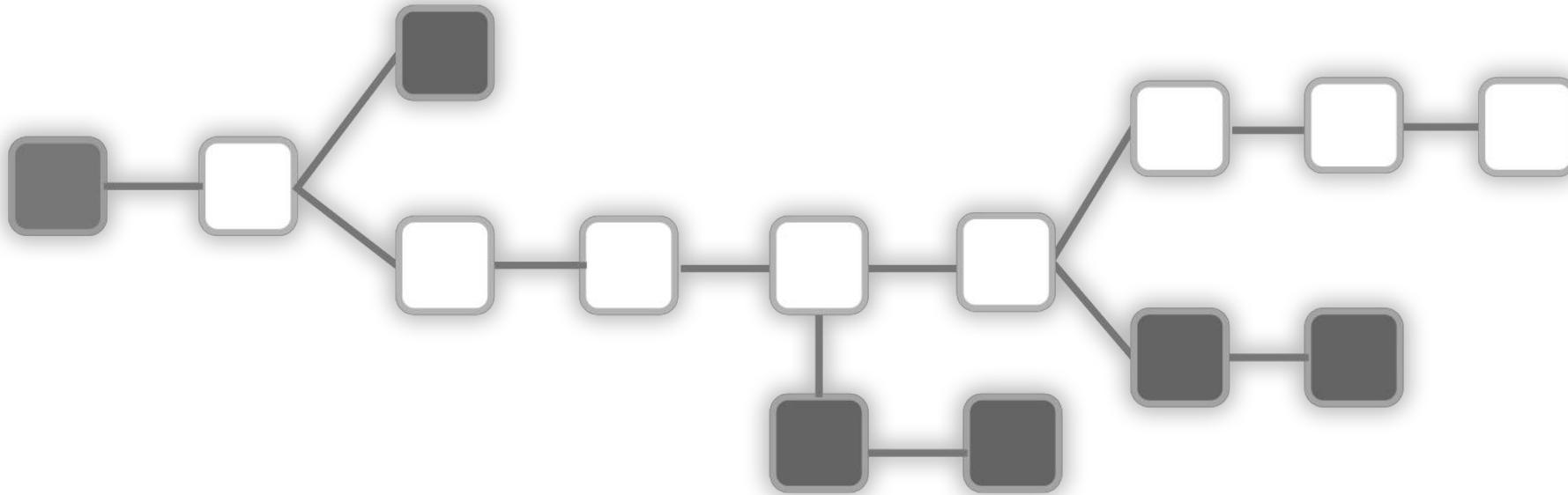


All existing block chain implementation generally

apply: $S \mapsto O \mapsto S$ or $S' = f(S, O)$

score: $S \mapsto N$ or $N = f(S)$

Abstract Blockchain



Tezos's most powerful features

set_test_protocol: replacing the protocol used in the test chain with a new protocol

promote_test_protocol: replacing the current protocol with the protocol being tested

On-chain governance case: Tezos

ZD Net Korea

테조스, 하드포크 없이 첫 번째 업데이트 성공

기사입력 2019.05.30. 오후 2:40 기사원문 스크랩 본문듣기 설정

5 2

요약 Beta 가 드롭说出来



"업데이트로 연산 능력 두배 증가"

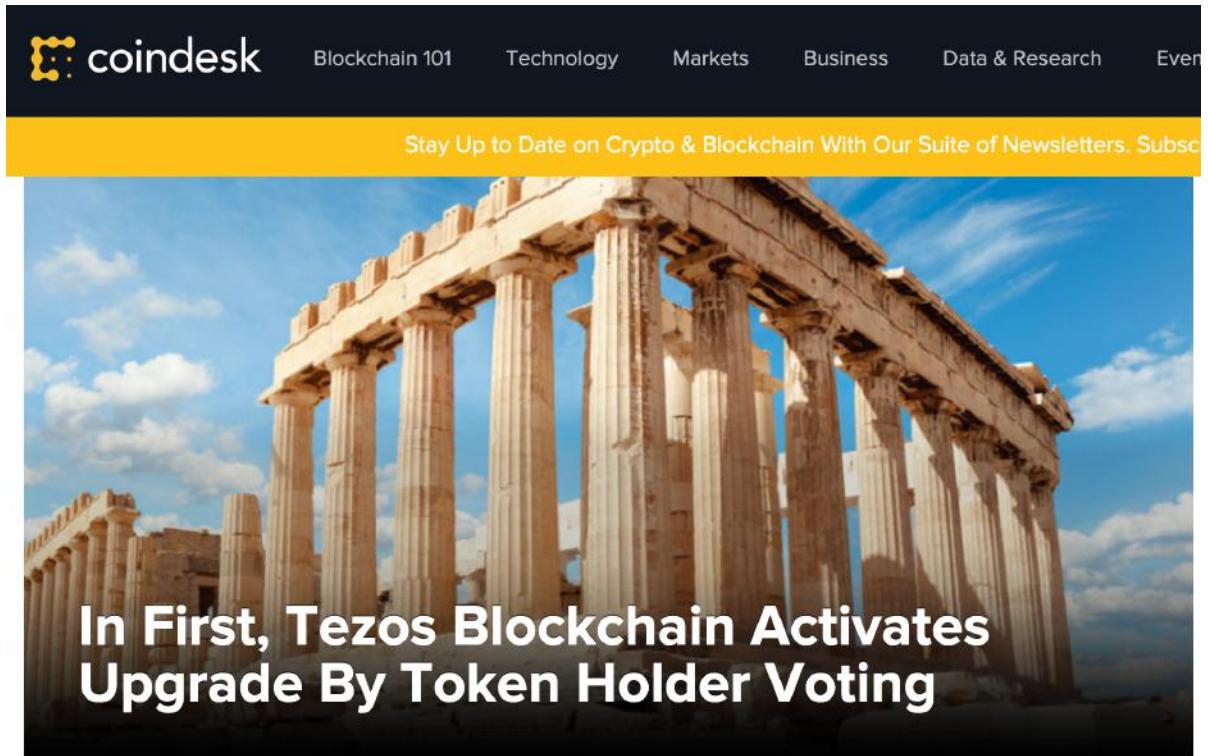
(지디넷코리아=임유경 기자)블록체인 플랫폼 테조스가 온체인 거버넌스 투표를 통해 하드포크 없이 첫 번째 업그레이드를 완료했다.

테조스코리아(대표 이진우)는 테조스 블록체인에 첫번째 프로토콜 업데이트 제안이 올라온 후, 약 3개 월만에 총 4단계 절차를 거쳐서 업데이트를 완료했다고 28일 밝혔다.

이번 업데이트 제안은 마지막 절차인 '적용 단계'에서 투표 참여율 84.35%에 찬성을 99.89%(Pass 제외)로 통과됐다. 458753번째 블록부터 업데이트가 적용됐다.

coindesk Blockchain 101 Technology Markets Business Data & Research Events

Stay Up to Date on Crypto & Blockchain With Our Suite of Newsletters. Subsc



In First, Tezos Blockchain Activates Upgrade By Token Holder Voting

On-chain governance case: **Tezos**

Athens Proposal

2019년 3월 ~ 5월

참여율: 84.35% (약 4.7억개 토큰)

찬성률: 99.89%

Stake unit ($10,000 \text{ \(\zeta\)} \rightarrow 8,000 \text{ \(\zeta\)}$)

Gas limit ($x2 \uparrow$)



Tezos is **formalizing** blockchain governance

Protocol amendments

프로토콜 수정, 추가, 제안

In-protocol voting procedures

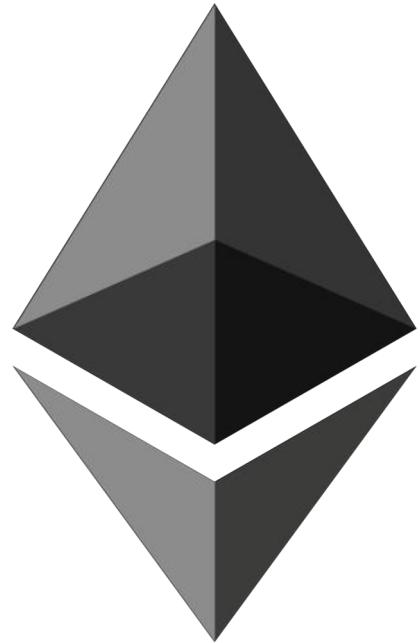
Rules about changing the rules

거버넌스의 보조 도구로서 블록체인(crypto-ledger) 사용

프로토콜 제안과 투표는 **primitive operations**

Bakers vote

Tezos is **formalizing** blockchain governance

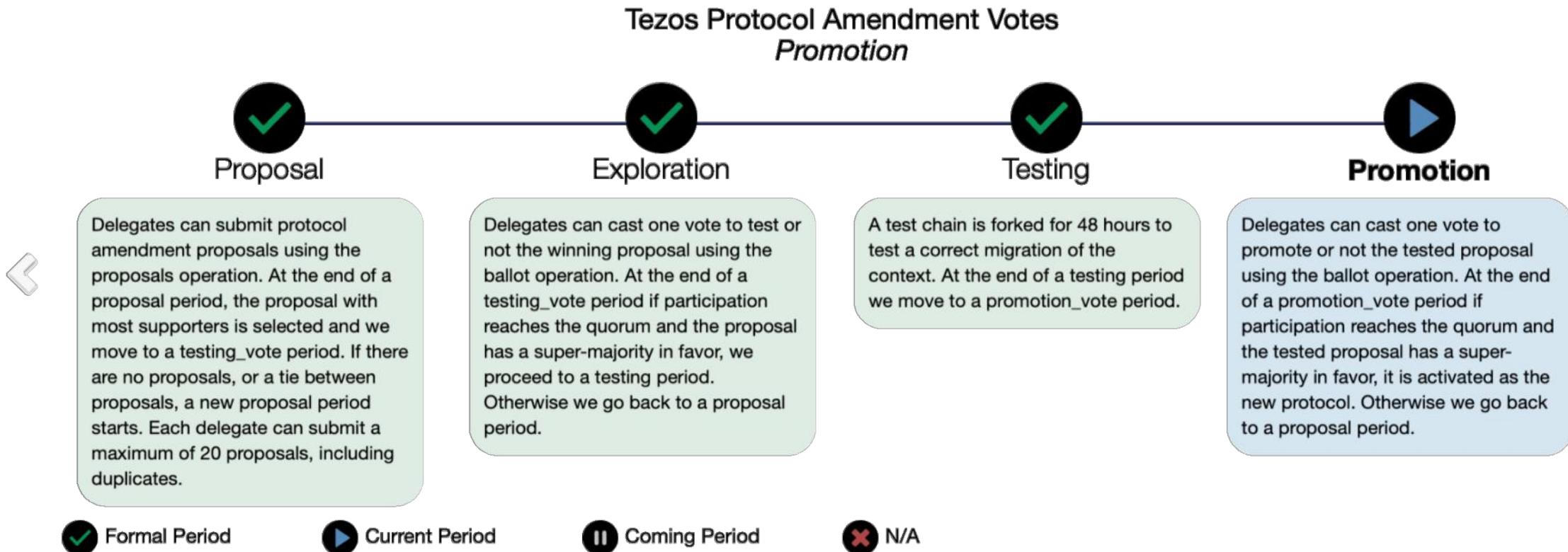


Thin protocol



Fat protocol

Tezos is formalizing blockchain governance



```
[ubuntu@ip-172-31-4-183:~$ tezos-client submit ballot for tezoskorea ]  
Pt24m4xiPbLDhVgVfABUjirbmda3yohdN82Sp9FeuAXJ4eV9otd yay
```

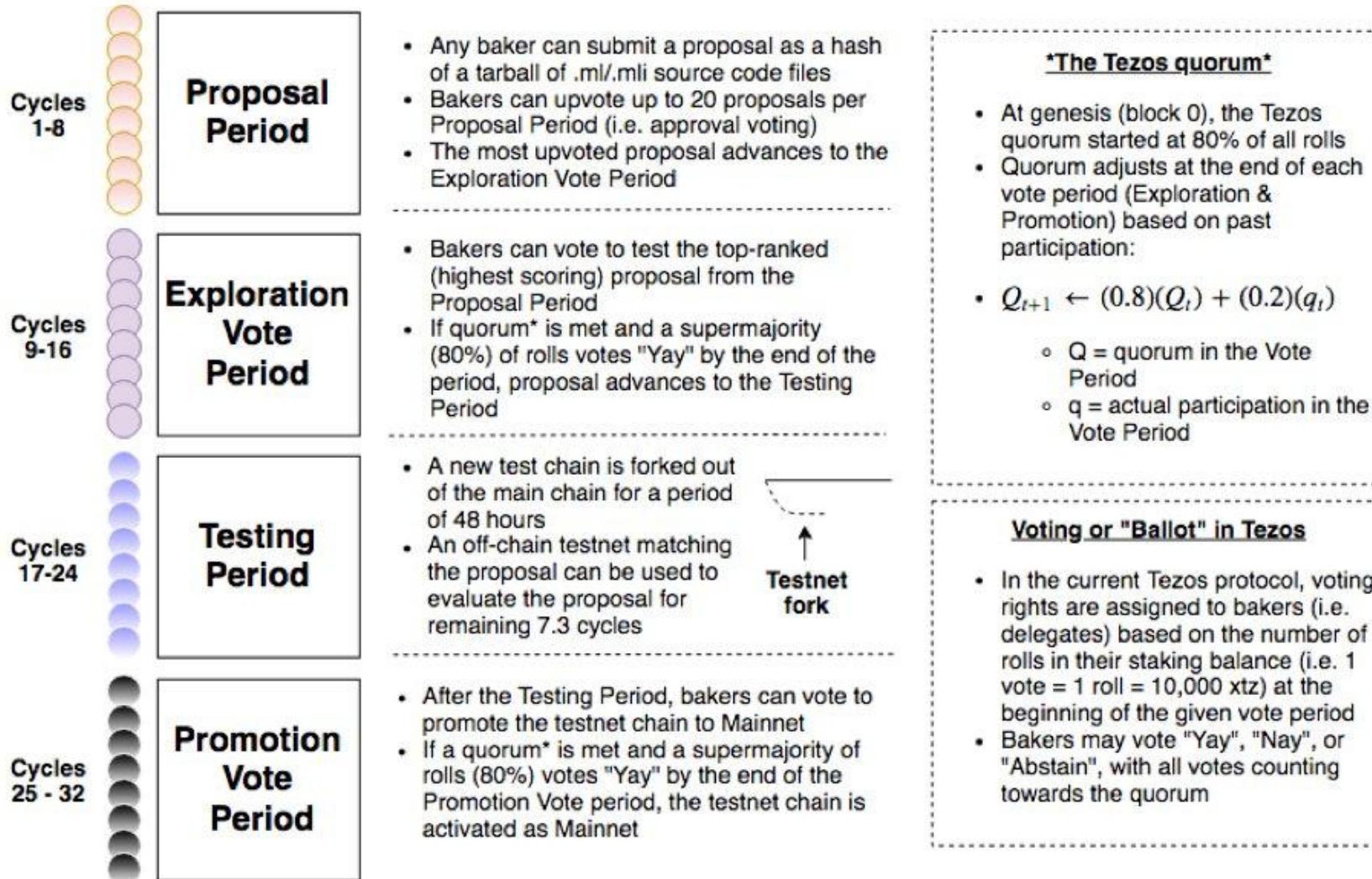
Tezos is formalizing blockchain governance

```
{ protocol: "PsddFKi32cMJ2qPjf43Qv5GDWLDPzb3T3bF6"
  next_protocol: "Pt24m4xiPbLDhVgVfABUjirbmida3yoh"
- test_chain_status: {
    status: "not_running"
  },
  max_operations_ttl: 60,
  max_operation_data_length: 16384,
  max_block_header_length: 238,
+ max_operation_list_length: [...],
  baker: "tz3NExpXn9aPNZPorRE4SdjJ2RGrfbJgMAaV",
- level: {
    level: 458752,
    level_position: 458751,
    cycle: 111,
    cycle_position: 4095,
    voting_period: 13,
    voting_period_position: 32767,
    expected_commitment: true
  },
}

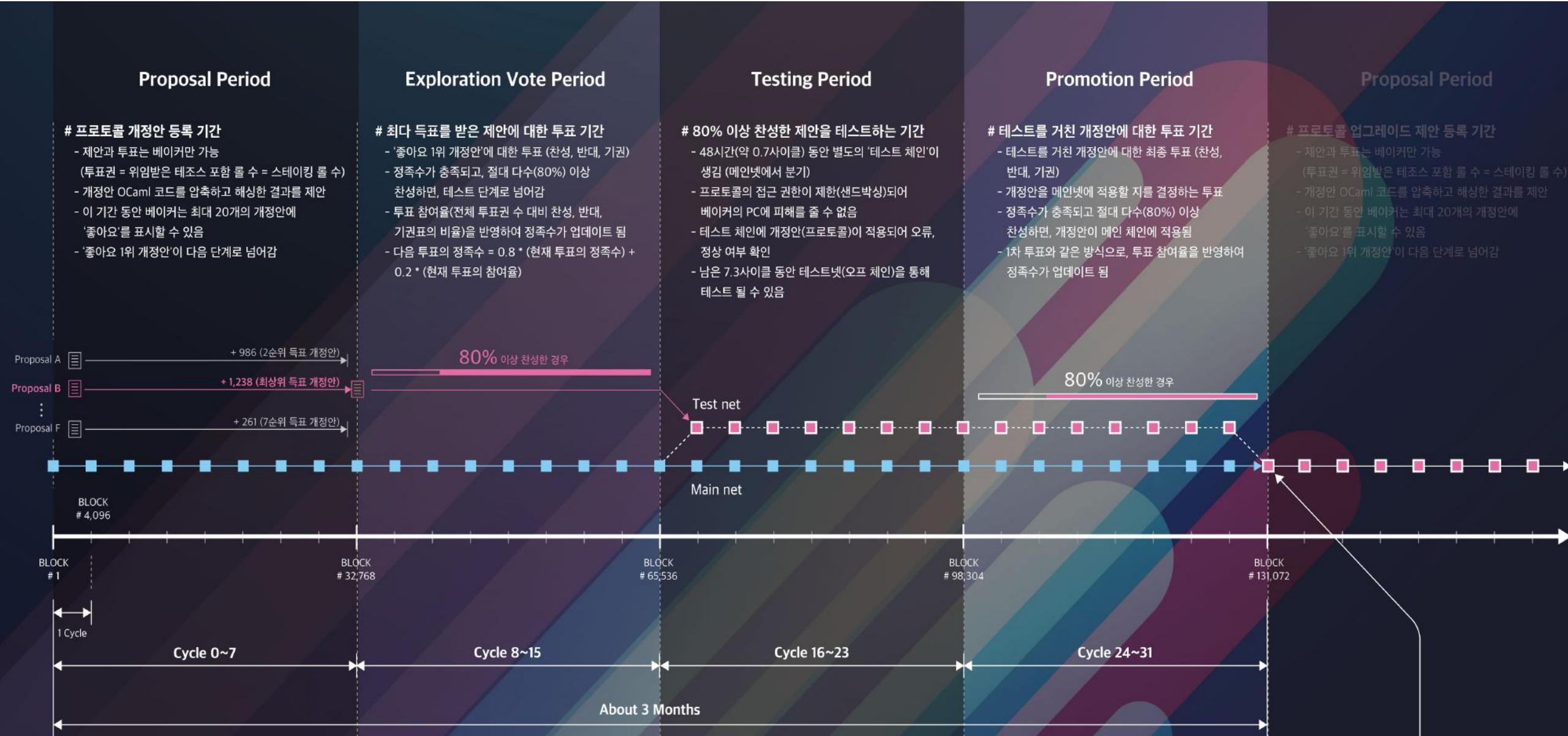
{
  protocol: "Pt24m4xiPbLDhVgVfABUjirbmida3yohdN82Sj"
  next_protocol: "Pt24m4xiPbLDhVgVfABUjirbmida3yohdN82Sj"
- test_chain_status: {
    status: "not_running"
  },
  max_operations_ttl: 60,
  max_operation_data_length: 16384,
  max_block_header_length: 238,
+ max_operation_list_length: [...],
  baker: "tz1NortRftucvAkD1J58L32EhSVrQEWCEnB",
- level: {
    level: 458753,
    level_position: 458752,
    cycle: 112,
    cycle_position: 0,
    voting_period: 14,
    voting_period_position: 0,
    expected_commitment: false
  },
}
```

Tezos is formalizing blockchain governance

An Overview of the Tezos Governance Mechanism



Tezos is formalizing blockchain governance



In Tezos, **stakeholders govern** the protocol

Decentralized development

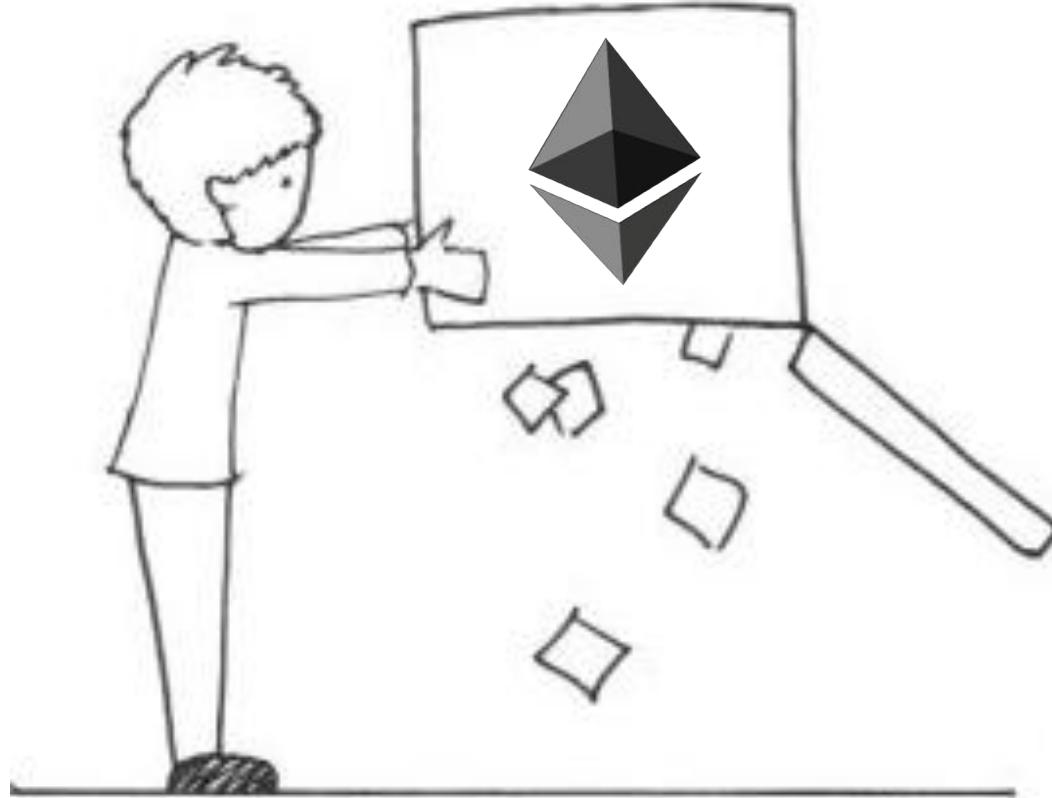
Anyone can propose an amendment

No roadmap

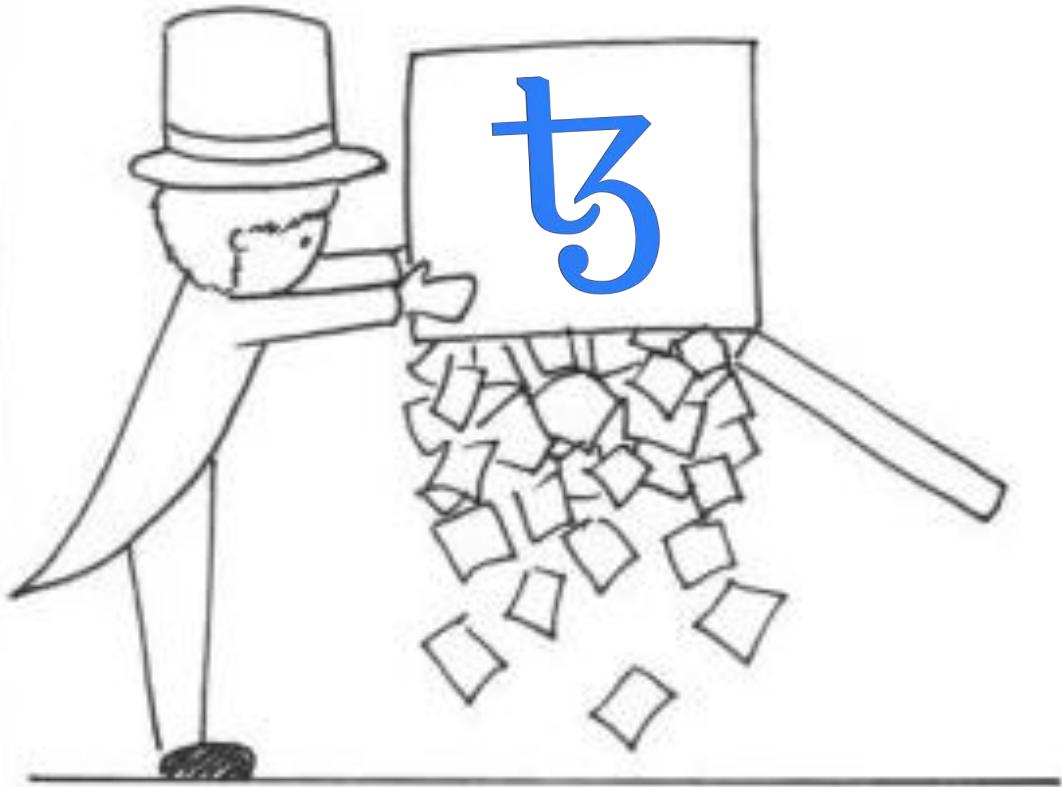
Amendments to the voting procedure itself

Funding by attaching an **invoice**

In Tezos, **stakeholders govern** the protocol



DAO Carbon Vote
4.5%



Proposal Athenes
84.35% (47,049 rolls)

Developments in parallel

Decentralized innovations

Ergo compiler for legal contract (L2, Rust binding)

Zk-Snark (L1, Rust binding)

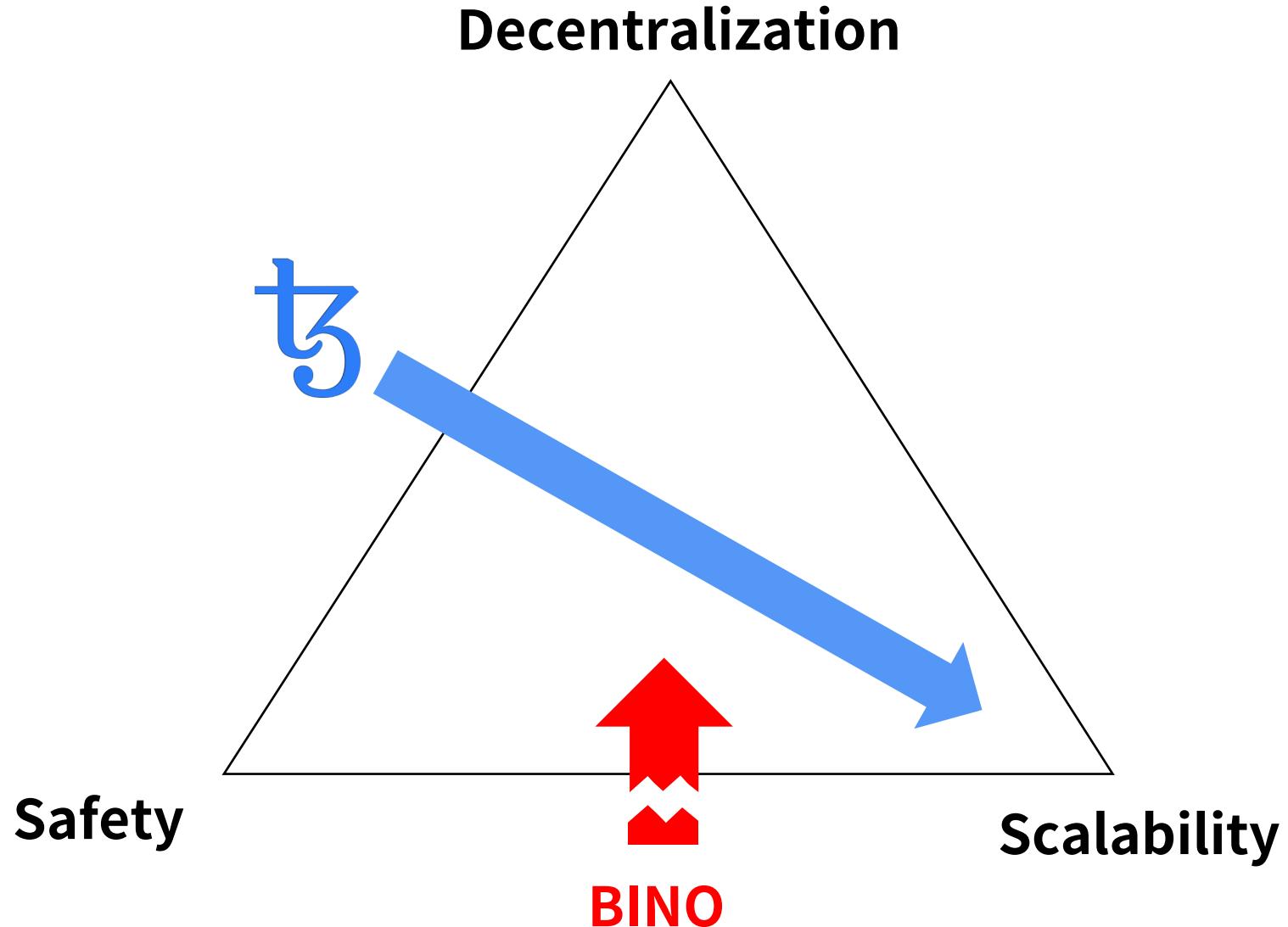
Tendermint as finality gadget (L1, Nomadic Labs, Cryptium Labs)

Sharding (L1, Emin)

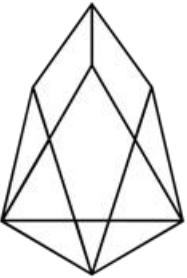
(Plasma-like) **Marigold** (L2, Nomadic Labs)

Somewhere we don't know

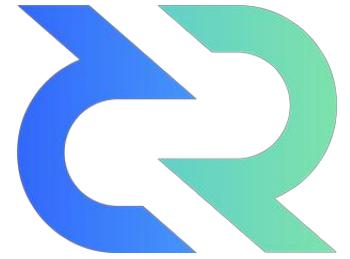
Blockchain trilemma



On-chain governance models

<p>Polkadot.</p> <p>End of 2019</p>	<ul style="list-style-type: none">- Stake-weighted referenda with adaptive super-majority thresholds- Proposing, voting, tallying- Locking for increasing voting power- Voting for council (veto, 1 year term)- Switching out the entire code of the runtime
	<ul style="list-style-type: none">- Governed blockchain (republicanism)- Amendable Constitution and ECAF (EOS Core Arbitration Forum)- 21 validators (Centralization, validator collisions)- Fork based protocol upgrade

On-chain governance models

 <p>Late Q2 of 2019</p>	<ul style="list-style-type: none">- Blockchain Nervous System (Neuron with deposit, Proposals, Evaluation)- Economics, Policy, Protocol, Client, even Change rewrite- Staking reward scaled by voting participant rate- Voting by itself or following (relationships are hidden)
	<ul style="list-style-type: none">- Time-lock DCR for staking with a mandatory voting- Treasury- Politeia (a web platform)- Consensus rule (DCP, new SW, upgrade, voting, activation)- Constitution

Off-chain vs. On-chain

On-chain governance

Decision making on chain (ex. representative/liquid democracy, futarchy)

Leaderless governance (why nakamoto remain anonymous)

In the very **early stage**

Bootstrapping problem (it takes a very long time to develop)

Off-chain vs. On-chain

why NOT On-chain

It's coercive (update should be Opt-in)

The danger of incentives and misaligned interests

(coin holders ≠ users of the protocol)

The dangers of direct democracy (Harvard, 2019)

Low-voter turnout (ex. DAO Carbonvote 4.5%)

Plutocracy (staked-based voting)

Off-chain vs. On-chain

Why On-chain

Opt-out (reduced hard fork risk & migration cost, it's a social consensus)

Aligned better than miner-based governance (Primary use case = money)
(coin holders ≈ users of the protocol)

More decentralized and democratic than foundation-led governance

Governance mechanism itself can be updated as well (quadratic voting)

Off-chain vs. On-chain

Why NOT On-chain

It's coercive (update should be Opt-in)

The danger of incentives and misaligned interests
(coin holders ≠ users of the protocol)

The dangers of direct democracy (Harvard, 2019)

Low-voter turnout (ex. DAO Carbonvote 4.5%)

Plutocracy (staked-based voting)

Why On-chain

Opt-out (reduced hard fork risk & migration cost, it's a social consensus)

Aligned better than miner-based governance (Primary use case = money)
(coin holders ≈ users of the protocol)

More decentralized and democratic than foundation-led governance

Governance mechanism itself can be updated as well (quadratic voting)

CHAPTER3 스마트 컨트랙트 이론

What is Smart Contract

What is smart contract? (ref #12)

Smart contract

A set of promises specified in digital form, including protocols within which the parties perform on these promises.

Why

Observability

Verifiability

Privity

Enforceability

Smart Contracts: Building Blocks for Digital Markets

Copyright (c) 1996 by Nick Szabo
permission to redistribute without alteration hereby granted

Glossary

(This is a partial rewrite of the article which appeared in Extropy #16)

Introduction

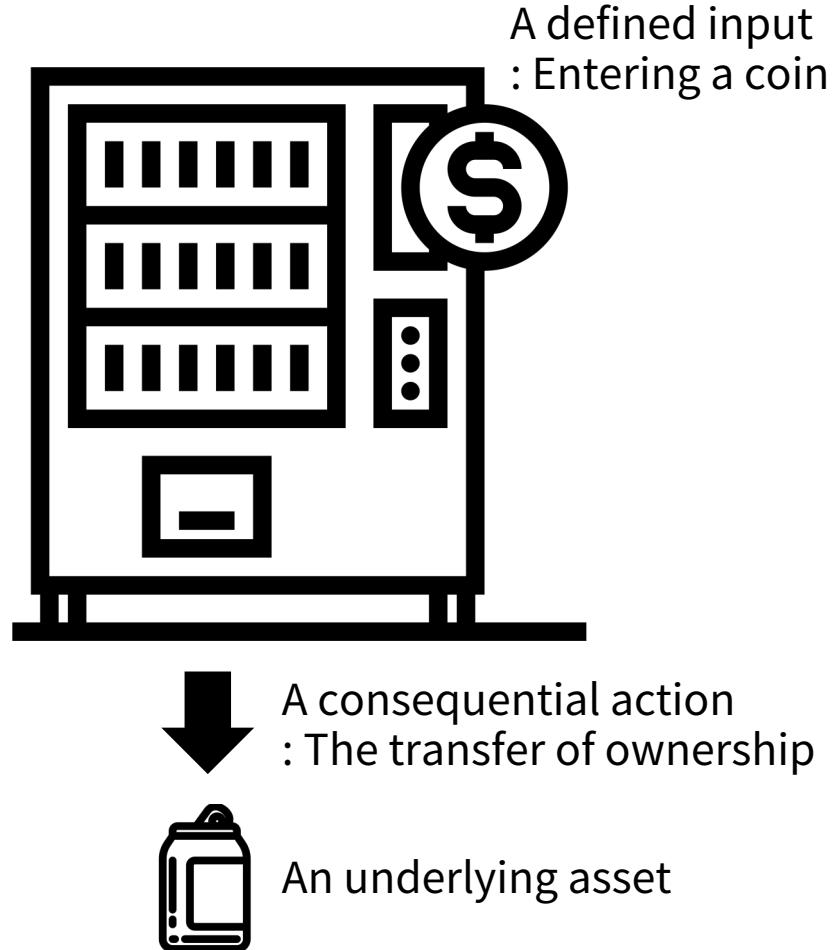
The contract, a set of promises agreed to in a "meeting of the minds", is the traditional way to formalize a relationship. While contracts are primarily used in business relationships (the focus of this article), they can also involve personal relationships such as marriages. Contracts are also important in politics, not only because of "social contract" theories but also because contract enforcement has traditionally been considered a basic function of capitalist governments.

Whether enforced by a government, or otherwise, the contract is the basic building block of a free market economy. Over many centuries of cultural evolution has emerged both the concept of contract and principles related to it, encoded into common law. [Algorithmic information theory](#) suggests that such evolved structures are often prohibitively costly to recompute. If we started from scratch, using reason and experience, it could take many centuries to redevelop sophisticated ideas like property rights that make the modern free market work [Hayek].

The success of the common law of contracts, combined with the high cost of replacing it, makes it worthwhile to both preserve and to make use of these principles where appropriate. Yet, the digital revolution is radically changing the kinds of relationships we can have. What parts of our hard-won legal tradition will still be valuable in the cyberspace era? What is the best way to apply these common law principles to the design of our on-line relationships?

Computers make possible the running of algorithms heretofore prohibitively costly, and networks the quicker

What is smart contract? (ref #13)



The Idea of Smart Contracts

Copyright (c) 1997 by Nick Szabo
permission to redistribute without alteration hereby granted

What is the meaning and purpose of "security"? How does it relate to the relationships we have? I argue that the formalizations of our relationships -- especially contracts -- provide the blueprint for ideal security.

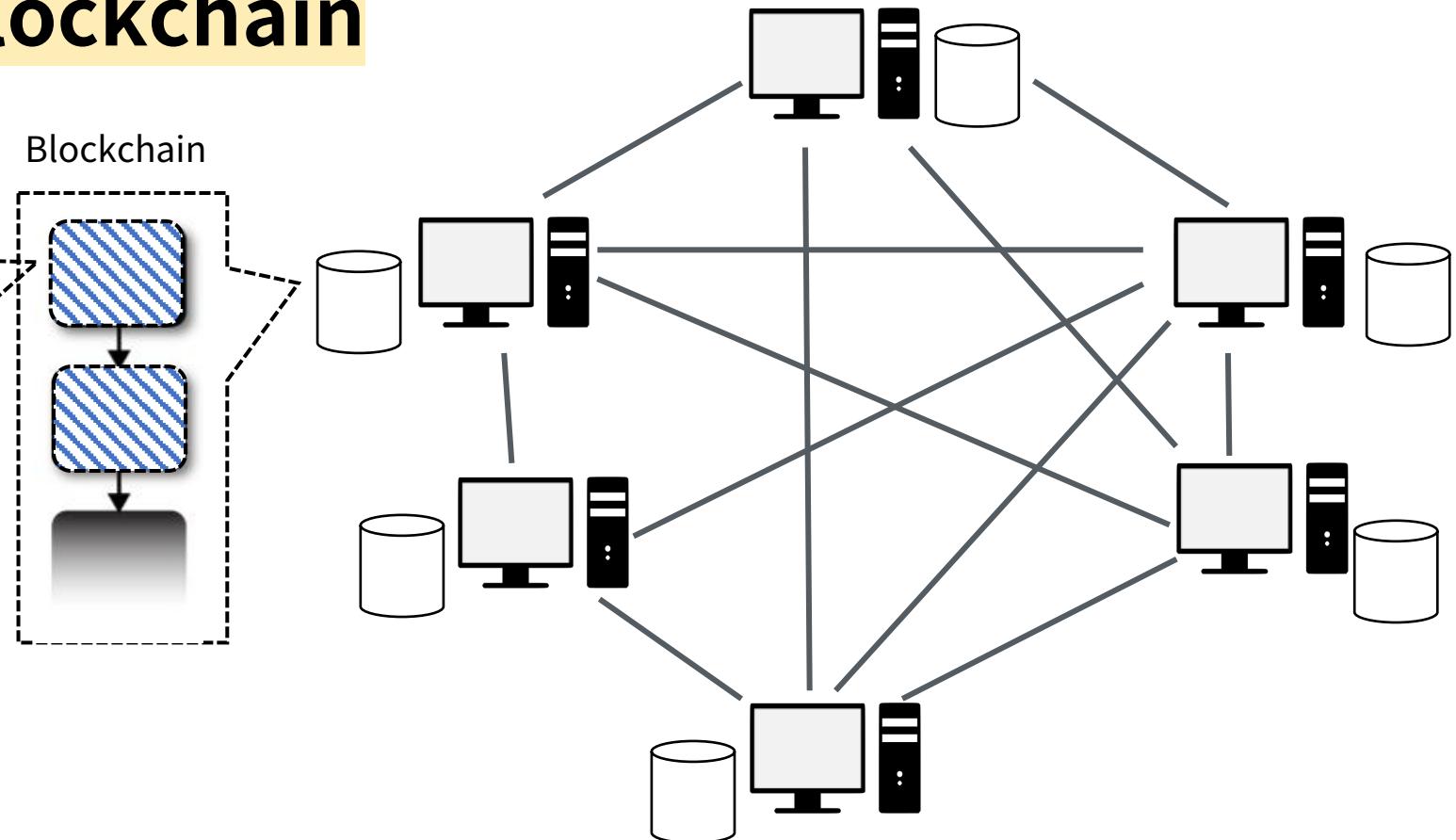
Many kinds of contractual clauses (such as collateral, bonding, delineation of property rights, etc.) can be embedded in the hardware and software we deal with, in such a way as to make breach of contract expensive (if desired, sometimes prohibitively so) for the breacher. A canonical real-life example, which we might consider to be the primitive ancestor of smart contracts, is the humble vending machine. Within a limited amount of potential loss (the amount in the till should be less than the cost of breaching the mechanism), the machine takes in coins, and via a simple mechanism, which makes a freshman computer science problem in design with finite automata, dispense change and product according to the displayed price. The vending machine is a contract with bearer: anybody with coins can participate in an exchange with the vendor. The lockbox and other security mechanisms protect the stored coins and contents from attackers, sufficiently to allow profitable deployment of vending machines in a wide variety of areas.

Smart contracts go beyond the vending machine in proposing to embed contracts in all sorts of property that is valuable and controlled by digital means. Smart contracts reference that property in a dynamic, often proactively enforced form, and provide much better observation and verification where proactive measures must fall short.

What is smart contract?

A computer program

running on top of blockchain



What is smart contract?

A computer program running on top of blockchain

Replicated (Across distributed nodes)

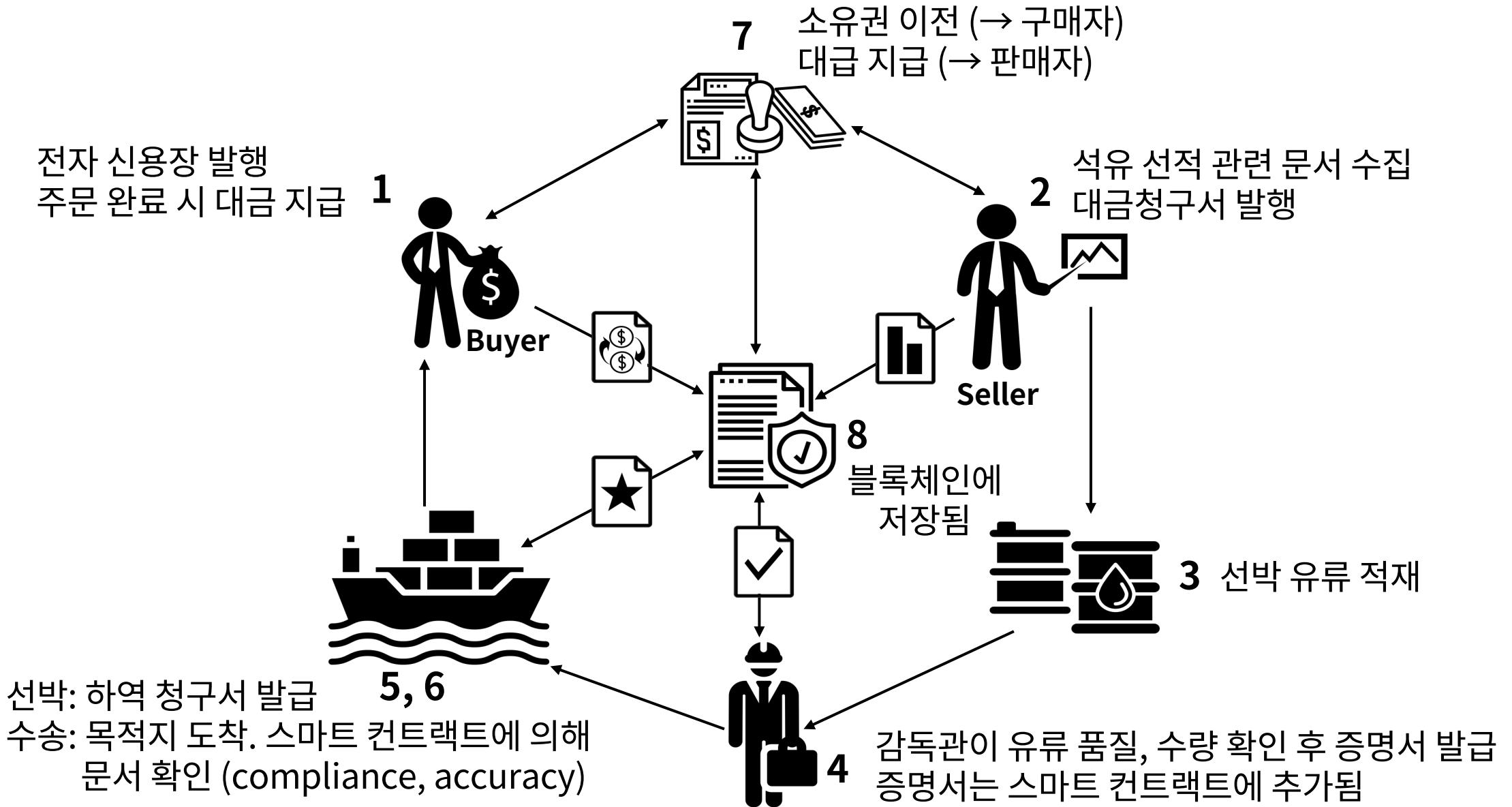
Immutable (Transparent)

Deterministic (The same for everyone)

Neutral & Passive (A bunch of codes)

Automated (Executed by nodes)

Smart Contracts in Trade Finance (WSJ)



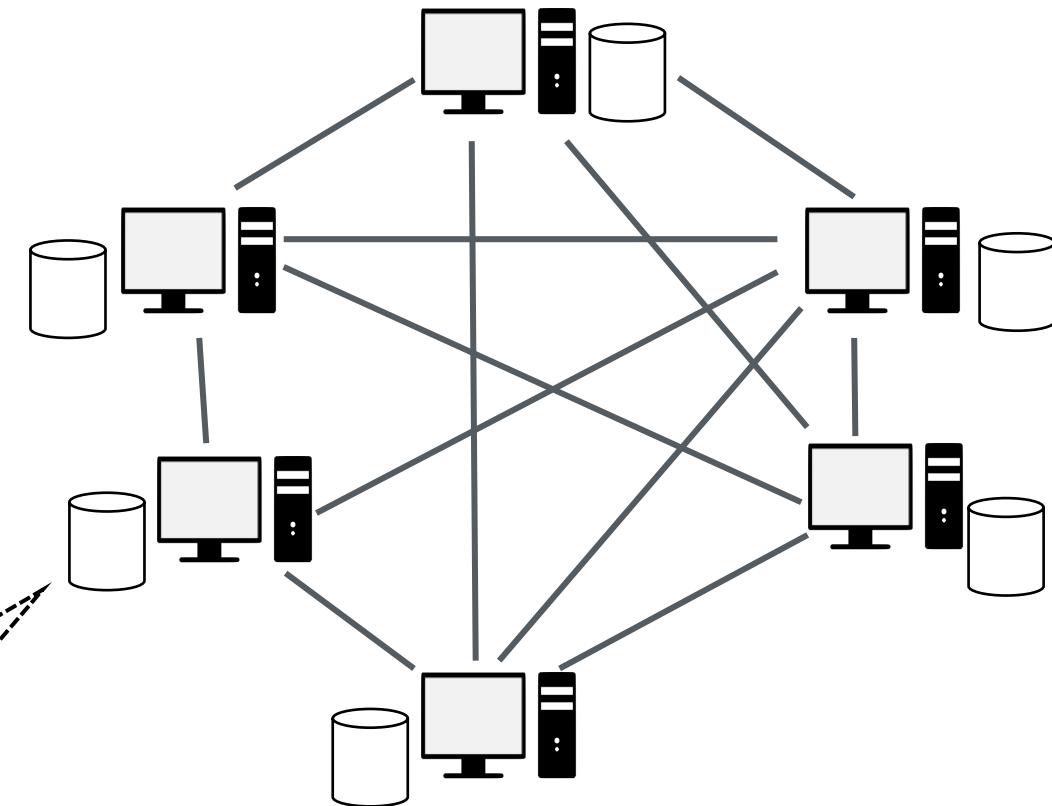
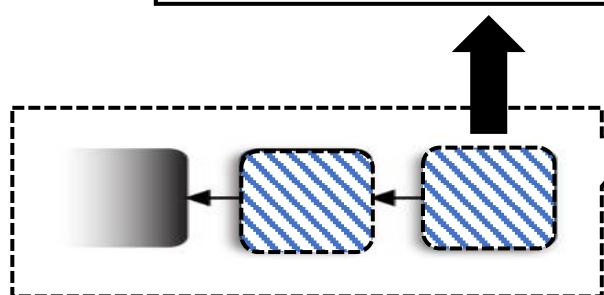
Smart Contracts in Account-based Model

Second Used Contract in Block 5

```
assets = [ (12, Sue), (34, Lee), ... ]  
sellCar(price) { IF A THEN B }  
buyCar(carID, price) { IF C THEN D }  
changeOwner(carID, newOwner) {  
    A new Owner owns a car of carID  
}
```

State

```
Alice: {  
    Balance: 6.1,  
    Counter: 12,  
    ...  
},  
carCompany: {  
    Balance: 100,  
    Contracts: [  
        CustomerSale,  
SecondUsed  
    ]  
},  
...  
}
```



Smart Contracts in Account-based Model

State

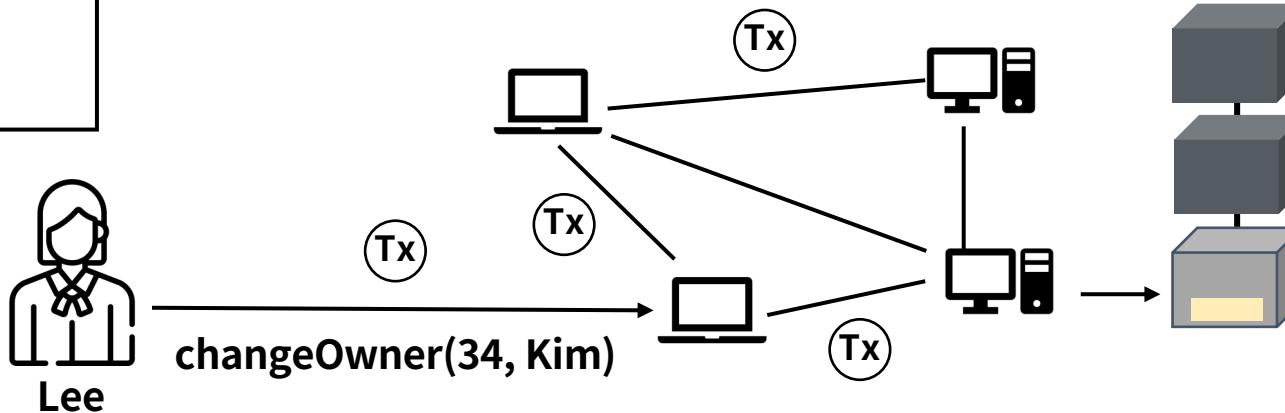
```
Alice: {  
    Balance: 6.1,  
    Counter: 12,  
    ...  
},  
carCompany: {  
    Balance: 100,  
    Contracts: [  
        CustomerSale,  
SecondUsed  
    ]  
},  
...  
},
```

Second Used Contract in Block 5

```
assets = [ (12, Sue), (34, Lee), ... ]  
  
sellCar(price) { IF A THEN B }  
  
buyCar(carID, price) { IF C THEN D }  
  
changeOwner(carID, newOwner) {  
    A new Owner owns a car of carID  
}
```

Second Used Contract in Block 9

```
assets = [ (12, Sue), (34, Kim), ... ]  
  
sellCar(price) { IF A THEN B }  
  
buyCar(carID, price) { IF C THEN D }  
  
changeOwner(carID, newOwner) {  
    A new Owner owns a car of carID  
}
```



```
From: Lee  
To: Second Used Contract  
Func: changeOwner  
Params: [ 34, Kim ]
```

Smart Contracts in Car Trading

Car Contract

```
assets = [ (123, Alice), (452, Bob), ... ]
```

```
sellCar(price) {  
    IF (msgSender has a car)  
    THEN (Lock this car)  
}
```

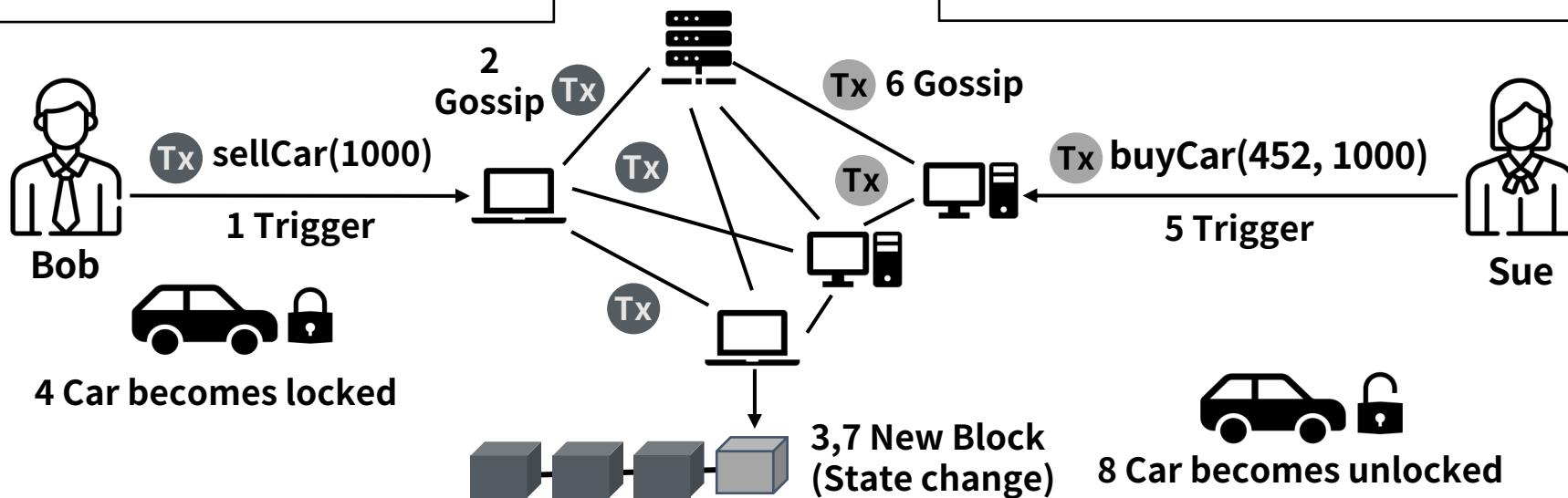
```
buyCar(carNum, price) {  
    IF (price is equal to carNum.price)  
    THEN (msgSender owns this car)  
}
```

Car Contract

```
assets = [ (123, Alice), (452, Sue), ... ]
```

```
sellCar(price) {  
    IF (msgSender has a car)  
    THEN (Lock this car)  
}
```

```
buyCar(carNum, price) {  
    IF (price is equal to carNum.price)  
    THEN (msgSender owns this car)  
}
```



Tezos SW Architecture

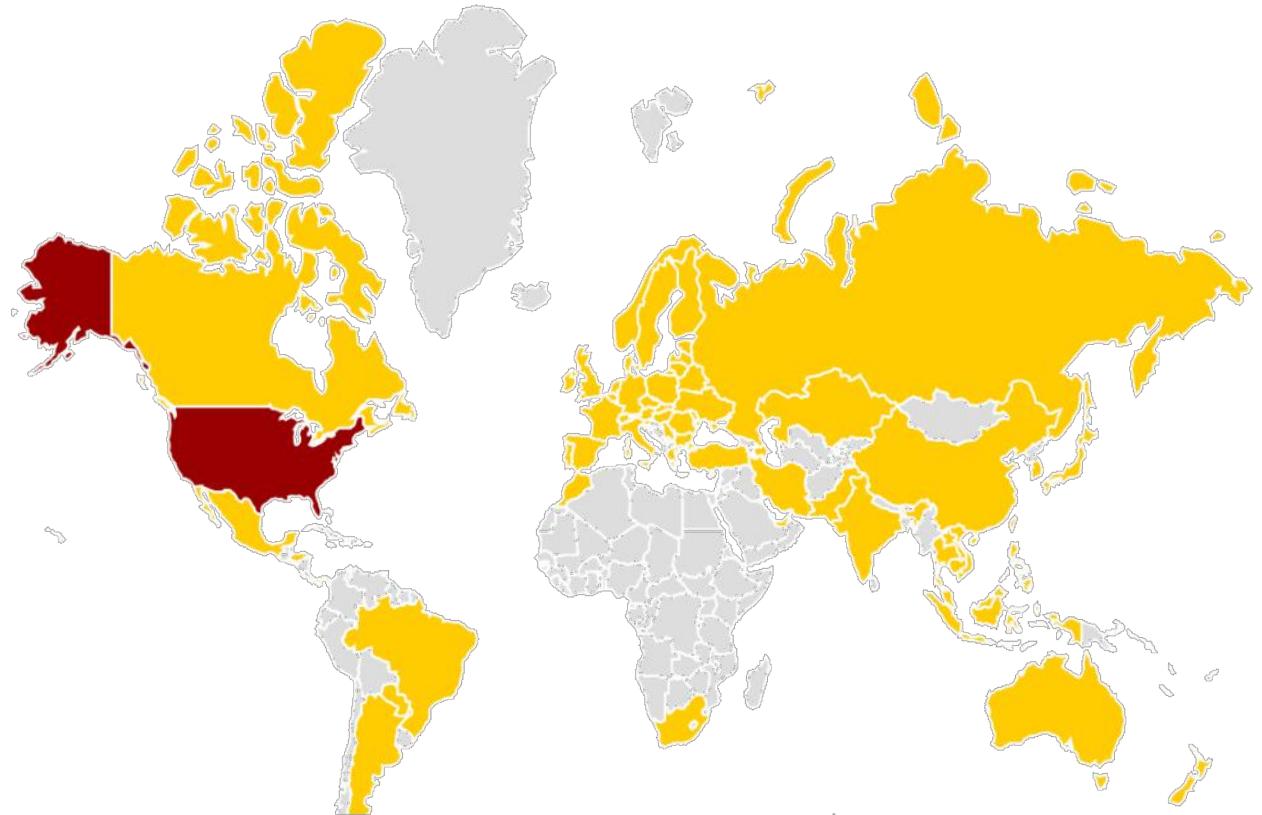
Tezos Network*

Mainnet (471 Bakers / 9,159 Nodes)

Live network

Production system

Real Tezzies



Alphanet (19 Bakers / 2,294 Nodes)

Test network

Support all updates (staging)

Fake Tezzies

Zeronet (12 Bakers / 114 Nodes)

Test network

Smaller (Cycle, intervals, ...)

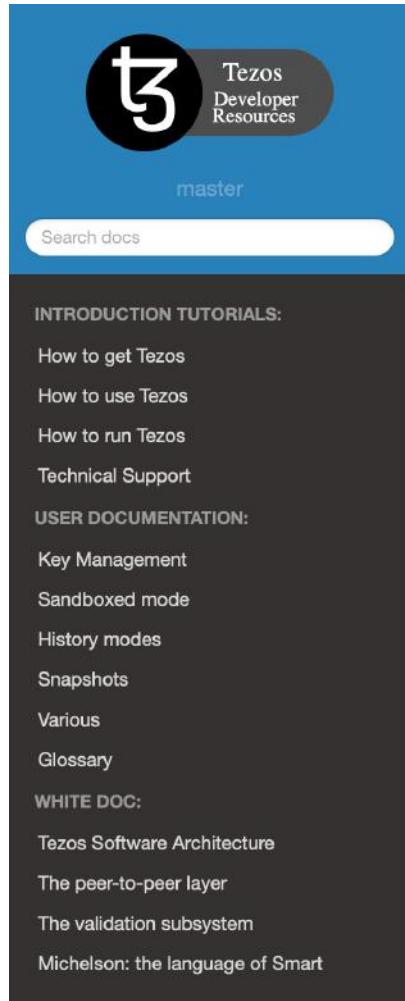
For core developers

* As of 19.06.17

Running Alphanet (ref #14)

알파넷 설치

Docker Image
Installing from source



Docs » Welcome to the Tezos Developer Documentation!

[View page source](#)

Welcome to the Tezos Developer Documentation!

The Project

Tezos is a distributed consensus platform with meta-consensus capability. Tezos not only comes to consensus about the state of its ledger, like Bitcoin or Ethereum. It also attempts to come to consensus about how the protocol and the nodes should adapt and upgrade.

- Developer documentation is available online at <https://tezos.gitlab.io/master>. The documentation is automatically generated for the master branch, the [main network \(mainnet\)](#) and the [test network \(alphanet\)](#). Make sure you are consulting the right version.
- The website <https://tezos.com/> contains more information about the project.
- All development happens on GitLab at <https://gitlab.com/tezos/tezos>

The source code of Tezos is placed under the MIT Open Source License.

The Community

- The website of the [Tezos Foundation](#).
- [Tezos sub-reddit](#) is an important meeting point of the community.
- Several community-built block explorers are available:
 - <https://tzscan.io>
- A few community-run websites collect useful Tezos links:

Running Alphanet

Installing Tezos SW from source

Being developed for Linux x86_64, mostly for Debian/Ubuntu and Archlinux.

Currently reported to work at:

- macOS/x86_64
- Linux/armv7h (32bit) (Raspberry Pi3, etc.)
- Linux/aarch64 (64bit) (Raspberry Pi3, etc.)

Hardware specification

- RAM: 4GB (8GB recommended)
- Storage: 20~40GB (SSD is strongly recommended)
- Good internet connection

Running Alphanet

Installing commands

```
$ sudo apt update
$ sudo apt install -y rsync git make m4 build-essential patch unzip wget
$ wget http://security.ubuntu.com/ubuntu/pool/main/b/bubblewrap/bubblewrap_0.2.1-1ubuntu0.1_amd64.deb
$ sudo dpkg -i bubblewrap_0.2.1-1ubuntu0.1_amd64.deb
$ wget https://github.com/ocaml/opam/releases/download/2.0.3/opam-2.0.3-x86_64-linux
$ sudo cp opam-2.0.3-x86_64-linux /usr/local/bin/opam
$ sudo chmod a+x /usr/local/bin/opam
$ git clone https://gitlab.com/tezos/tezos.git
$ cd tezos
$ git checkout alphanet
$ export OPAMNO=true
$ opam init --bare
$ export OPAMNO=false
$ sudo apt install -y libev-dev libgmp-dev pkg-config libhidapi-dev
$ make build-deps
$ eval $(opam env)
$ make
$ export PATH=~/tezos:$PATH
$ source ./src/bin_client/bash-completion.sh
$ export TEZOS_CLIENT_UNSAFE_DISABLE DISCLAIMER=Y
```

Running Alphanet (ref #15)

Shortcut (Automated script)

All commands are packed together into a bash script.

It will install dependencies, download the source code, and build the code called “make process”.

```
$ curl "https://gitlab.com/tezoskorea/quickstart/raw/master/tz_install.sh" | bash -s alphanet
```

For macOS, use this

```
$ curl "https://gitlab.com/tezoskorea/quickstart/raw/master/tz_install_mac.sh" | bash -s alphanet
```

It takes 10 ~ 15 minutes.

```
Info: creating file vendors/ocplib-resto/lib_resto-directory/dune-project with this contents:  
| (lang dune 1.8)  
| (name ocplib-resto-directory)
```

```
Info: creating file vendors/ocplib-resto/lib_resto-json/dune-project with this contents:  
| (lang dune 1.8)  
| (name ocplib-resto-json)
```

```
Finished!
```

```
ubuntu@ip-172-31-18-57:~$
```

Running Alphanet

Binaries (Executable files)

```
[ubuntu@ip-172-31-23-117:~/tezos$ ls -l
total 253932
-rw-rw-r-- 1 ubuntu ubuntu      1114 Jun 24 12:32 LICENSE
-rw-rw-r-- 1 ubuntu ubuntu     5031 Jun 24 12:32 Makefile
-rw-rw-r-- 1 ubuntu ubuntu     1692 Jun 24 12:32 README.md
drwxrwxr-x 6 ubuntu ubuntu    4096 Jun 24 12:53 _build
drwxr-xr-x 9 ubuntu ubuntu    4096 Jun 24 12:42 _opam
-rw-rw-r-- 1 ubuntu ubuntu       13 Jun 24 12:32 active_protocol_versions
-rw-rw-r-- 1 ubuntu ubuntu    2419 Jun 24 12:32 contributing.md
drwxrwxr-x 12 ubuntu ubuntu   4096 Jun 24 12:32 docs
-rw-rw-r-- 1 ubuntu ubuntu      193 Jun 24 12:32 dune
-rw-rw-r-- 1 ubuntu ubuntu       16 Jun 24 12:53 dune-project
-rw-rw-r-- 1 ubuntu ubuntu       16 Jun 24 12:32 dune-workspace
drwxrwxr-x 2 ubuntu ubuntu    4096 Jun 24 12:32 emacs
drwxrwxr-x 5 ubuntu ubuntu    4096 Jun 24 12:32 scripts
drwxrwxr-x 36 ubuntu ubuntu   4096 Jun 24 12:32 src
drwxrwxr-x 9 ubuntu ubuntu    4096 Jun 24 12:32 tests_python
-rwxrwxr-x 1 ubuntu ubuntu 31047200 Jun 24 12:55 tezos-accuser-004-Pt24m4xi
-rwxrwxr-x 1 ubuntu ubuntu 36000992 Jun 24 12:55 tezos-admin-client
-rwxrwxr-x 1 ubuntu ubuntu 31047168 Jun 24 12:55 tezos-baker-004-Pt24m4xi
-rwxrwxr-x 1 ubuntu ubuntu 36019608 Jun 24 12:55 tezos-client
-rwxrwxr-x 1 ubuntu ubuntu 31047216 Jun 24 12:55 tezos-endorser-004-Pt24m4xi
-rwxrwxr-x 1 ubuntu ubuntu 49637424 Jun 24 12:55 tezos-node
-rwxrwxr-x 1 ubuntu ubuntu 26499448 Jun 24 12:55 tezos-protocol-compiler
-rwxrwxr-x 1 ubuntu ubuntu 18645920 Jun 24 12:55 tezos-signer
drwxrwxr-x 13 ubuntu ubuntu   4096 Jun 24 12:32 vendors
```

Tezos SW Components

tezos-node

Network shell, The Tezos daemon

tezos-client

A command-line client and basic wallet

tezos-admin-client

Administration tool for the node

tezos-{baker, endorser, accuser}-004-Pt24m4xi

Daemons to bake, endorse and accuse

tezos-signer

A client to remotely sign operations or blocks

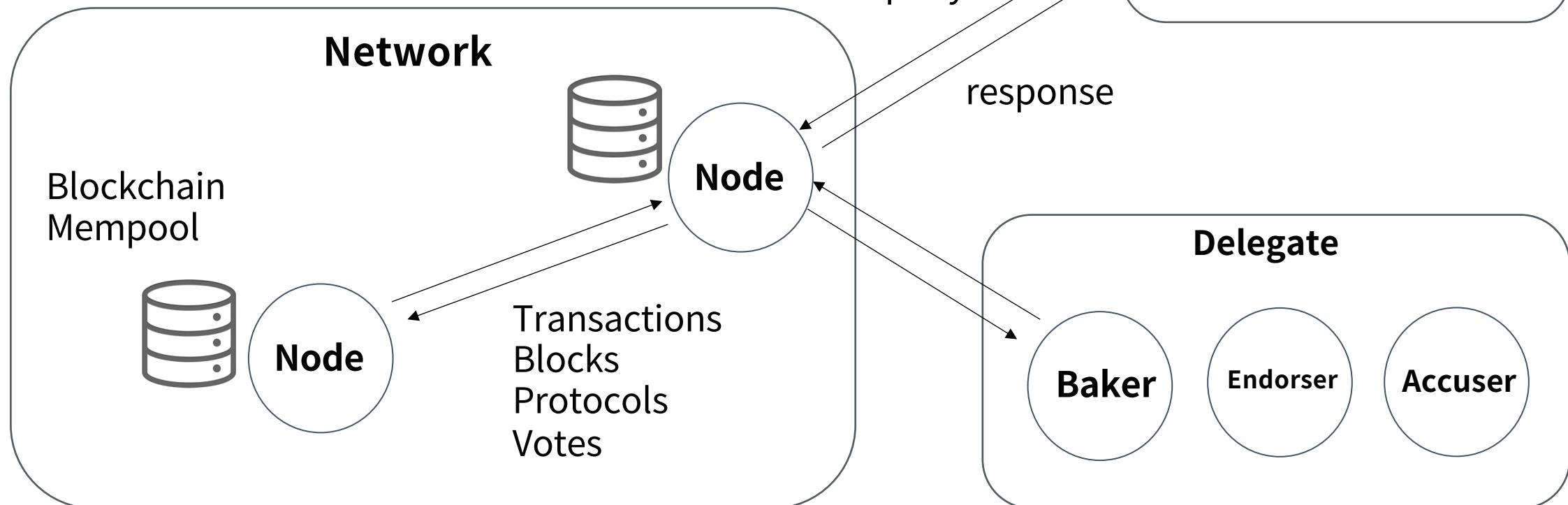
Tezos SW Components

Node (Network Shell, API end point)

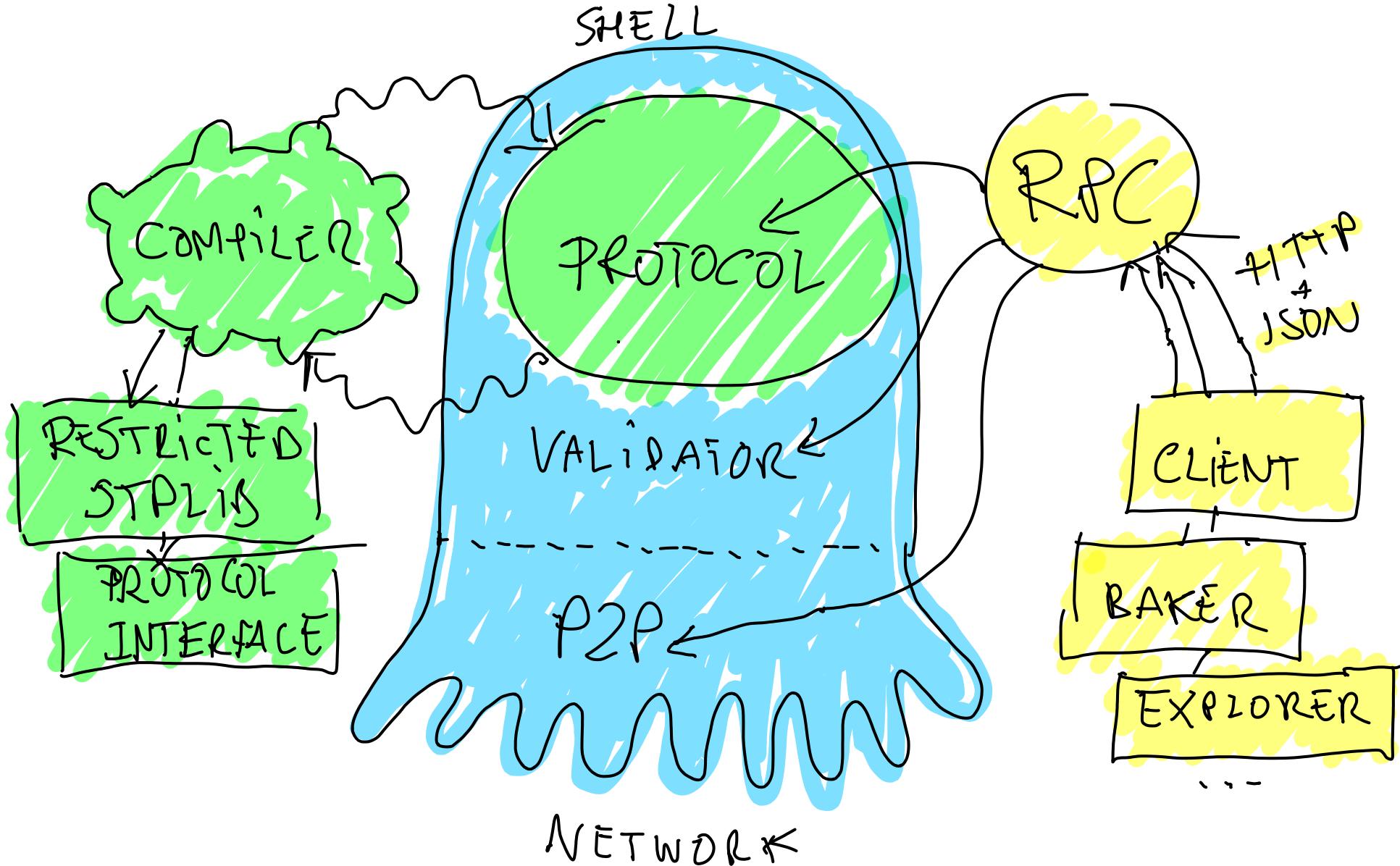
Client (Library + built-in wallet)

Admin Client (Administration tool for the node)

Delegate (Baker, Endorser, Accuser)



Tezos SW Architecture (ref #16)



Blockchain Protocols

Network protocol

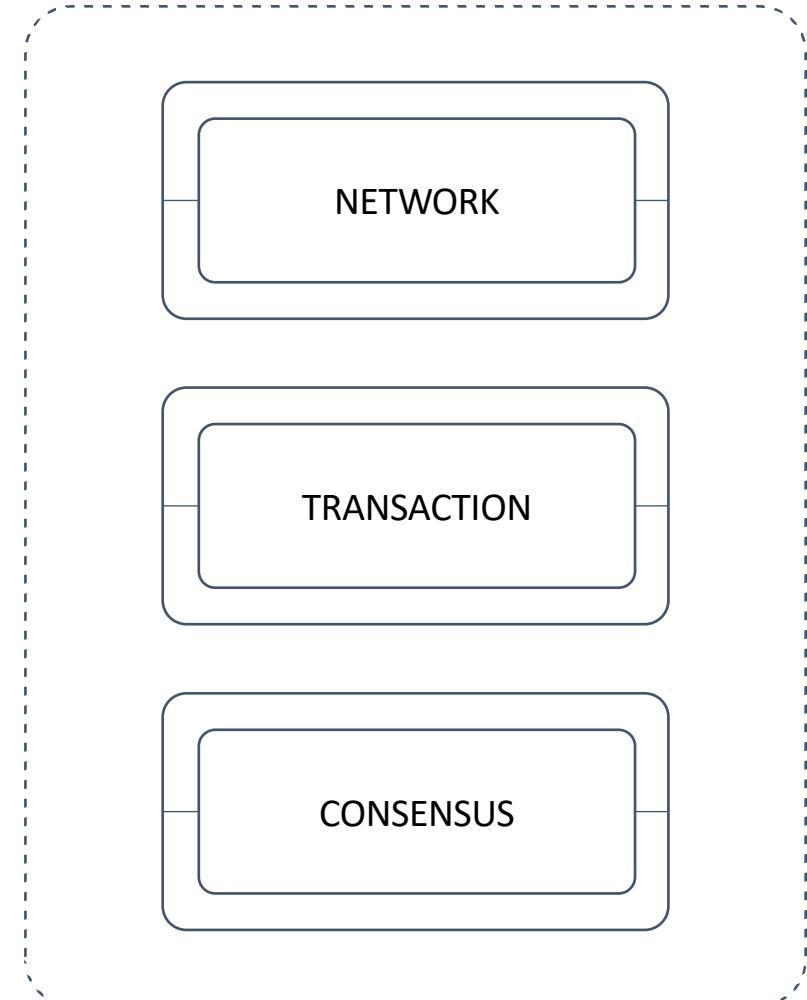
Transactions / Blocks download and broadcast
P2P network management (discover, add, remove)
Relatively uncontroversial

Transaction protocol

What makes transactions valid
More controversial

Consensus protocol

The way consensus is built around the one chain
The most difficult to change



Blockchain Protocols

Network protocol

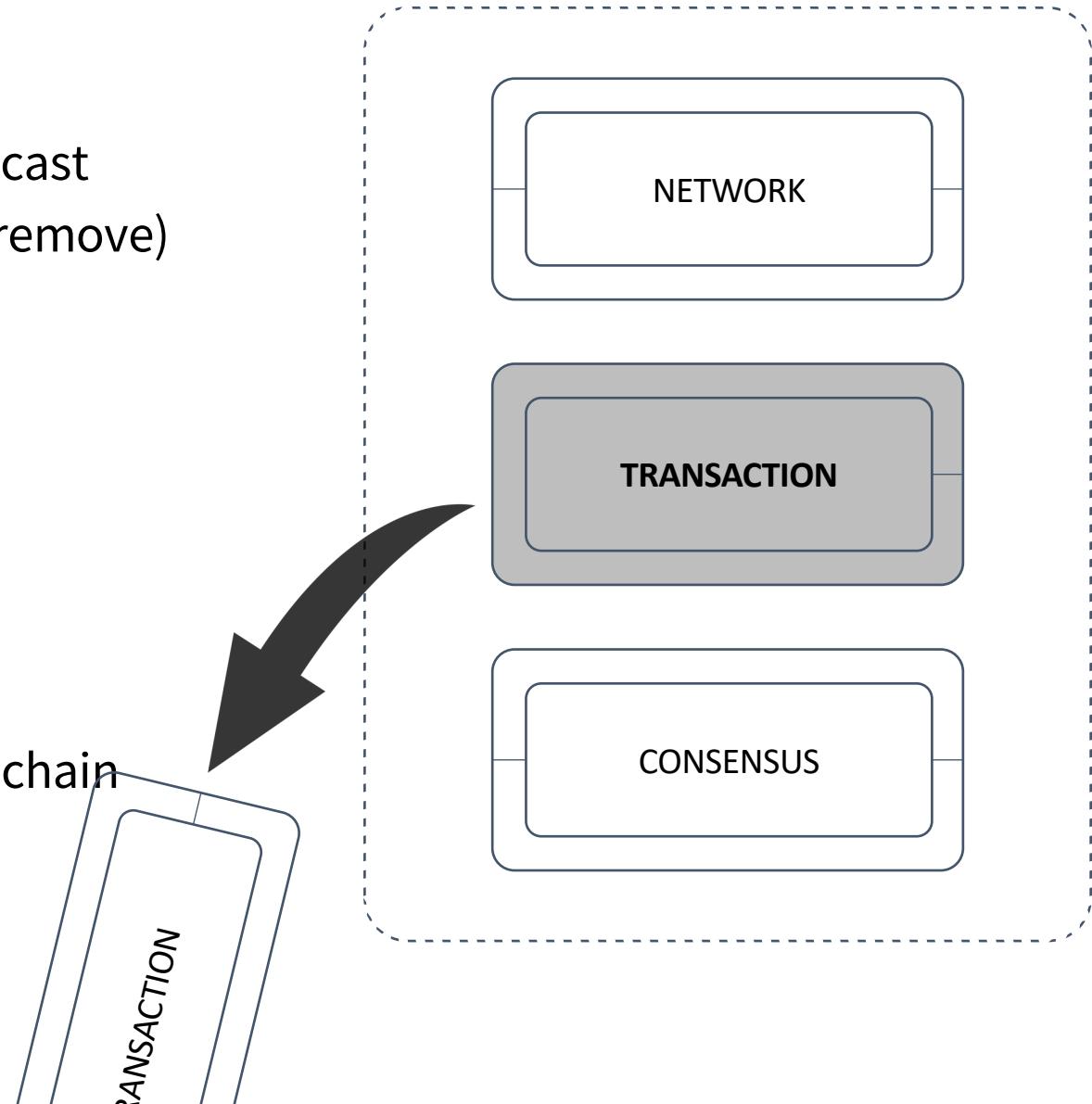
Transactions / Blocks download and broadcast
P2P network management (discover, add, remove)
Relatively uncontroversial

Transaction protocol

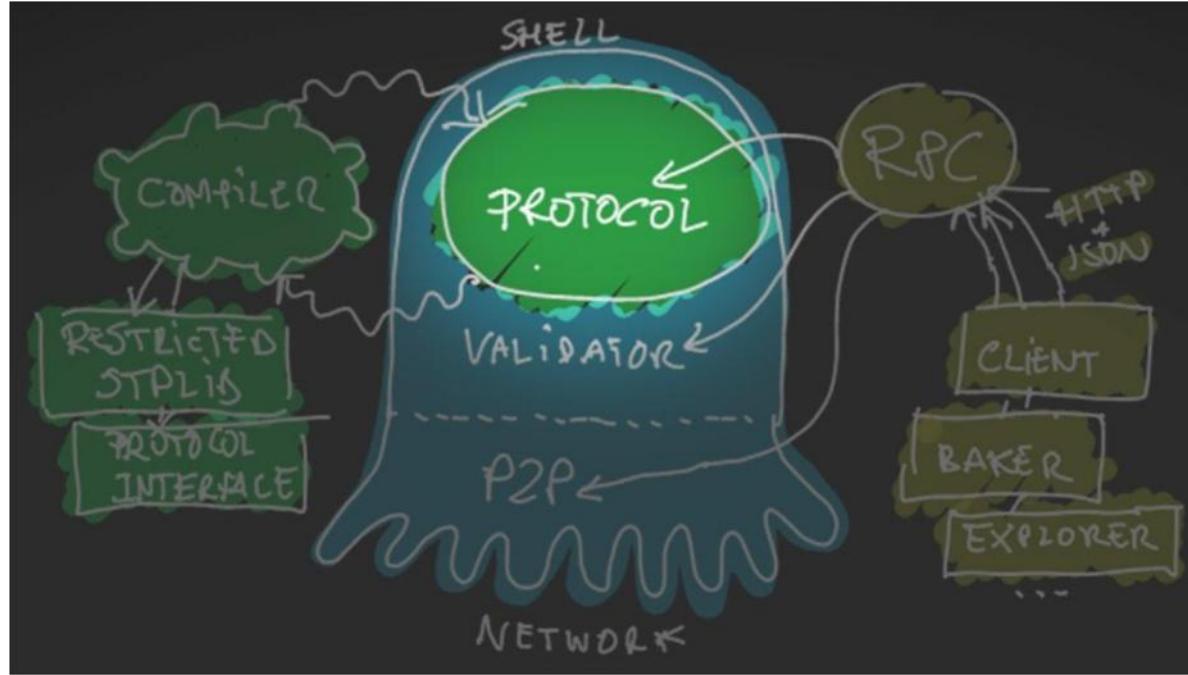
What makes transactions valid
More controversial

Consensus protocol

The way consensus is built around the one chain
The most difficult to change



Tezos SW Architecture



Protocol

Tx protocol + consensus protocol

Also called as the economic protocol, block chain protocol, protocol, proto

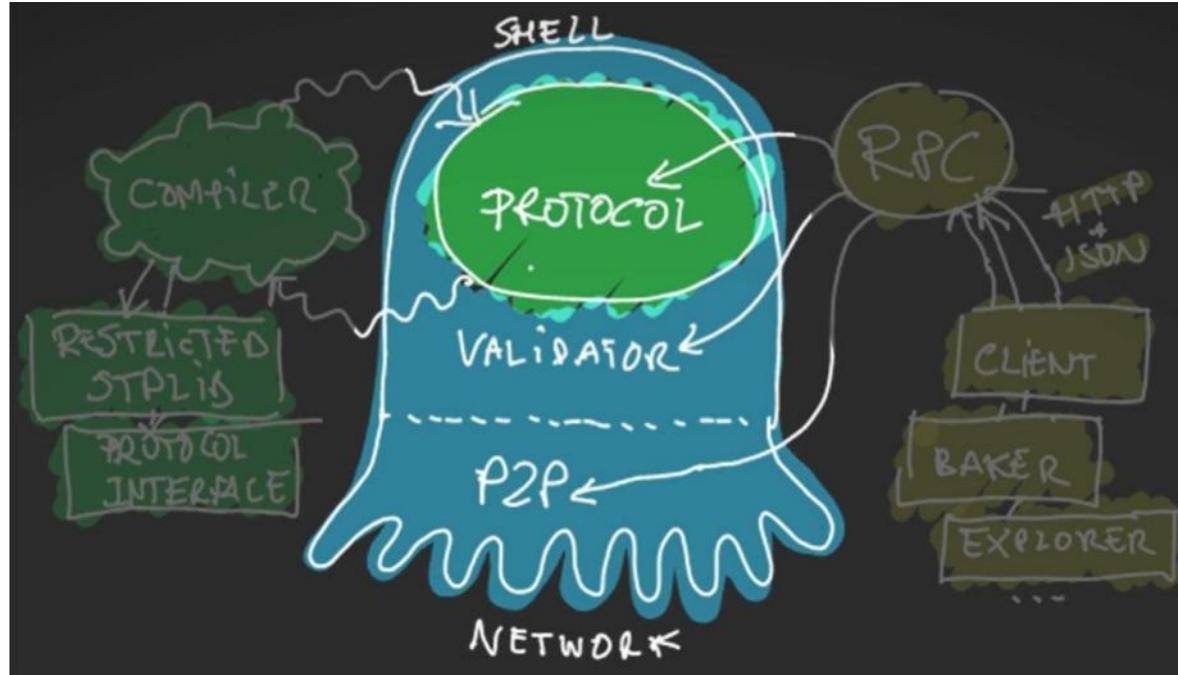
Self-amendment (plug-ins)

Interpreting operations including tx and blocks

Given only one chain from the shell

Network agnostic

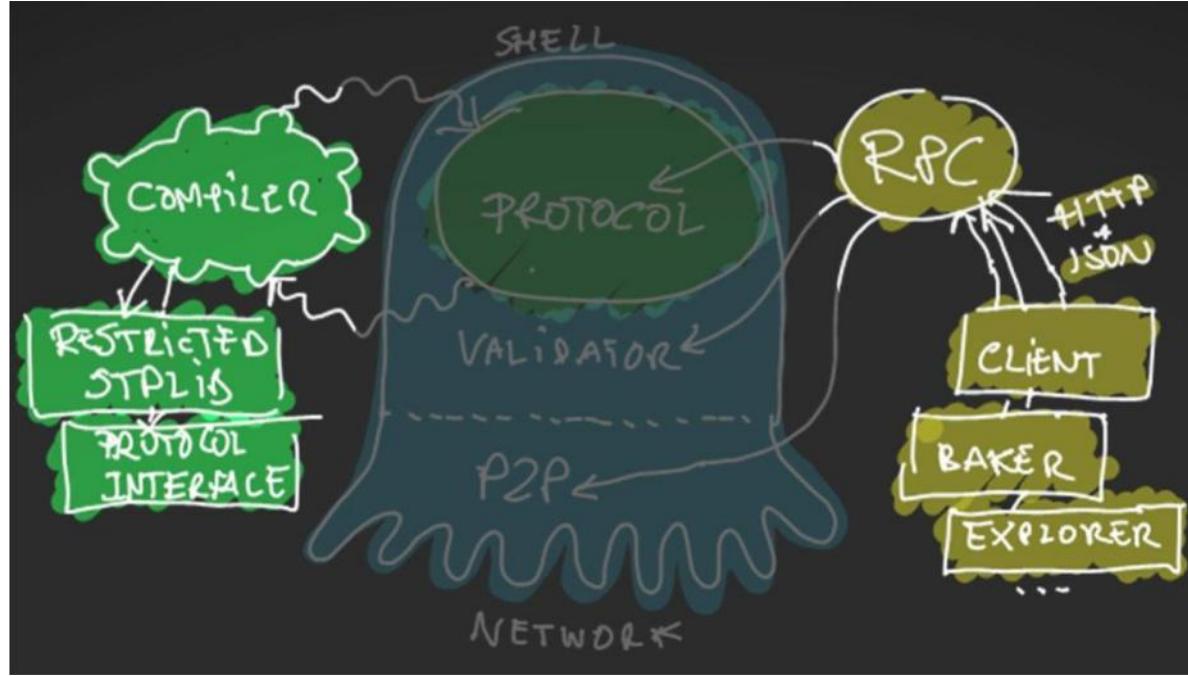
Tezos SW Architecture



Network shell

- Running the gossip network + Updating the context
- Validator + P2P Layer + Others (the storage of blocks and the state of the ledger)
- Maintaining the best chain of highest absolute score (validator)
- Managing P2P pool (detects, connects, bans)
- Aware of 3 type of objects: transactions(operation), blocks + protocol

Tezos SW Architecture



RPC

The way to interact with the node for the client and third parties
JSON and HTTP

Compiler

Type checking of the protocol's main module
Statically enforced by sandboxing

Getting started with Tezos Node

Main functions

Running the gossip network

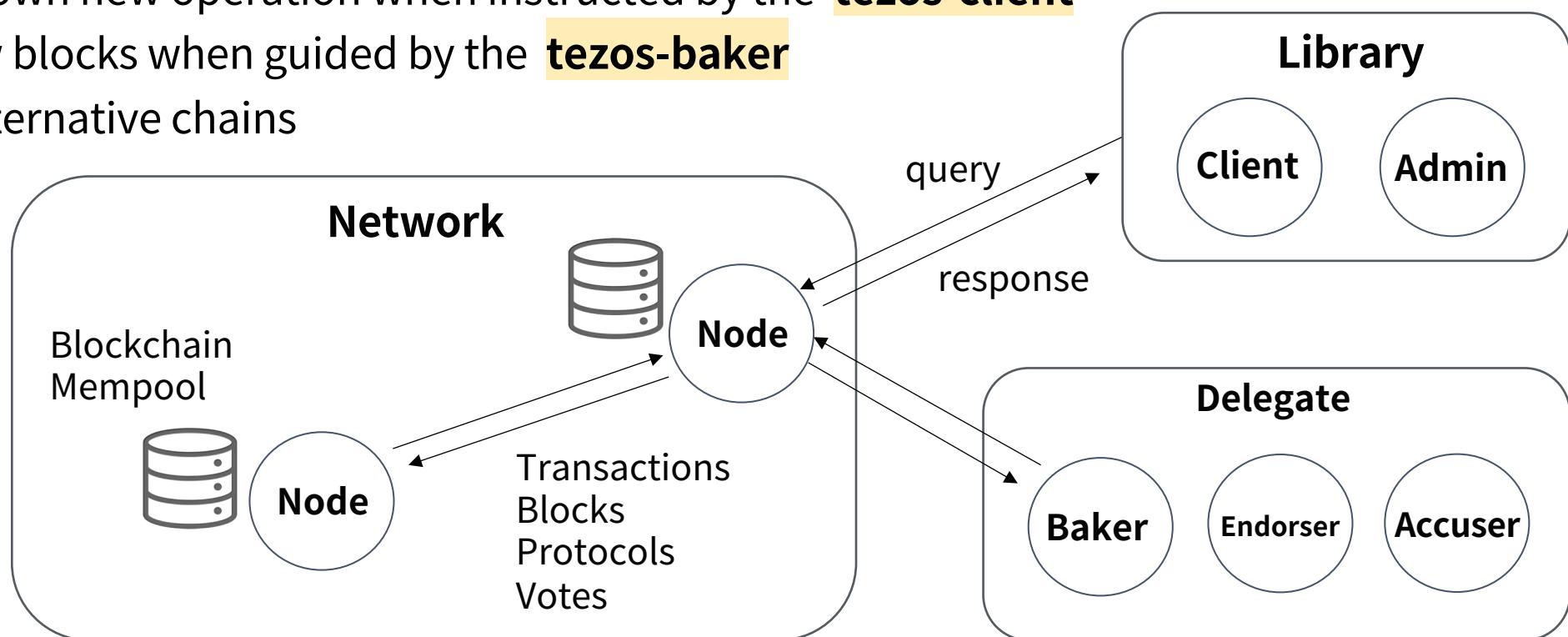
Updating the context

Also

Injecting its own new operation when instructed by the **tezos-client**

Sending new blocks when guided by the **tezos-baker**

Managing alternative chains



Getting started with Tezos Node

Node identity

Let other nodes know itself

Detected by other nodes in the Tezos network.

```
$ ./tezos-node identity generate
```

```
[ubuntu@ip-172-31-18-57:~/tezos$ ./tezos-node identity generate
Generating a new identity... (level: 26.00)
Stored the new identity (idsCcz7pFjbgRD6pfixWev1418znik) into '/home/ubuntu/.tezos-node/identity.json'.
ubuntu@ip-172-31-18-57:~/tezos$ ]
```

The identity file in the node's storage.

By default, all block-chain data is stored under **\$HOME/.tezos-node/**

```
[ubuntu@ip-172-31-18-57:~/tezos$ ls ~/.tezos-node/
identity.json  version.json ]
```

Getting started with Tezos Node

Running Node

JSON interface is the only interface to the node, but disabled by default.

It can be enabled for the clients to communicate with the node. (**Default port is 8732**)

```
$ ./tezos-node run --rpc-addr 127.0.0.1
```

Syncing

After the node is executed, it starts to find peers and downloads blocks from those peers.

```
[ubuntu@ip-172-31-23-117:~/tezos$ ./tezos-node run --rpc-addr 127.0.0.1 ]  
Jul 1 04:30:56 - node.main: Starting the Tezos node...  
Jul 1 04:30:56 - node.main: No local peer discovery.  
Jul 1 04:30:56 - node.main: Peer's global id: idrVn8WZxjWk7Z1TcHTtUN9jwV  
UkPz  
Jul 1 04:30:56 - main: shell-node initialization: bootstrapping  
Jul 1 04:30:56 - main: shell-node initialization: p2p_maintain_started  
Jul 1 04:30:56 - validator.block: Worker started  
Jul 1 04:30:56 - validation_process.sequential: Initialized  
Jul 1 04:30:56 - node.validator: activate chain NetXgtSLGNJvNye
```

Getting started with Tezos Node

Node is running

```
[ubuntu@ip-172-31-18-57:~/tezos$ ./tezos-node run --rpc-addr 127.0.0.1
Jul 25 10:23:19 - node.main: Starting the Tezos node...
Jul 25 10:23:19 - node.main: No local peer discovery.
Jul 25 10:23:19 - node.main: Peer's global id: idsCcz7pFjbgRD6pfixWev1418znik
Jul 25 10:23:19 - main: shell-node initialization: bootstrapping
Jul 25 10:23:19 - main: shell-node initialization: p2p_maintain_started
Jul 25 10:23:19 - validator.block: Worker started
Jul 25 10:23:19 - validation_process.sequential: Initialized
Jul 25 10:23:19 - node.validator: activate chain NetXgtSLGNJvNye
Jul 25 10:23:19 - validator.chain_1: Worker started for NetXgtSLGNJvN
Jul 25 10:23:19 - p2p.maintenance: Too few connections (0)
Jul 25 10:23:19 - node.chain_validator: no prevalidator filter found for protocol 'Ps6mwMrF2ER2'
Jul 25 10:23:19 - prevalidator.NetXgtSLGNJvN.Ps6mwMrF2ER2_1: Worker started for NetXgtSLGNJvN.Ps6mwMrF2ER2
Jul 25 10:23:19 - node.main: Starting a RPC server listening on ::ffff:127.0.0.1:8732.
Jul 25 10:23:19 - node.main: The Tezos node is now running!
Jul 25 10:23:20 - validator.peer_1: Worker started for NetXgtSLGNJvN:idrm1KhfdmV9
Jul 25 10:23:20 - validator.peer_2: Worker started for NetXgtSLGNJvN:idtdNBn7HYqM
Jul 25 10:23:20 - validator.peer_3: Worker started for NetXgtSLGNJvN:idsohLhiCYYW
Jul 25 10:23:20 - validator.peer_4: Worker started for NetXgtSLGNJvN:idr2piupJd1r
Jul 25 10:23:20 - validator.peer_5: Worker started for NetXgtSLGNJvN:idse7w6uFvRy
Jul 25 10:36:19 - validator.block: Block BMPtRJqFGQJRTfn8bXQR2grLE1M97XnUmG5vgjHMW7St1Wub7Cd successfully validated
Jul 25 10:36:19 - validator.block: Pushed: 2019-07-25T10:36:17-00:00, Treated: 2019-07-25T10:36:17-00:00, Completed: 2019-07-25T10:36:19-00:00
Jul 25 10:36:19 - node.chain_validator: no prevalidator filter found for protocol 'PsddFKi32cMJ'
Jul 25 10:36:19 - prevalidator.NetXgtSLGNJvN.PsddFKi32cMJ_1: Worker started for NetXgtSLGNJvN.PsddFKi32cMJ
Jul 25 10:36:19 - prevalidator.NetXgtSLGNJvN.Ps6mwMrF2ER2_1: Worker terminated [NetXgtSLGNJvN.Ps6mwMrF2ER2]
Jul 25 10:36:19 - validator.chain_1: Update current head to BMPtRJqFGQJRTfn8bXQR2grLE1M97XnUmG5vgjHMW7St1Wub7Cd (fitness 00::0000000000000001), same branch
Jul 25 10:36:19 - validator.chain_1: Pushed: 2019-07-25T10:36:19-00:00, Treated: 2019-07-25T10:36:19-00:00, Completed: 2019-07-25T10:36:19-00:00
Jul 25 10:36:25 - validator.block: Block BLwKksYwrxt39exDei7yi47h7aMcVY2kZMZhTwEEoSUwToQUIDV successfully validated
Jul 25 10:36:25 - validator.block: Pushed: 2019-07-25T10:36:25-00:00, Treated: 2019-07-25T10:36:25-00:00, Completed: 2019-07-25T10:36:25-00:00
Jul 25 10:36:25 - prevalidator.NetXgtSLGNJvN.PsddFKi32cMJ_1: switching to new head BLwKksYwrxt39exDei7yi47h7aMcVY2kZMZhTwEEoSUwToQUIDV
```

Getting started with Tezos **Client**

Library

Interacting with the node by **querying** its status or **asking** the node to perform some actions
A built-in wallet

To check the timestamp of the head of the chain (UTC)

```
$ ./tezos-client get timestamp
```

```
[ubuntu@ip-172-31-20-138:~$ tezos-client get timestamp
2019-05-31T10:04:27Z
[ubuntu@ip-172-31-20-138:~$ tezos-client get timestamp
2019-05-31T10:06:27Z
```