DFINITY

Intelligent Decentralized Cloud

EDCON 18th February 2017 (V1.0)

DFINITY

Experimental Ethereum Sister Network

ethereum

Casper

Extreme availability

The Code is Law

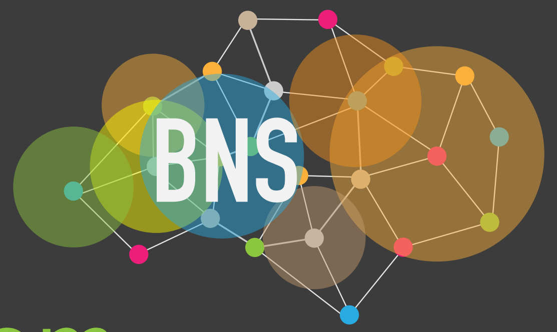
Governance by community

DFINITY

crypto:3

Speed, scale-out...

The AI is Law



Blockchain Nervous System

Experimental Ethereum Sister Network

ethereum

Derived & maintains compatibility

→

“EVM Singularity”

DFINITY

Casper

Extreme availability

The Code is Law

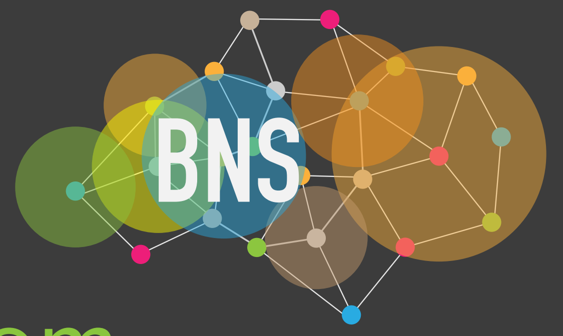
Governance by community

crypto:3

Speed, scale-out…

The AI is Law

Blockchain Nervous System



Experimental Ethereum Sister Network

ethereum

DFINITY

New techniques from work dating back to 2014

Casper ← crypto:3

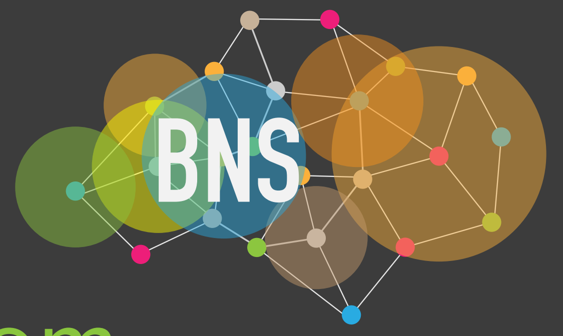
Extreme availability

Speed, scale-out...

The Code is Law

Governance by community

The AI is Law



Blockchain Nervous System

Experimental Ethereum Sister Network

ethereum

DFINITY

Casper

Extreme availability

crypto:3

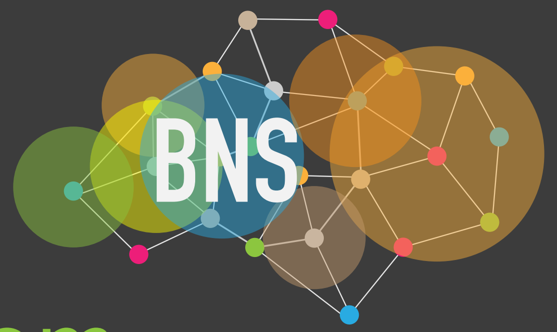
Speed, scale-out...

DEFINING DIFFERENCE

The Code is Law

Governance by community

The AI is Law



Blockchain Nervous System

Everything subject to distributed intelligence.
DFINITY is not a conventional blockchain...

TODAY WE HAVE LIMITED TIME

Let’s examine a crucial crypto:3 technique

Delivers finality 50X faster than today…

“Threshold Relay in 10 minutes”

Boneh-Lynn-Shacham Signatures (BLS)

UNIQUE DETERMINISTIC THRESHOLD SIGNATURE SCHEME

SUPPORTING DISTRIBUTED KEY GENERATION

Parameters

- Two groups G_1, G_2 of prime order r (on two elliptic curves)
- Generators $Q_1 \in G_1, Q_2 \in G_2$
- Bi-linear pairing $e : G_1 \times G_2 \mapsto G_T$

Key Generation

- Secret key: $x \bmod r$
- Public key: $P = xQ_2 \in G_2$

Signing

- Message hashed to $H(m) \in G_1$
- Signature: $s = xH(m) \in G_1$

Verification $e(s, Q_2) = e(H(m), P)$?

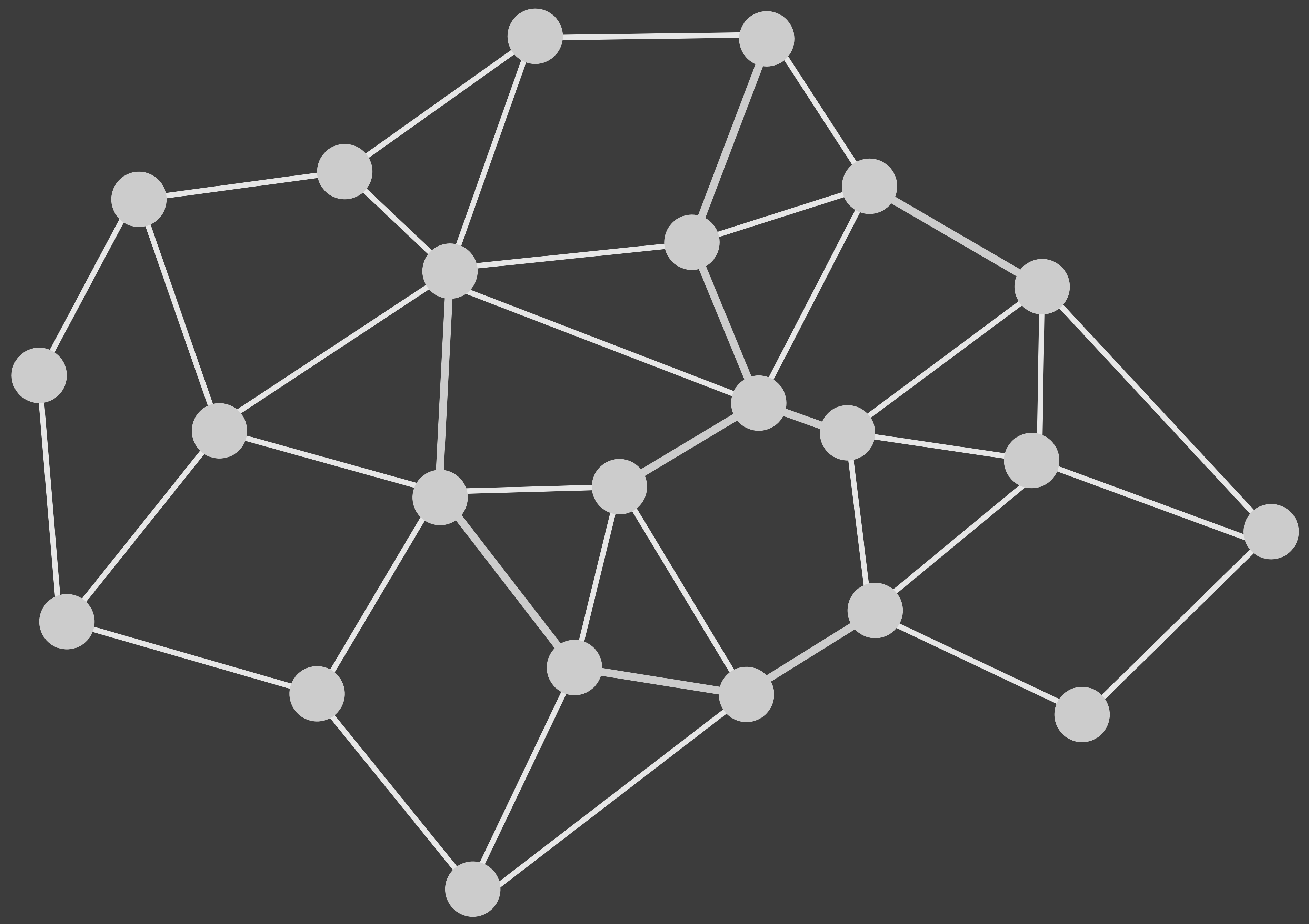
BLS, 2003

1

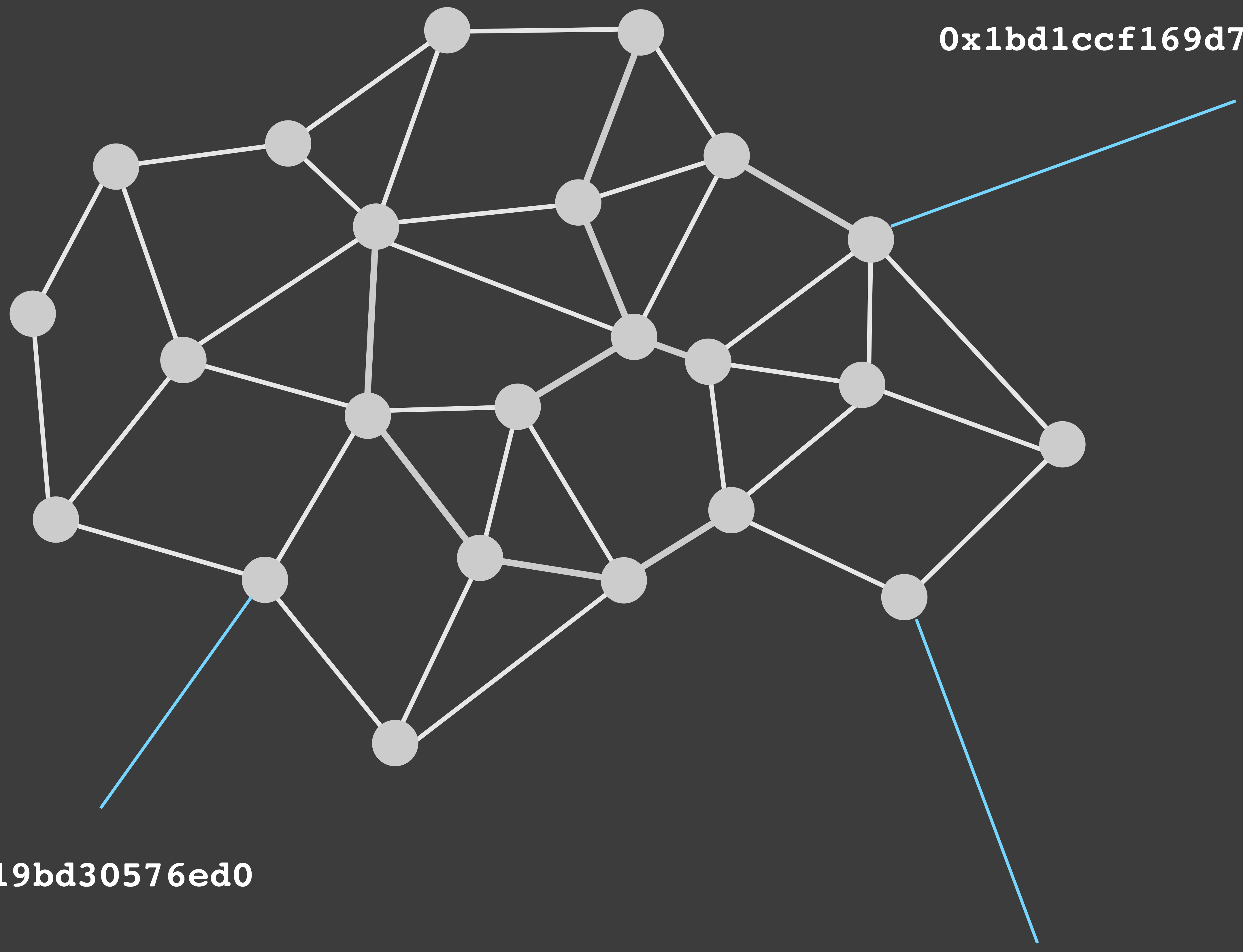
Basic Threshold Relay

Incorruptible, unmanipulable and unpredictable randomness

A vast peer-to-peer broadcast network of mining clients...



That are registered on the ledger



0x1bd1ccf169d755306e077b38cb9aeae28e245351

DEPOSIT: 1000 DFN

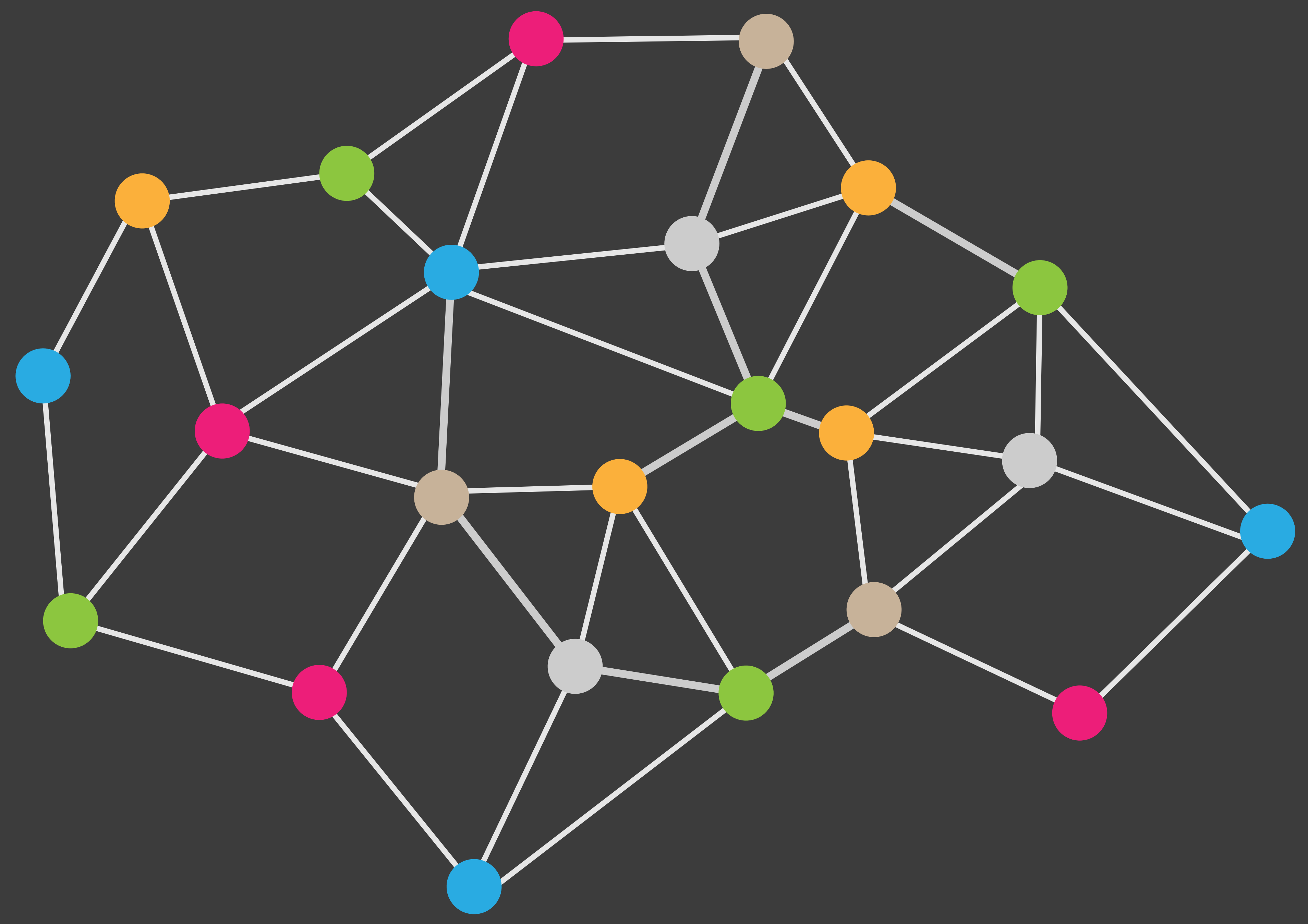
0x2b197453dcfabe85be2fbe31c8cc19bd30576ed0

DEPOSIT: 1000 DFN

0x2b197453dcfabe85be2fbe31c8cc19bd30576ed0

DEPOSIT: 1000 DFN

Are randomly assigned to groups that...



● GROUP

● GROUP

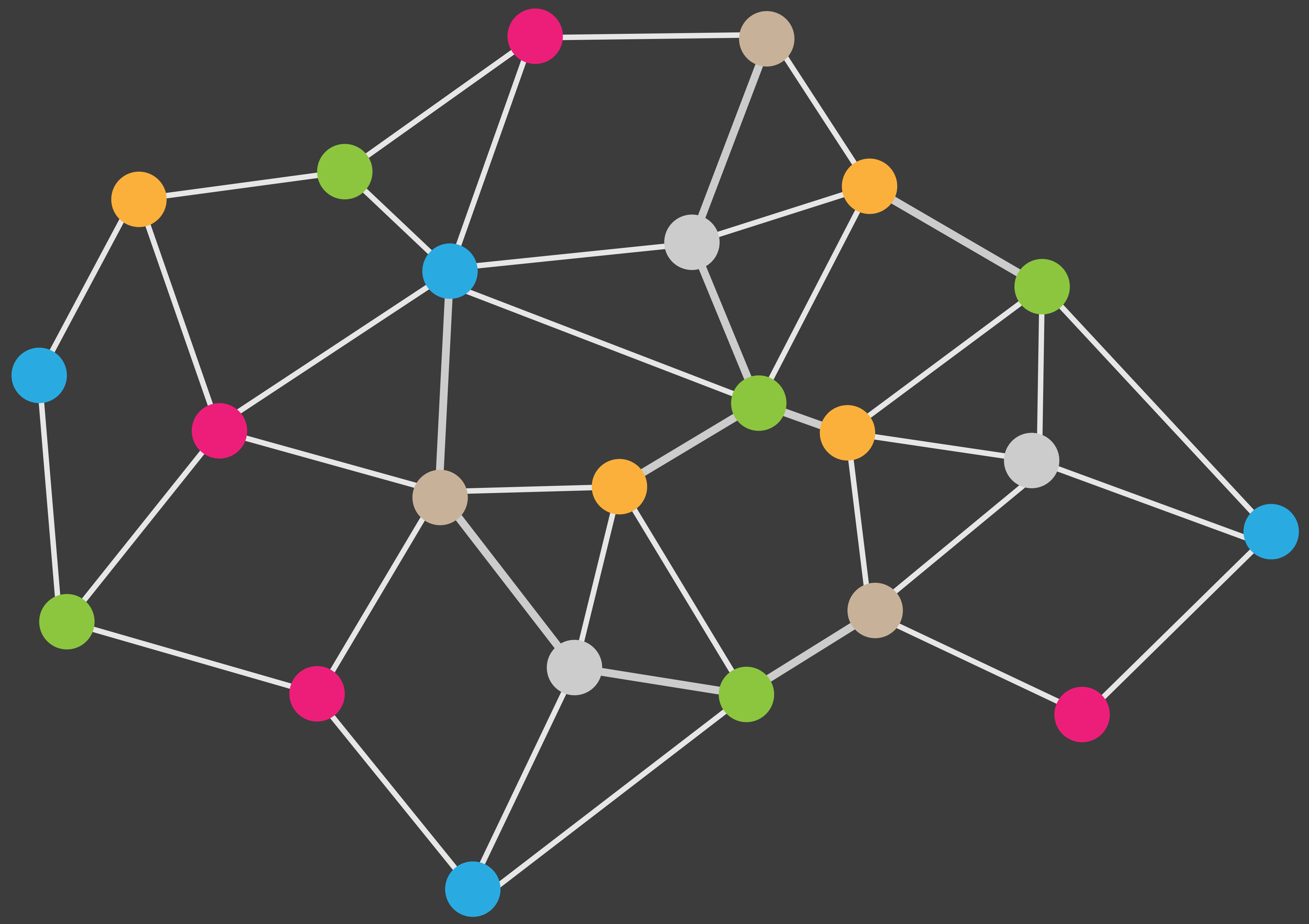
● GROUP

● GROUP

● GROUP

•••

Try to setup a “BLS threshold” scheme using DKG...



Joint Feldman DKG

● GROUP

● GROUP

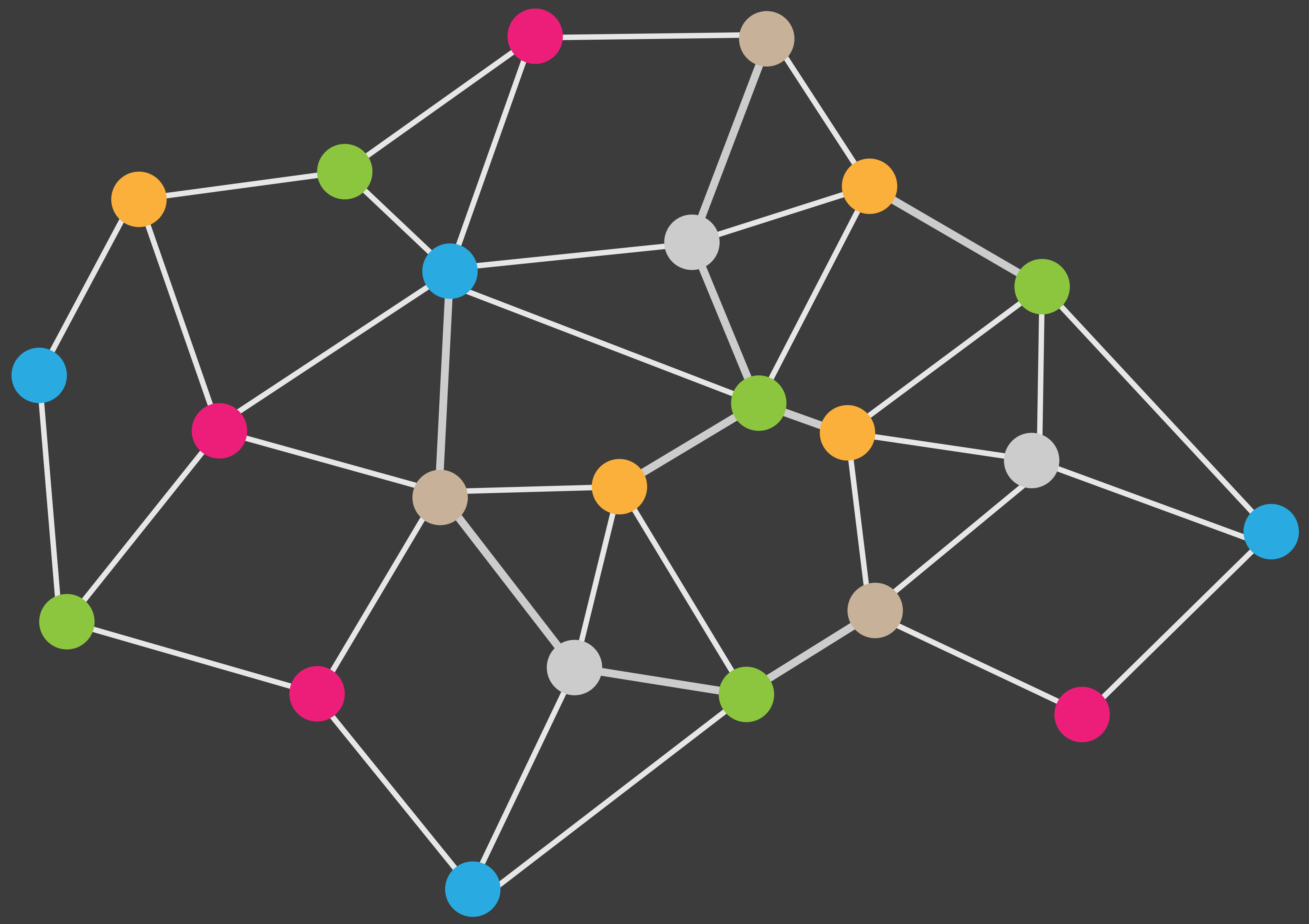
● GROUP

● GROUP

● GROUP

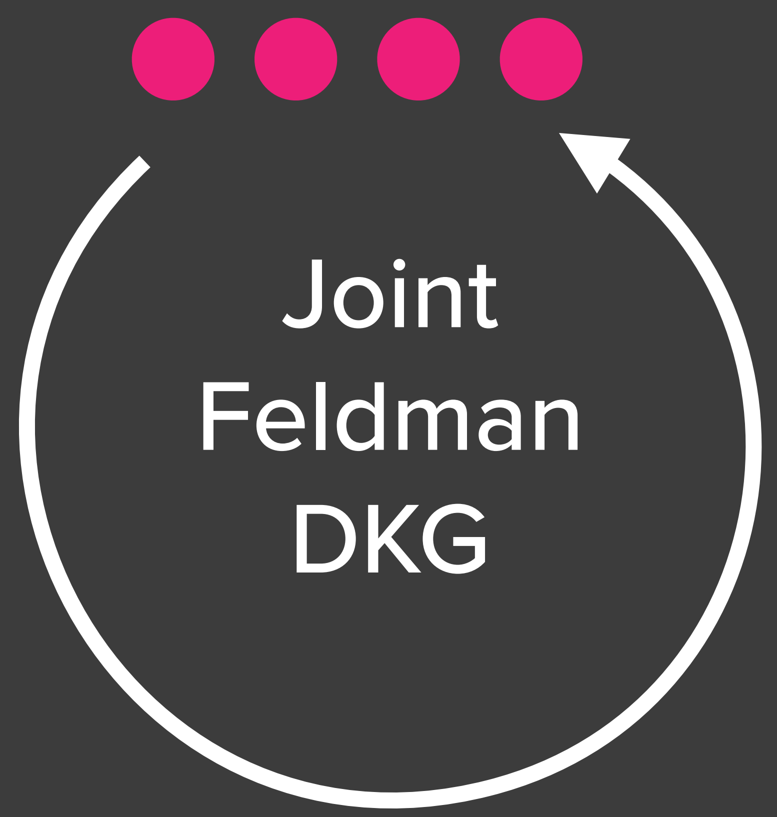
•••

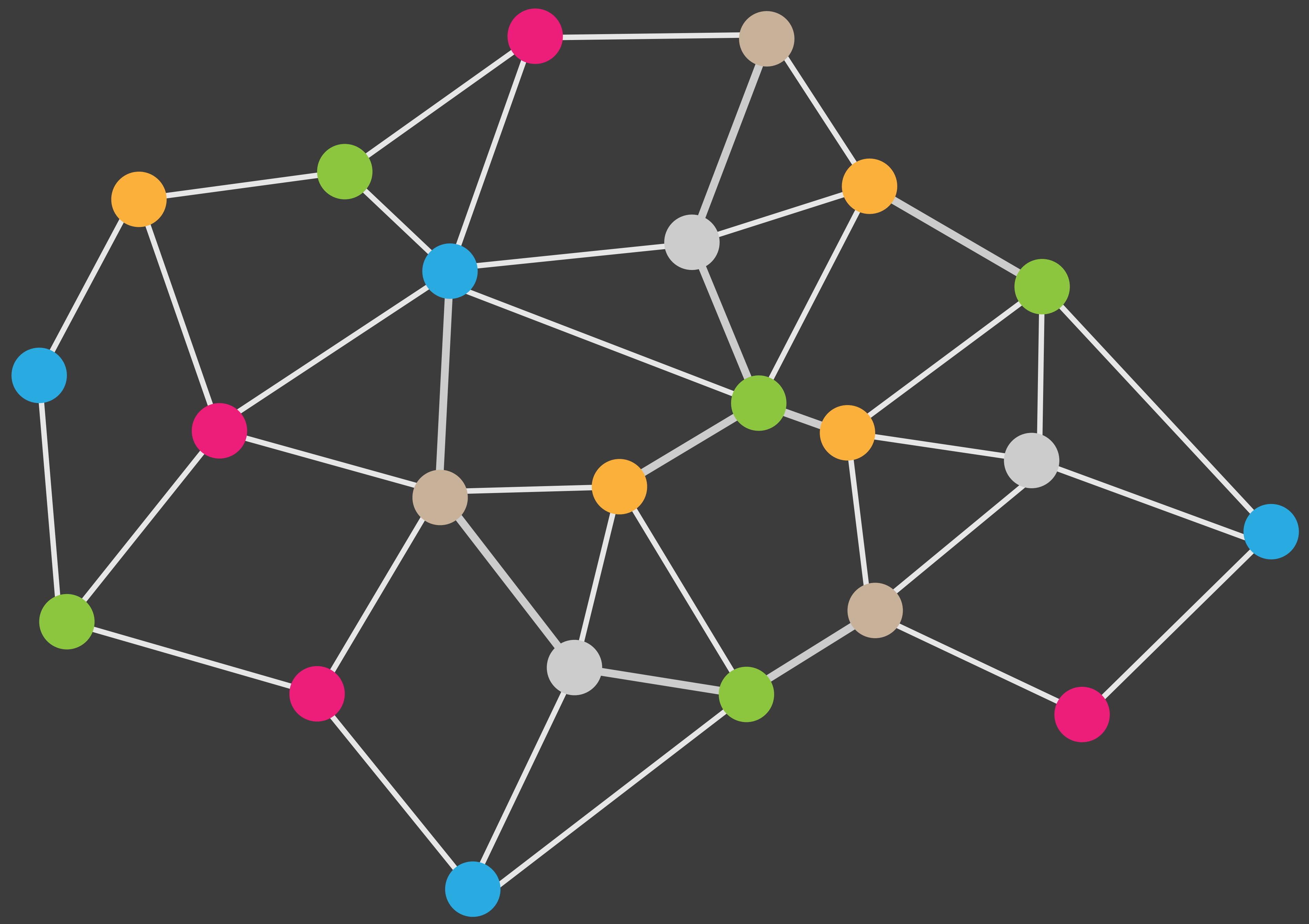
And register their PubKey on the ledger too

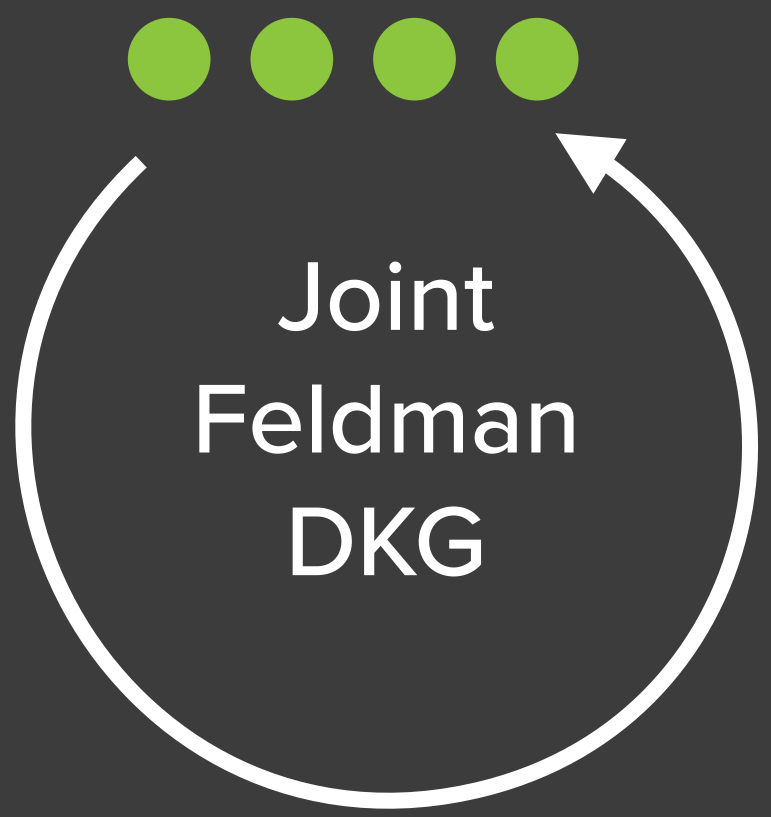


●	●	●	●	●	•••
GROUP	GROUP	GROUP	GROUP	GROUP	
—	—	—	0x2b197453…	—	

Setup is independent of blockchain progression...







● GROUP —

● GROUP —

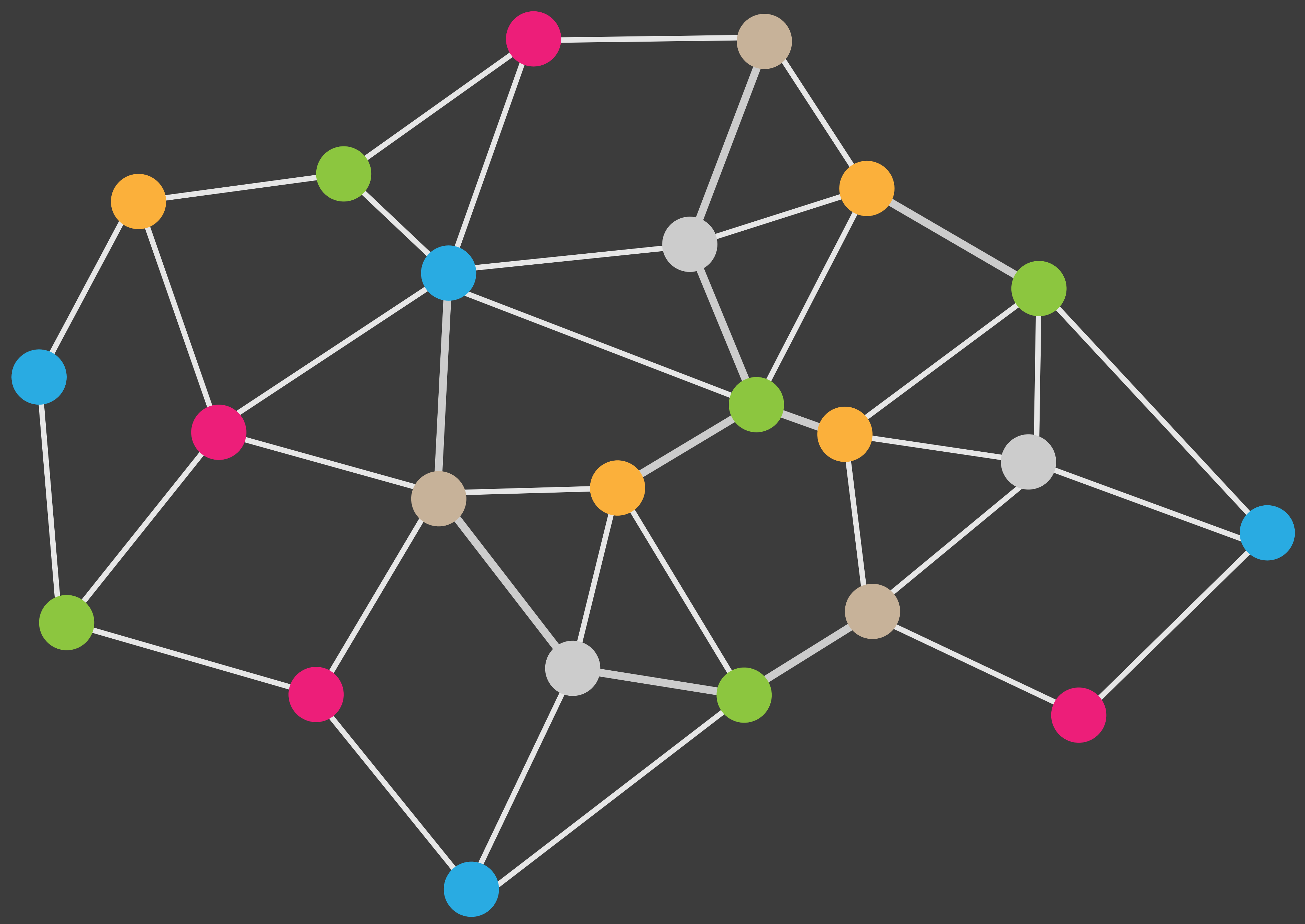
● GROUP —

● GROUP 0x2b197453…

● GROUP —

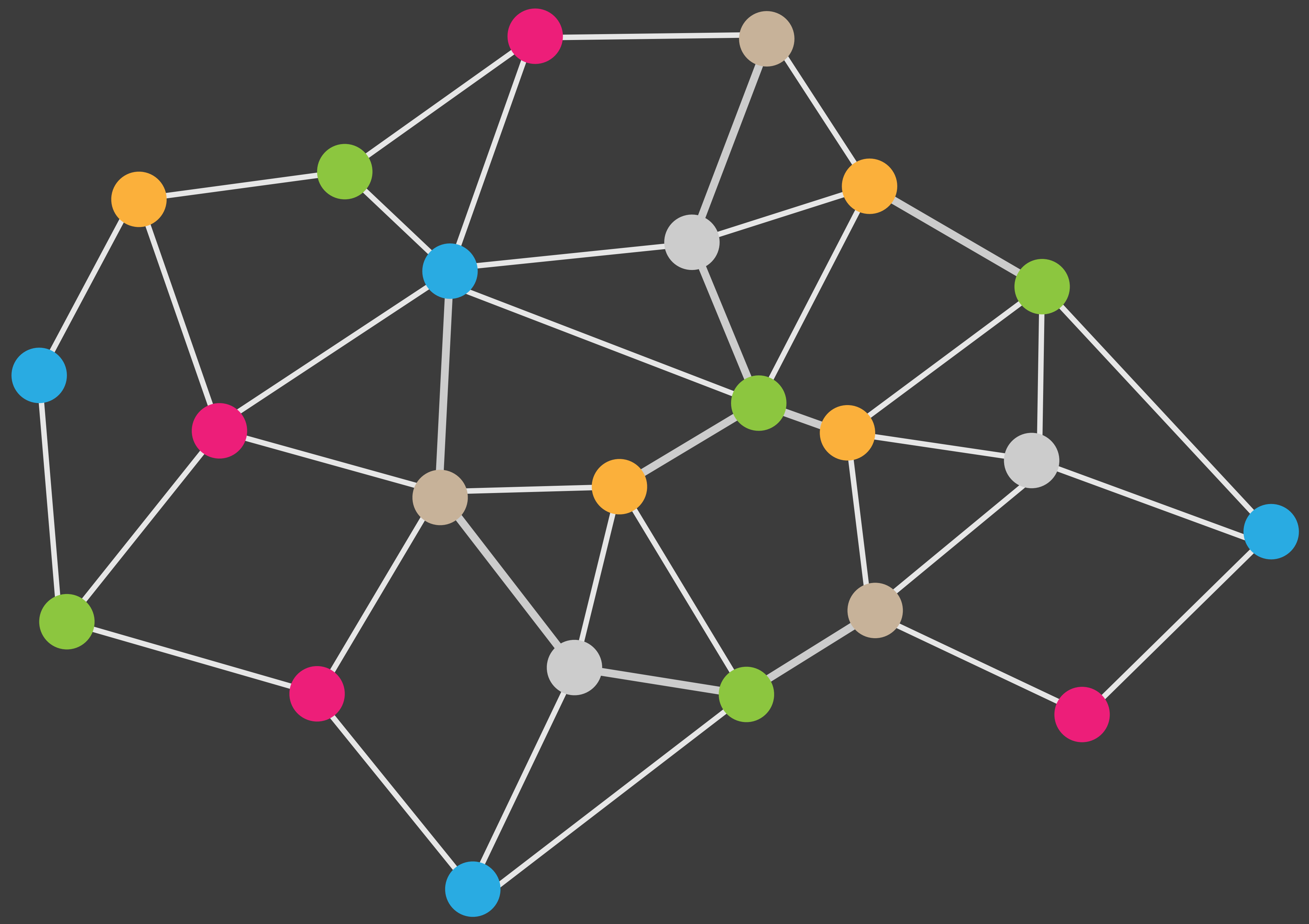
...

And occurs asynchronously

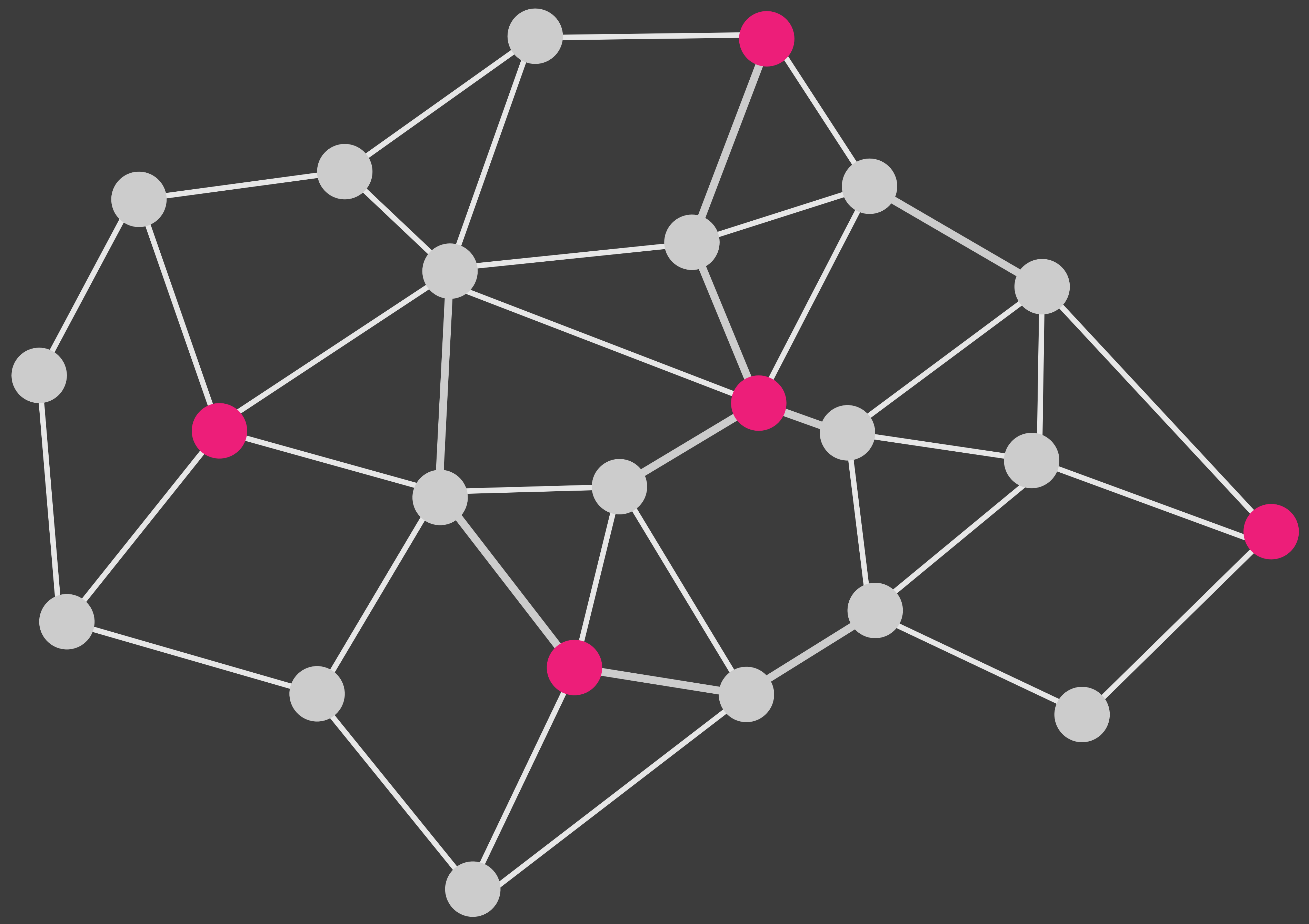


●	●	●	●	●	•••
GROUP	GROUP	GROUP	GROUP	GROUP	
0x7de4ac5…	0x8fb251b…	—	0x2b197453…	—	

As regards the blockchain itself...

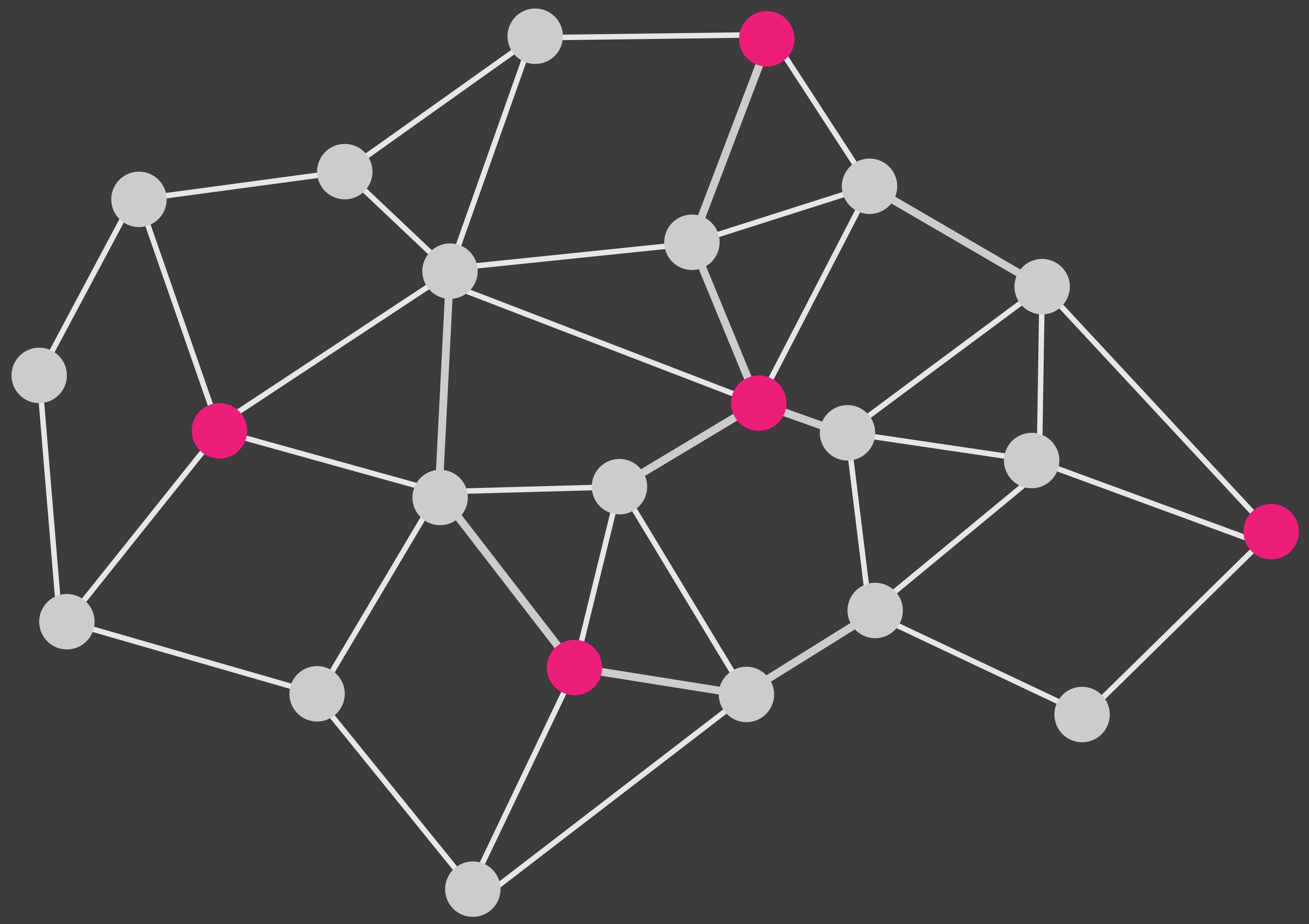


There is always a current group...



h

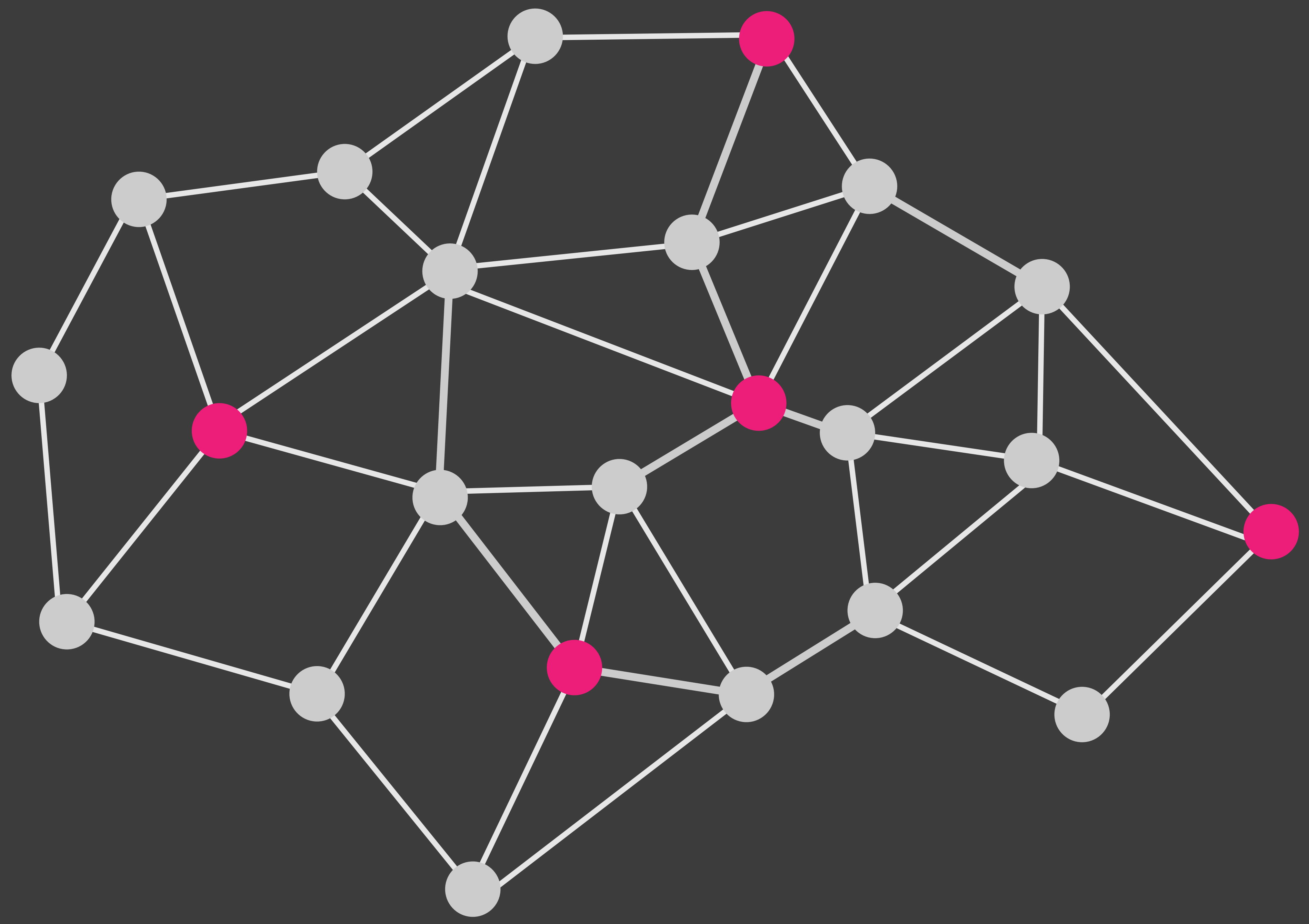
That signs the previous group’s signature...



$$e(\sigma, g) = e(H(m), g^{x})$$

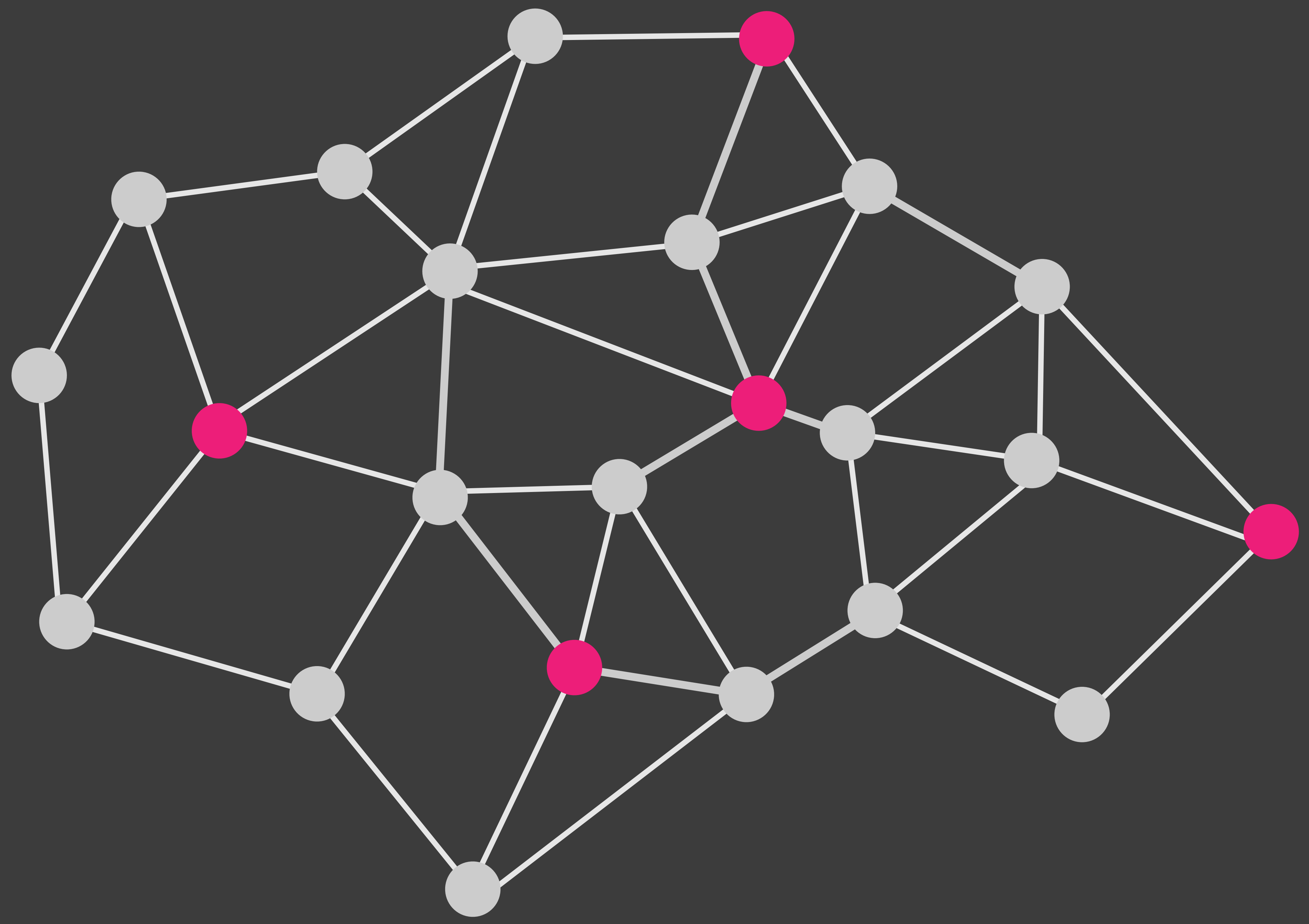
BLS Signature Scheme

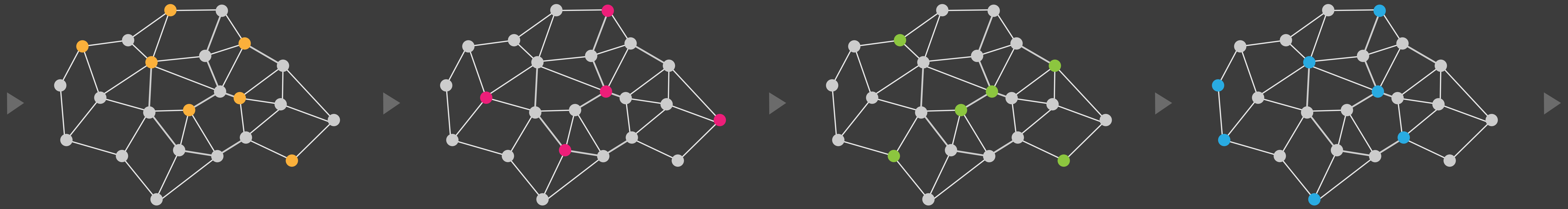
To select the next group and “relay”



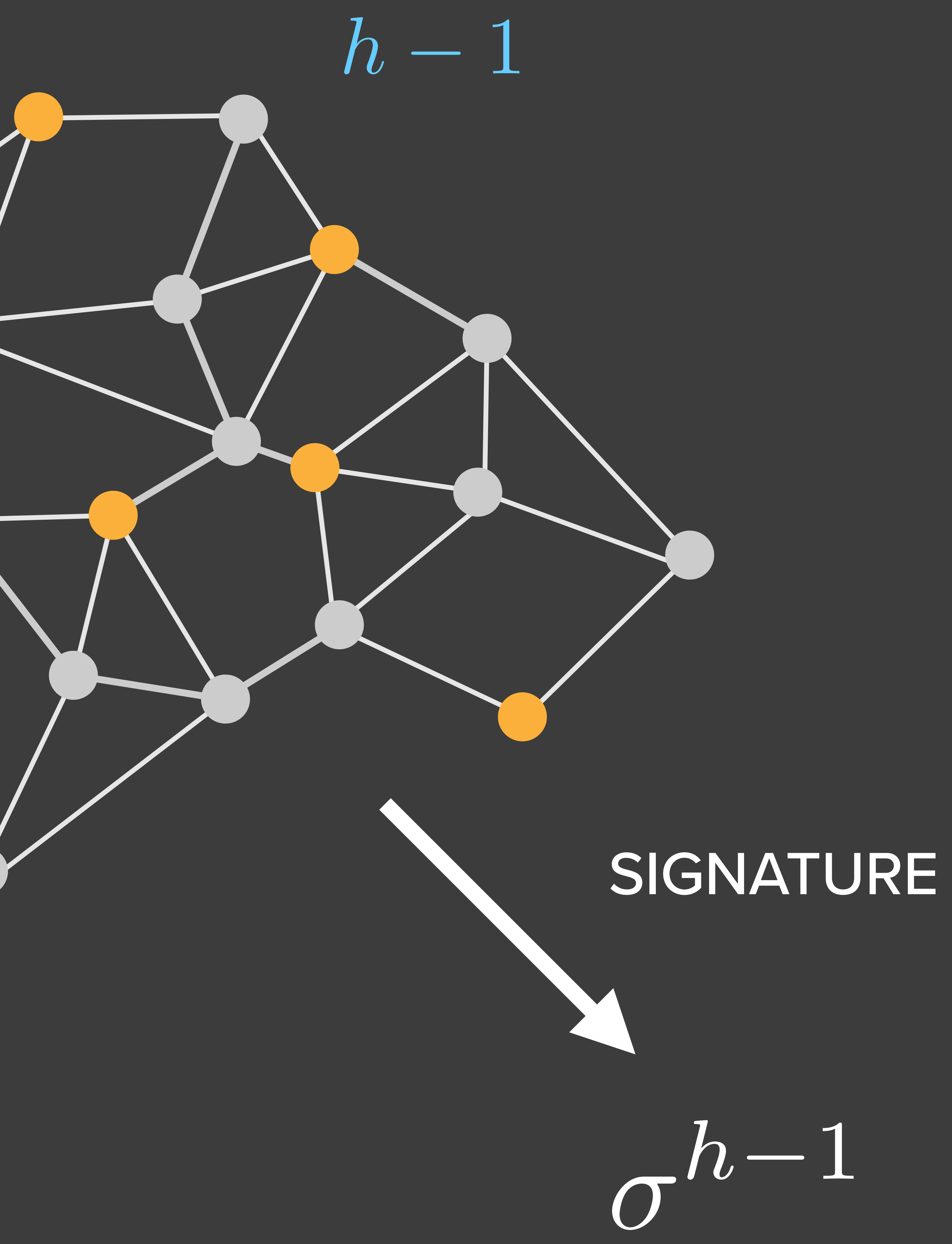
$$G^{h+1} = \mathcal{G}[\sigma^h \bmod |\mathcal{G}|]$$

To select the next group and “relay”





This is what Threshold Relay looks like

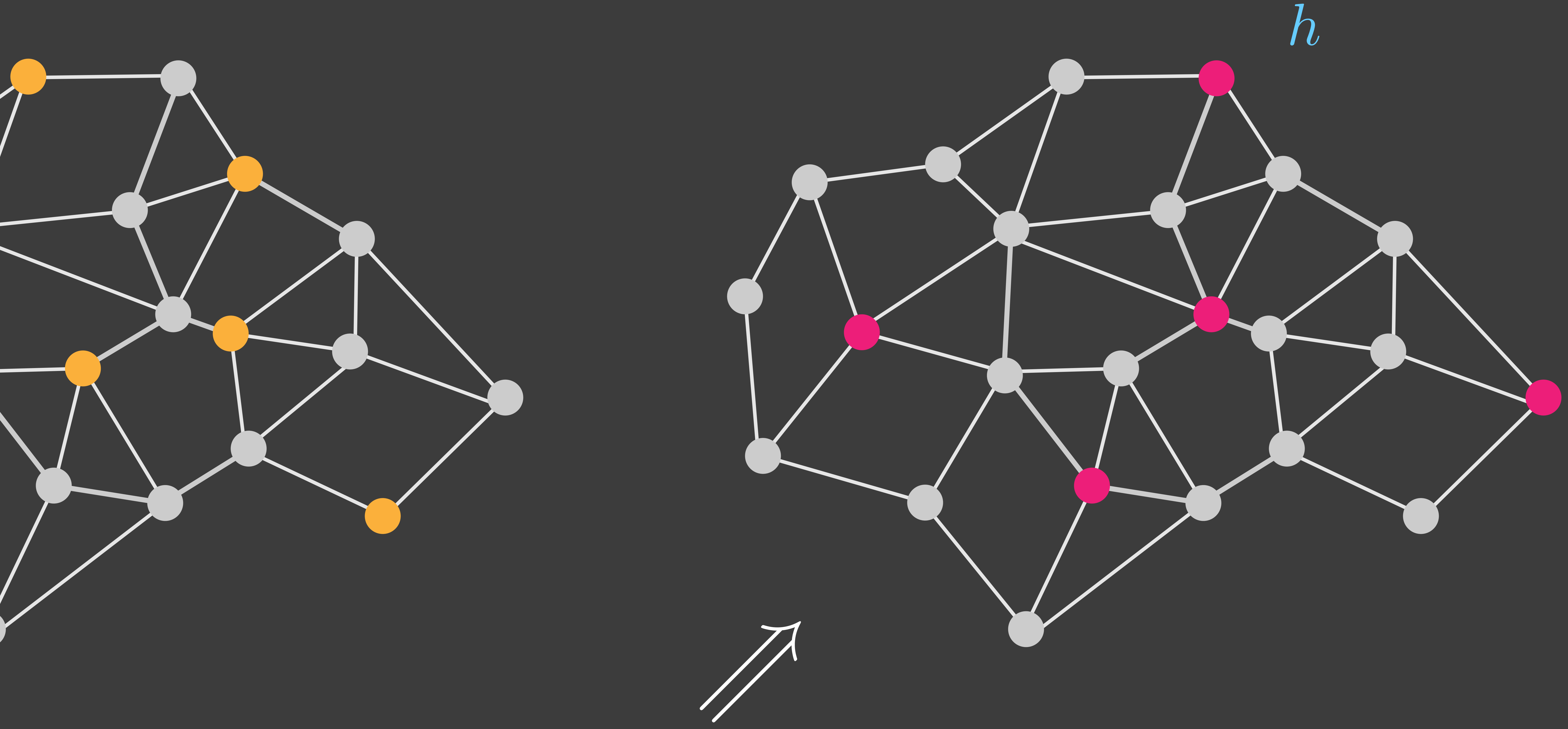


$h - 1$

SIGNATURE

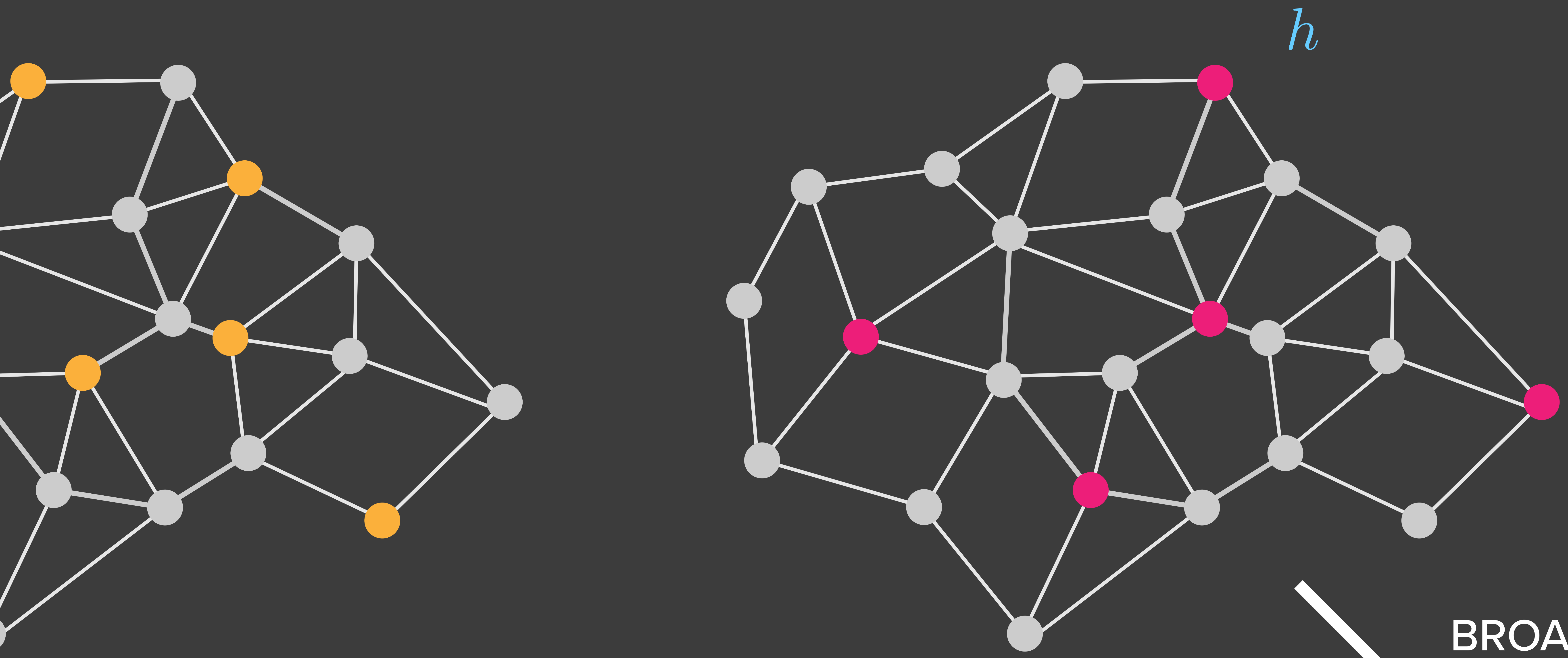
σ^{h-1}

The signature created at *h-1* selects the group at *h*



$$G^h = \mathcal{G}[\sigma^{h-1} \bmod |\mathcal{G}|]$$

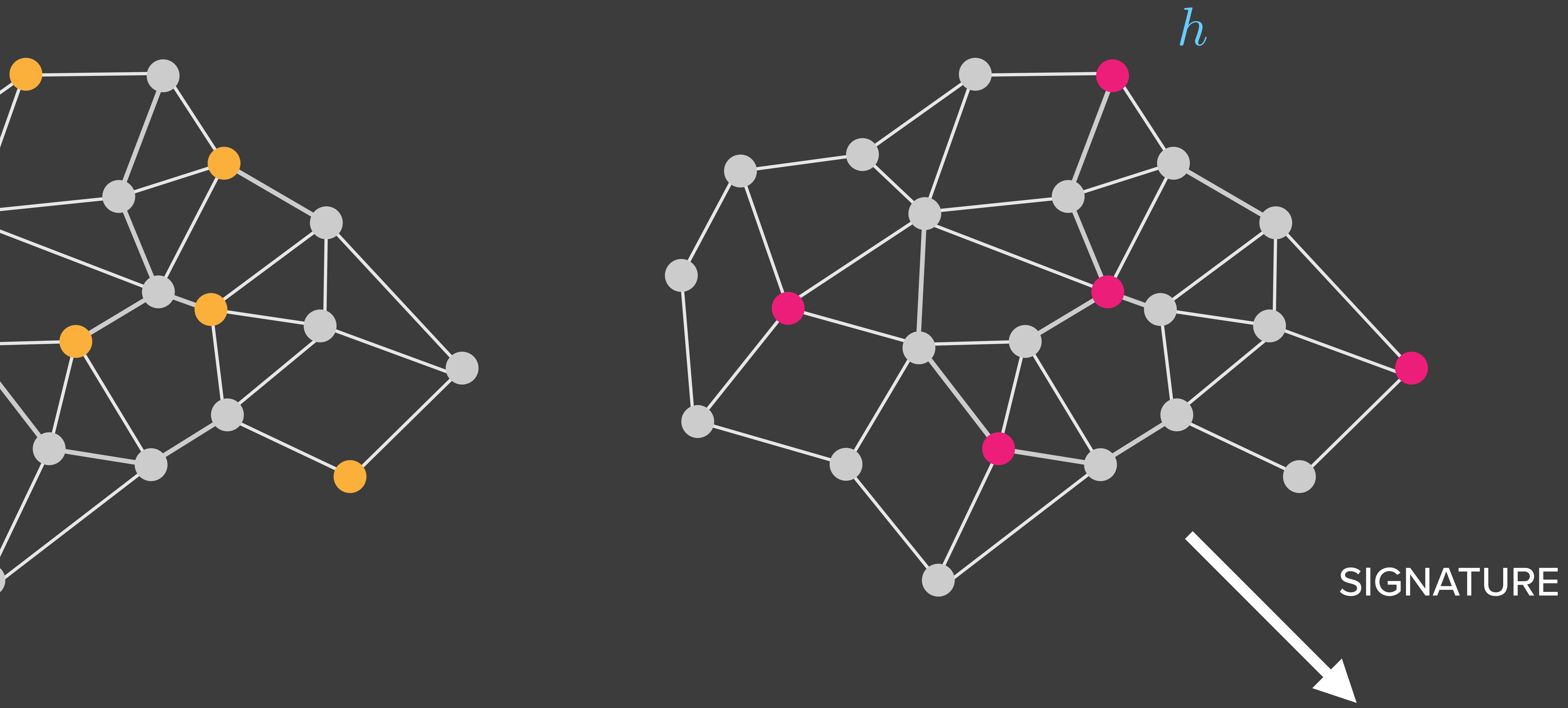
Group members at *h* broadcast signature shares



BROADCAST

$$\{\sigma_p^h, p \in G^h\}$$

Collect threshold of shares & create only possible group sig…



$$\sigma^h = bls(\{\sigma_p^h, p \in G^h\})$$

That selects the next group, ad infinitum

$$G^{h+1} = \mathcal{G}[\sigma^h \bmod |\mathcal{G}|]$$

This creates a decentralized VRF

$$\ldots h-7, \quad \sigma^{h-6}, \quad \sigma^{h-5}, \quad \sigma^{h-4}, \quad \sigma^{h-3}, \quad \sigma^{h-2}, \quad \sigma^{h-1}, \quad \sigma^{h} \longrightarrow$$

A sequence of random numbers that is...

Deterministic • Verifiable • Unmanipulable

Next value released on agreement a threshold of the current group...

Unpredictable

> “Random numbers should not be generated with a method chosen at random
>
> \- Donald Knuth

TLDR; unmanipulable randomness is v useful…

Scale-out Decentralized Network Protocols

DFINITY

PSP Blockchain Designs

Validation Towers

Validation Trees

USCIDs

Lottery Charging

Lazy Validation

Advanced Decentralized "Applications"

ϕ

Autonomous loan issuance and crypto "fiat"

Financial exchanges

Data harvesting

Fault Tolerance Example

NETWORK METRICS

Processes	10,000
Faulty	3,000
(Correct)	7,000
Group Size	400
Threshold	201

Note: in practice the probability 30% of professionally run mining processes “just stop” is very low. Miners will generally deregister IDs to retrieve deposits when exiting.

$$P(Faulty \geq 200)$$

$$1e-17$$

Probability that a sufficient proportion of the group are faulty that it cannot produce a signature

Calculated using hypergeometric probability.
http://www.geneprof.org/GeneProf/tools/hypergeometric.jsp

Note: groups should expire to thwart “adaptive” adversaries

Communications Overhead Example

MESSAGE FORMAT

Process ID	20 bytes
Signature share	32 bytes
Signature on comms	32 bytes
Total	84 bytes

In order for a group to produce a threshold signature, its members must broadcast "signature shares" on the message that can be combined. Here is a typical packet carrying a signature share.

GROUP SIZE

Group size	400
Threshold	201

COMMUNICATION OVERHEAD

Maximum	34 KB

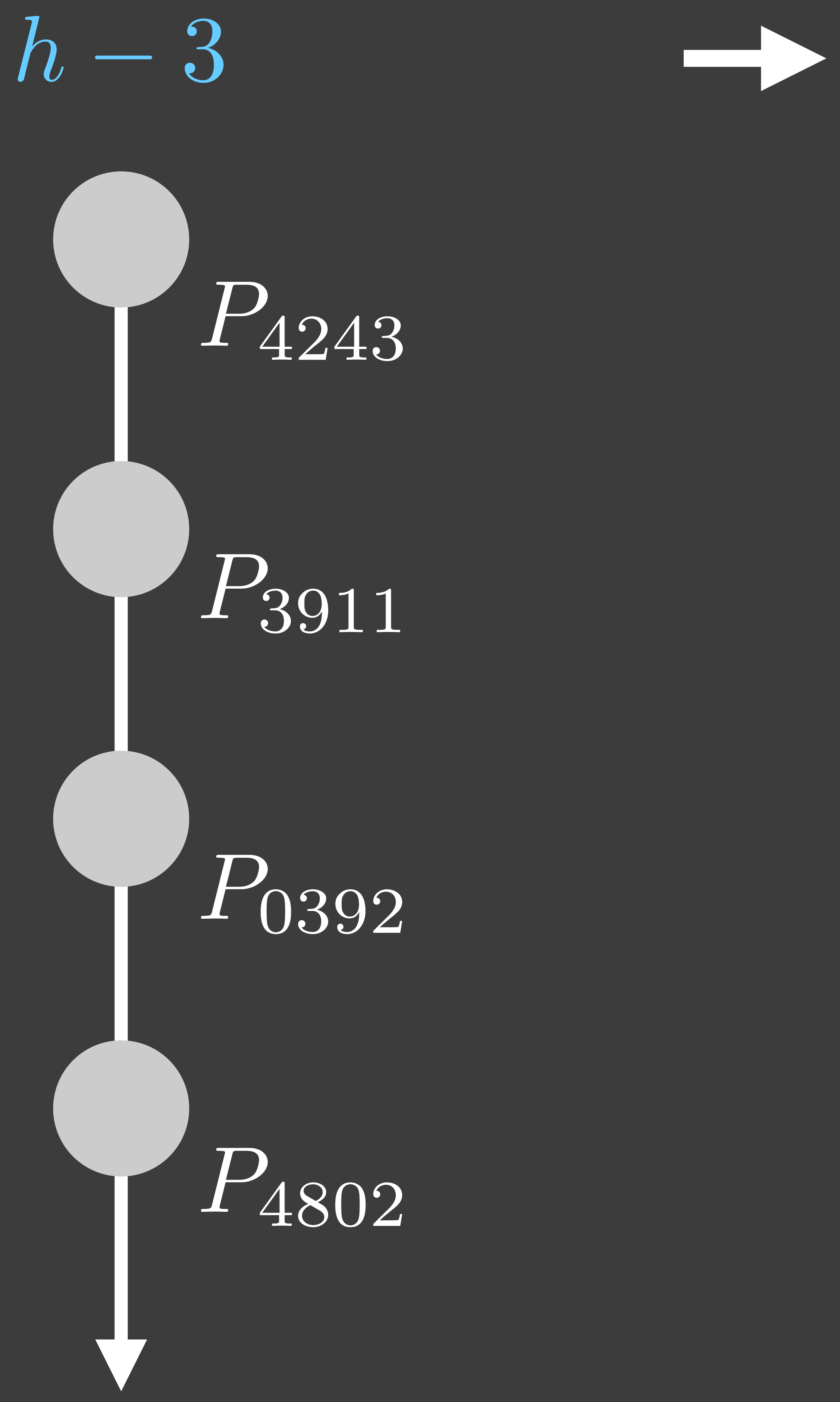
400 messages involve 34 KB of data transfer. However, only 17 KB (half the messages) are required to construct the signature. Thereafter signature shares are not relayed, so a more typical overhead is 22 KB.

2

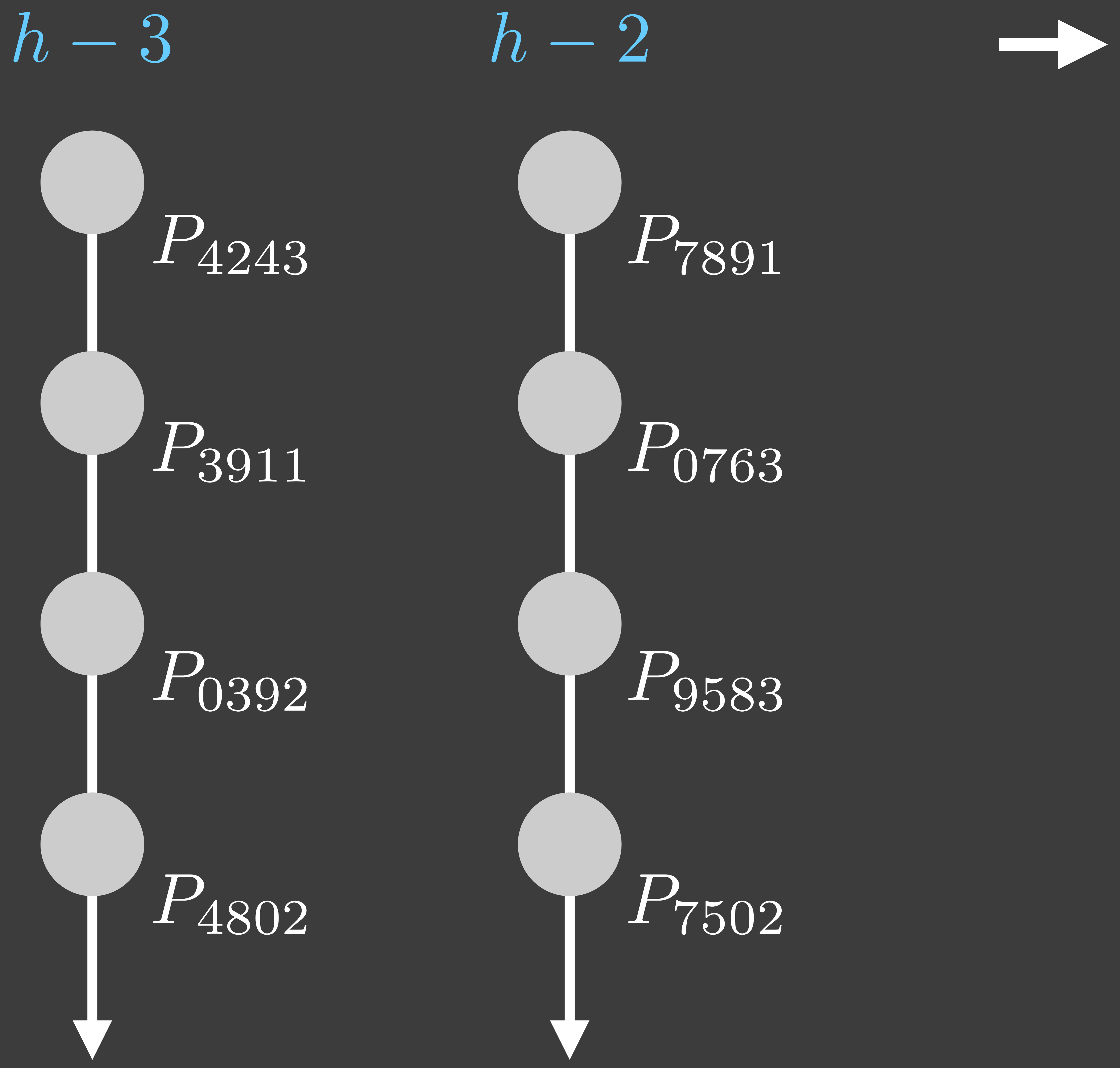
Threshold Relay Blockchain

A Simple “Probabilistic Slot Protocol” (PSP)

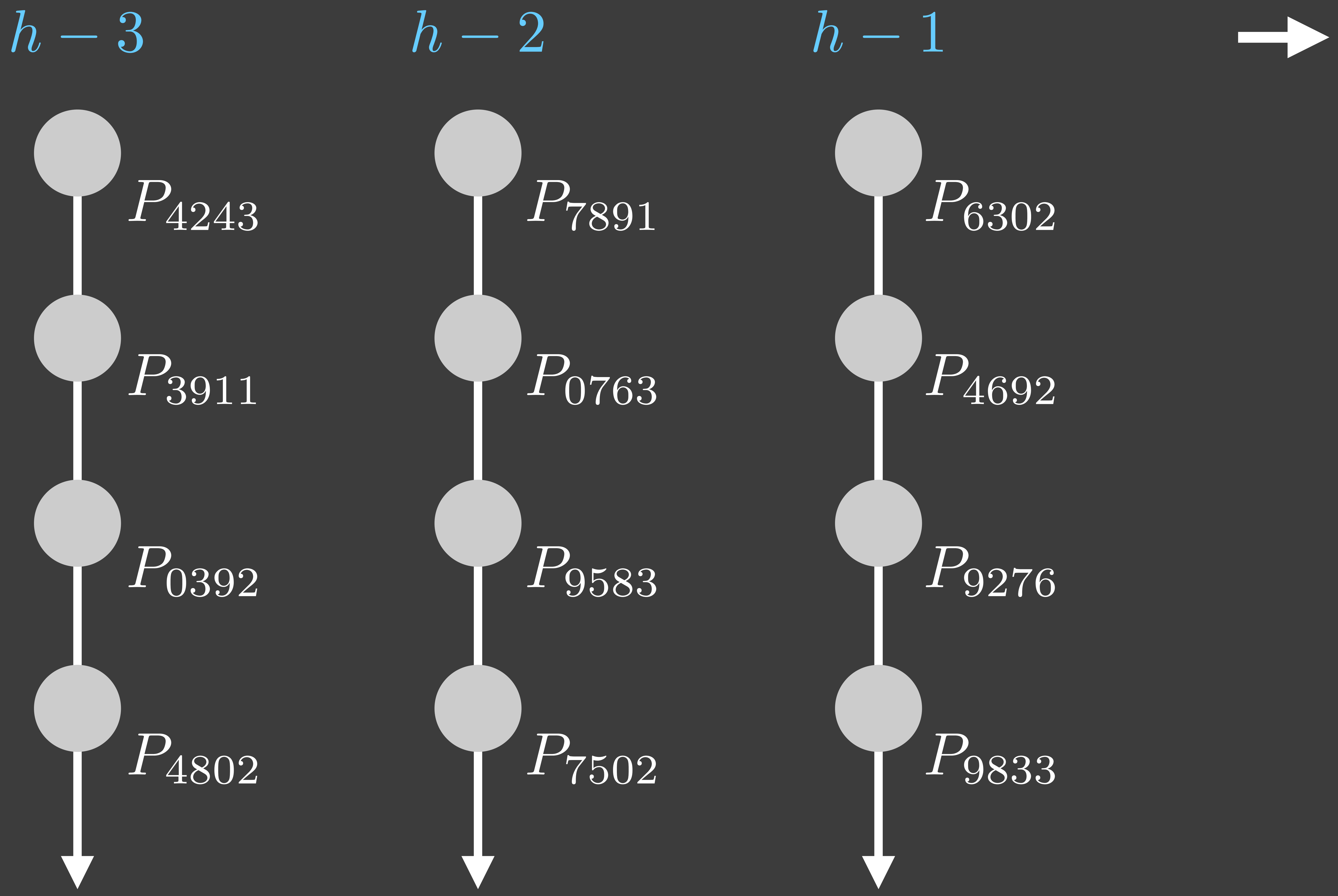
At each height, the randomness orders the processes...



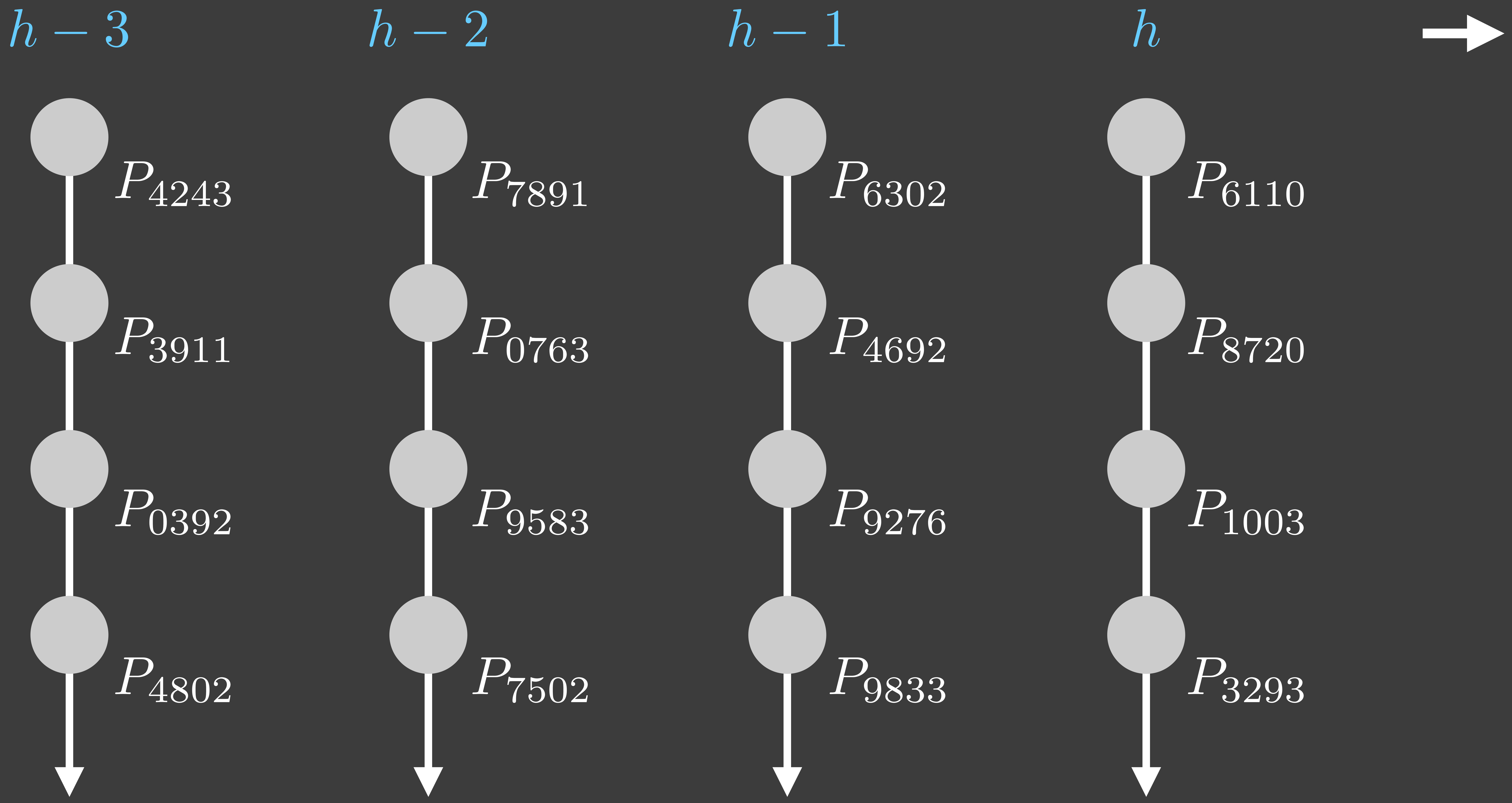
At each height, the randomness orders the processes...



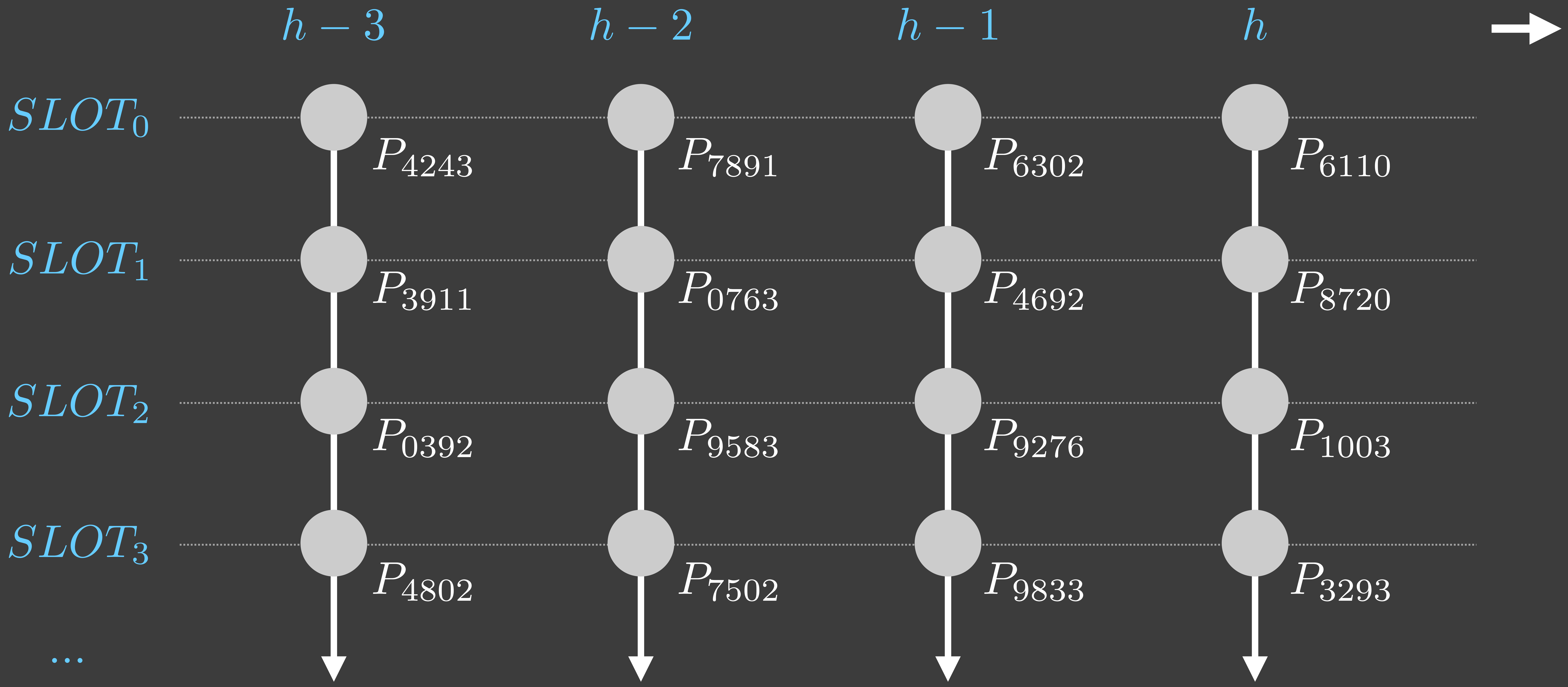
At each height, the randomness orders the processes...



At each height, the randomness orders the processes...

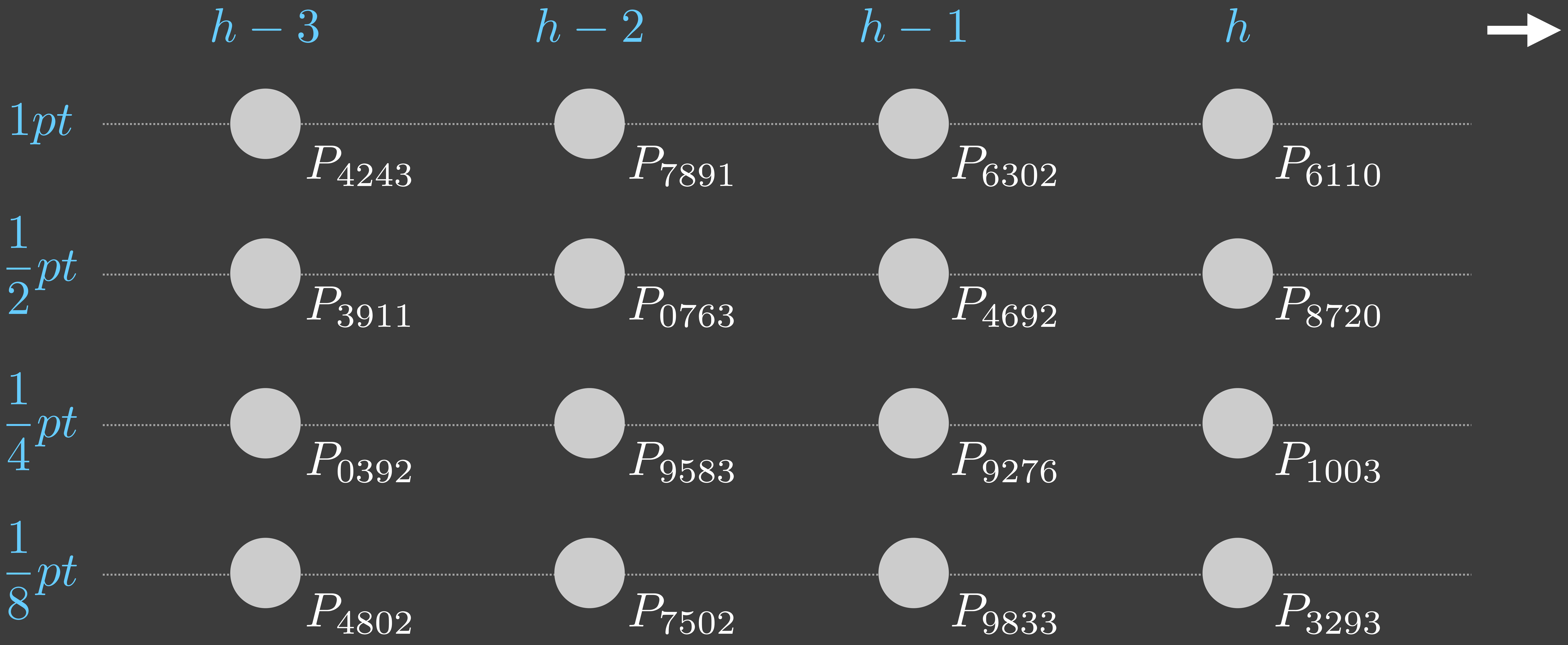


Indexes are priority “slots” for forging (zero highest)



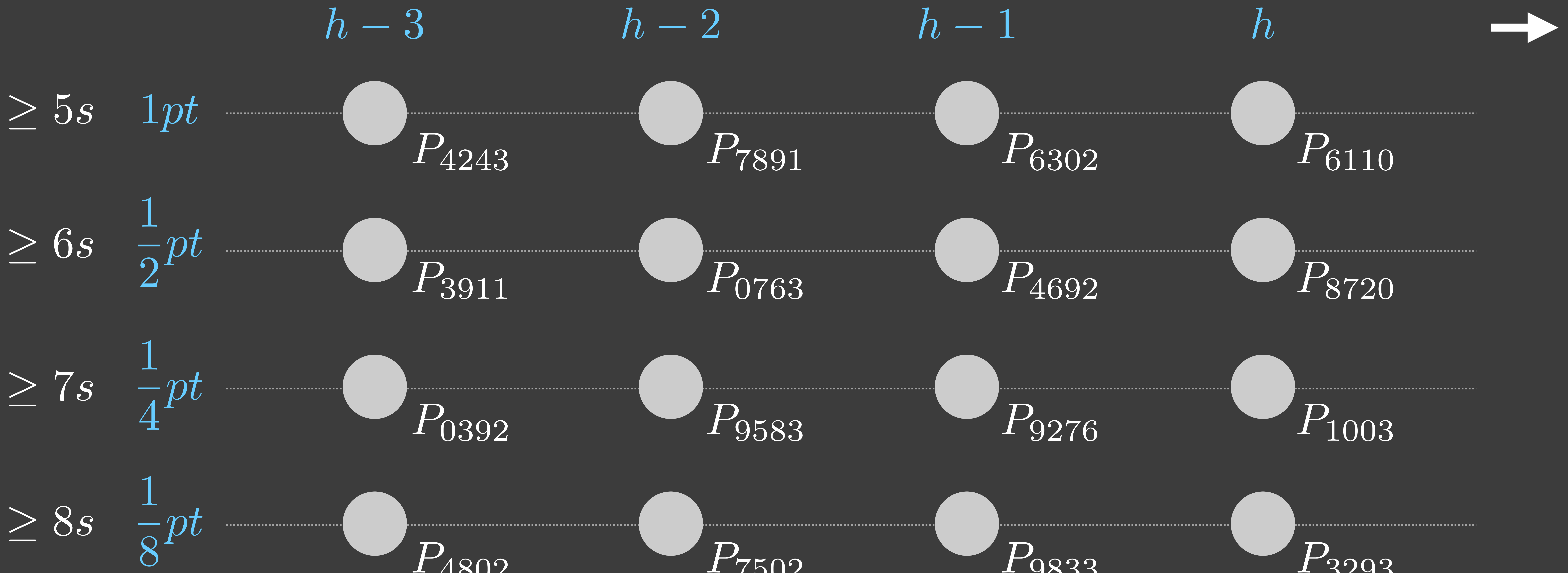
	$h-3$	$h-2$	$h-1$	h
$SLOT_0$	P_{4243}	P_{7891}	P_{6302}	P_{6110}
$SLOT_1$	P_{3911}	P_{0763}	P_{4692}	P_{8720}
$SLOT_2$	P_{0392}	P_{9583}	P_{9276}	P_{1003}
$SLOT_3$	P_{4802}	P_{7502}	P_{9833}	P_{3293}
...				

Value of candidate blocks scored by author's slot...



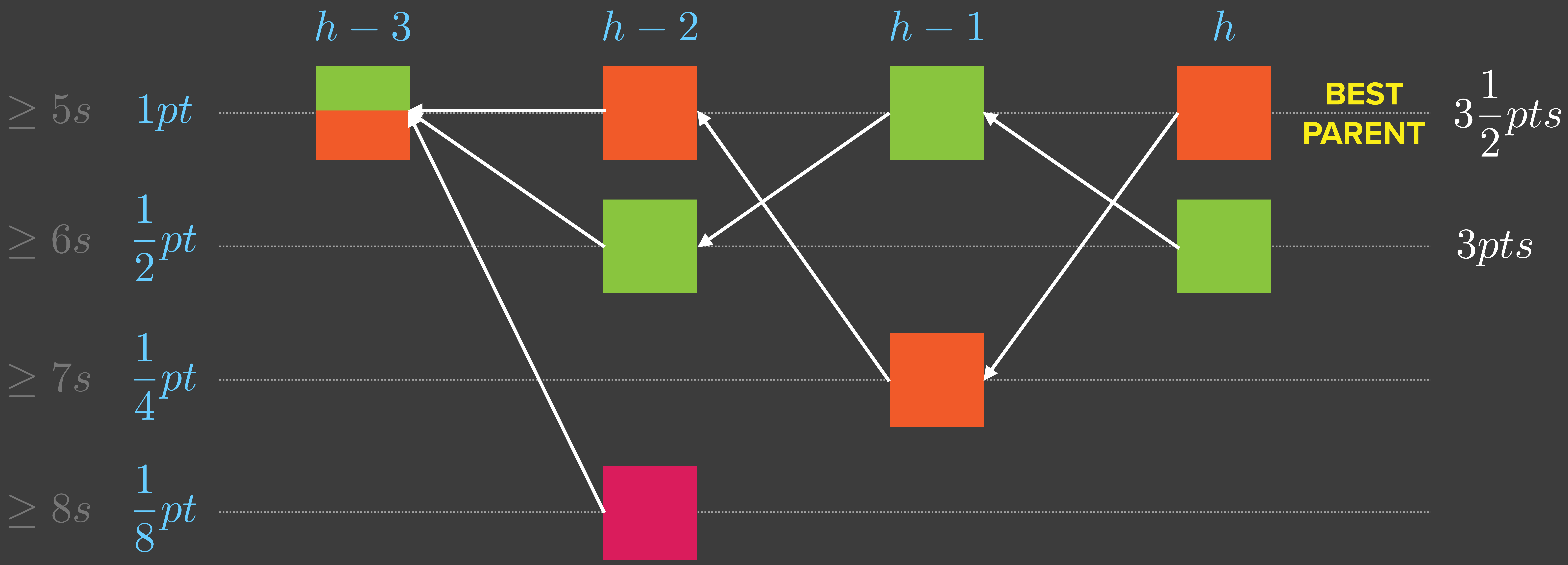
	$h-3$	$h-2$	$h-1$	h
$1pt$	P_{4243}	P_{7891}	P_{6302}	P_{6110}
$\frac{1}{2}pt$	P_{3911}	P_{0763}	P_{4692}	P_{8720}
$\frac{1}{4}pt$	P_{0392}	P_{9583}	P_{9276}	P_{1003}
$\frac{1}{8}pt$	P_{4802}	P_{7502}	P_{9833}	P_{3293}

First publish/relay delay too (an optimization)...



		$h-3$	$h-2$	$h-1$	h
$\geq 5s$	$1pt$	P_{4243}	P_{7891}	P_{6302}	P_{6110}
$\geq 6s$	$\frac{1}{2}pt$	P_{3911}	P_{0763}	P_{4692}	P_{8720}
$\geq 7s$	$\frac{1}{4}pt$	P_{0392}	P_{9583}	P_{9276}	P_{1003}
$\geq 8s$	$\frac{1}{8}pt$	P_{4802}	P_{7502}	P_{9833}	P_{3293}

We can create & score blockchains that converge



$h-3$
$h-2$
$h-1$
h
$\geq 5s$ $1pt$
$\geq 6s$ $\frac{1}{2}pt$
$\geq 7s$ $\frac{1}{4}pt$
$\geq 8s$ $\frac{1}{8}pt$
BEST PARENT
$3\frac{1}{2}pts$
$3pts$

Very nice. But usual limitations. O no...

SELFISH MINING ATTACKS

The adversary can withhold blocks to gain an advantage over honest processes.

Selfish mining attacks increase the confirmations necessary for finality.

NOTHING AT STAKE

The adversary can go back in time and create forks from below *h* to Double Spend.

He only needs to be lucky and be granted a sequence of zero slots.

Solution?

Threshold groups “**notarize**” (sign) at least one block at their height before relaying…

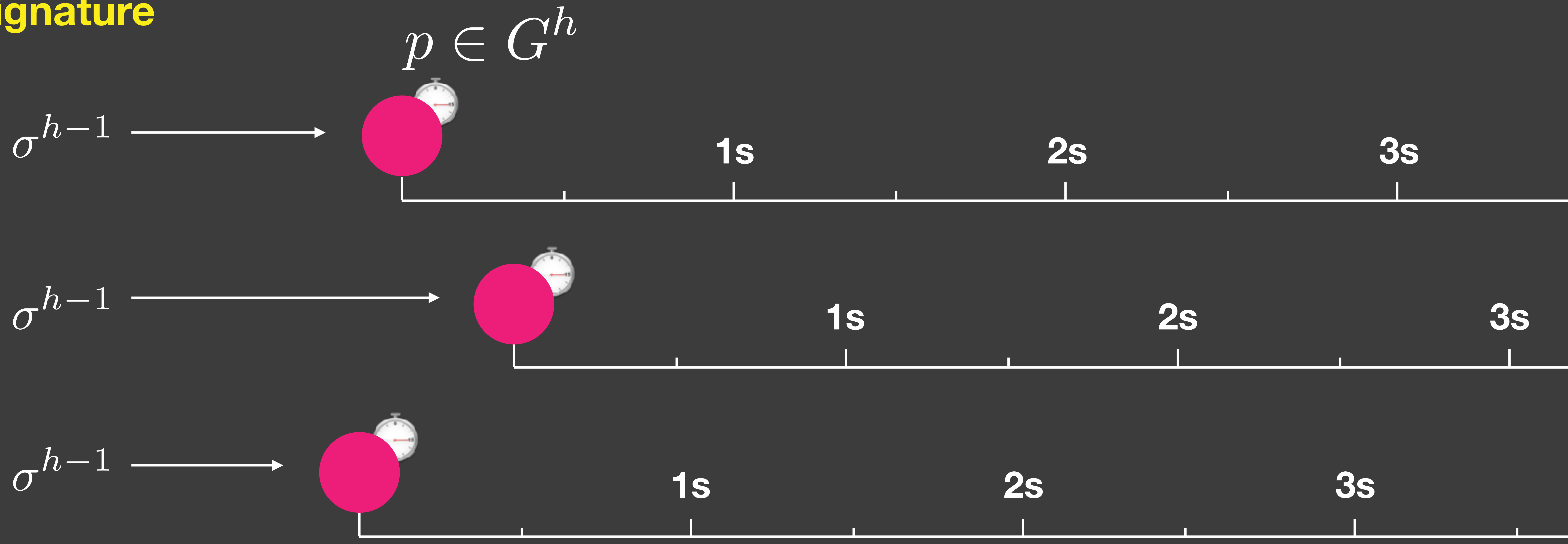
A valid block proposed at ***h*** must reference a block that was notarized at ***h-1***

Thus, blocks must be published in good time or have no chance of notarization

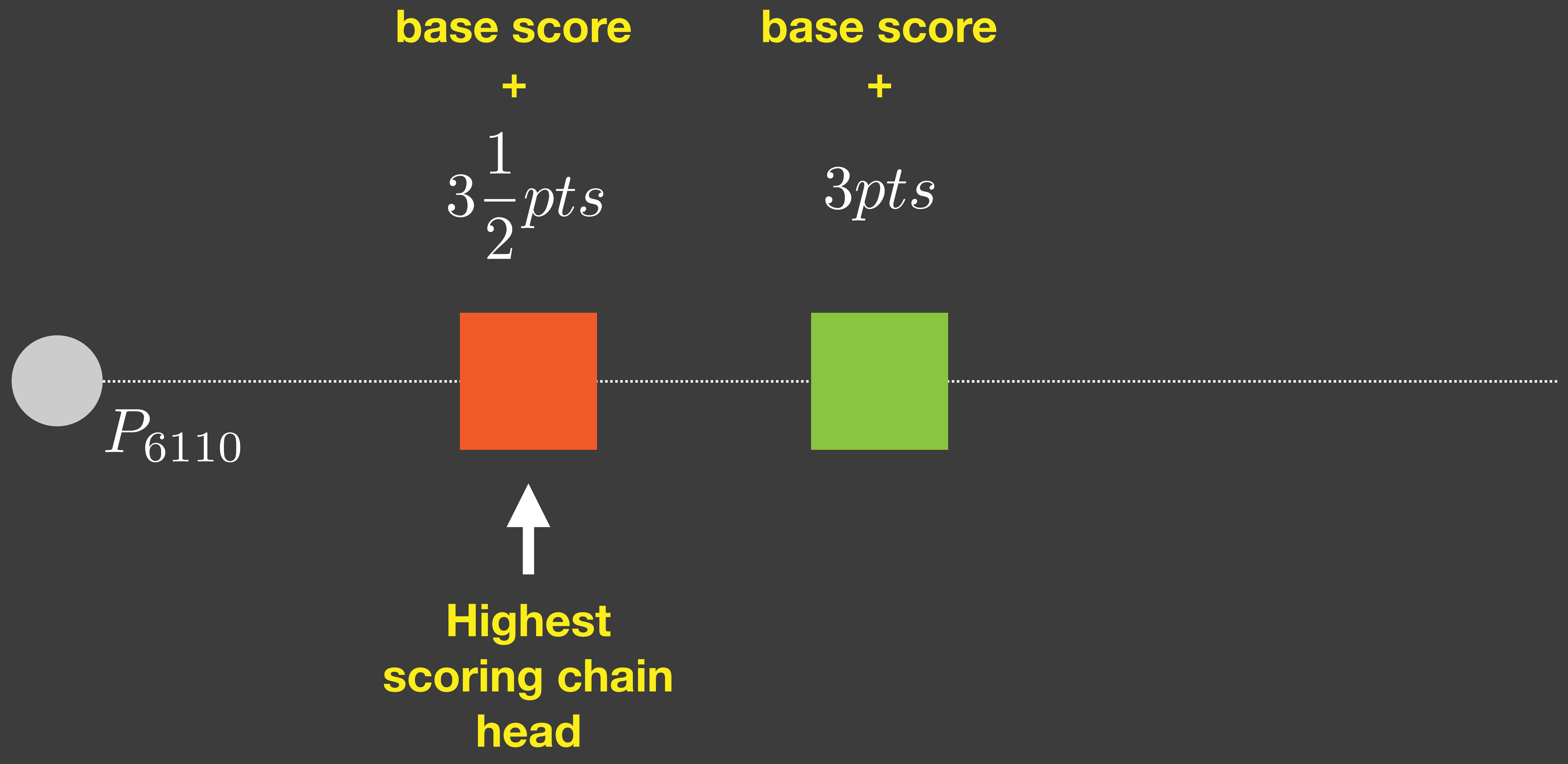
When group selected, its members start their timers...

Triggered by propagation threshold signature

Members start processing blocks after expiry BLOCK_TIME. Clocks will be slightly out-of-sync, but that's OK!

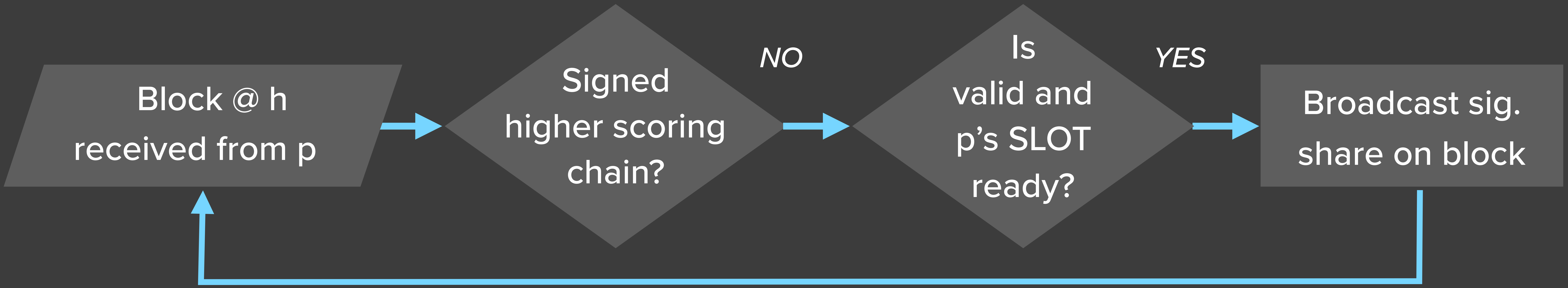


Queue blocks score order while waiting BLOCK_TIME



When BLOCK_TIME expires, start notarizing...

Group members sign until ≥1 blocks receive threshold signature



Block @ h received from p

Signed higher scoring chain?

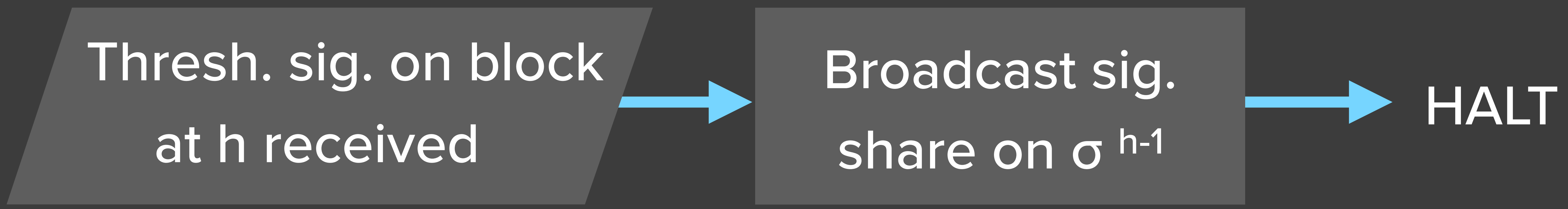
NO

Is valid and p’s SLOT ready?

YES

Broadcast sig. share on block

Sign best or equal best blocks



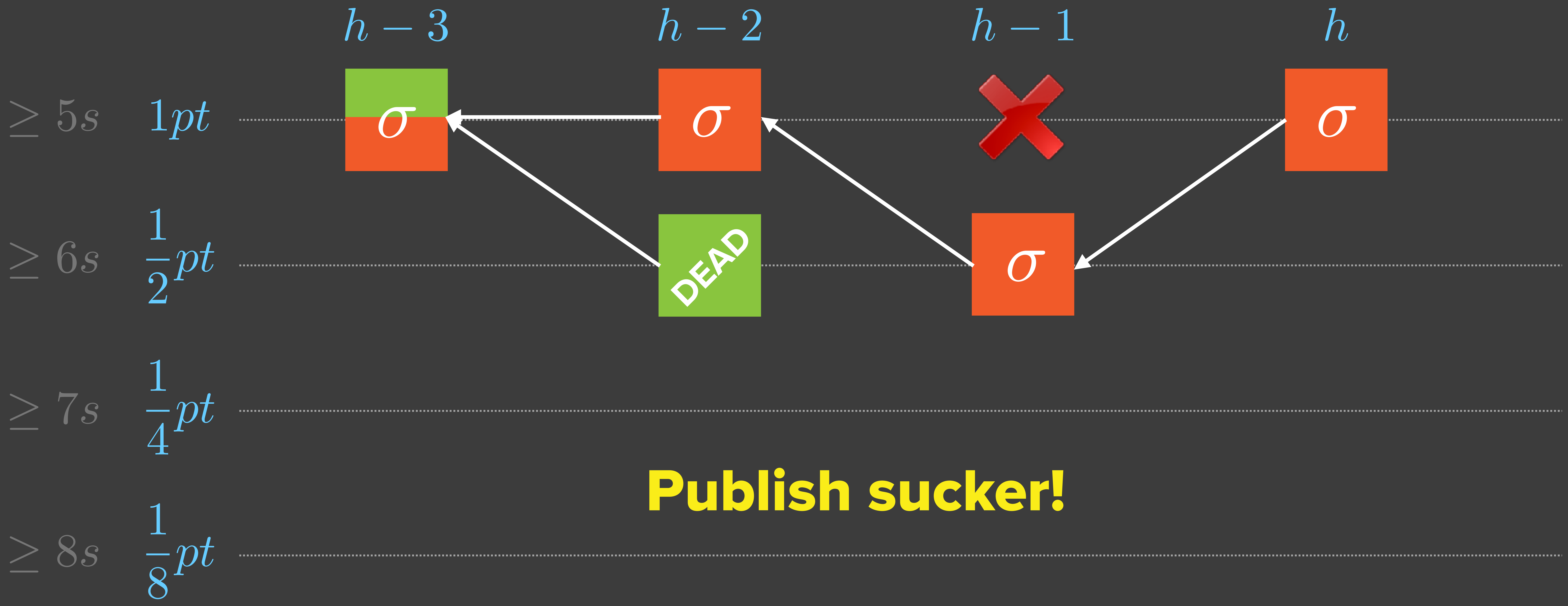
Thresh. sig. on block at h received

Broadcast sig. share on σ^{h-1}

HALT

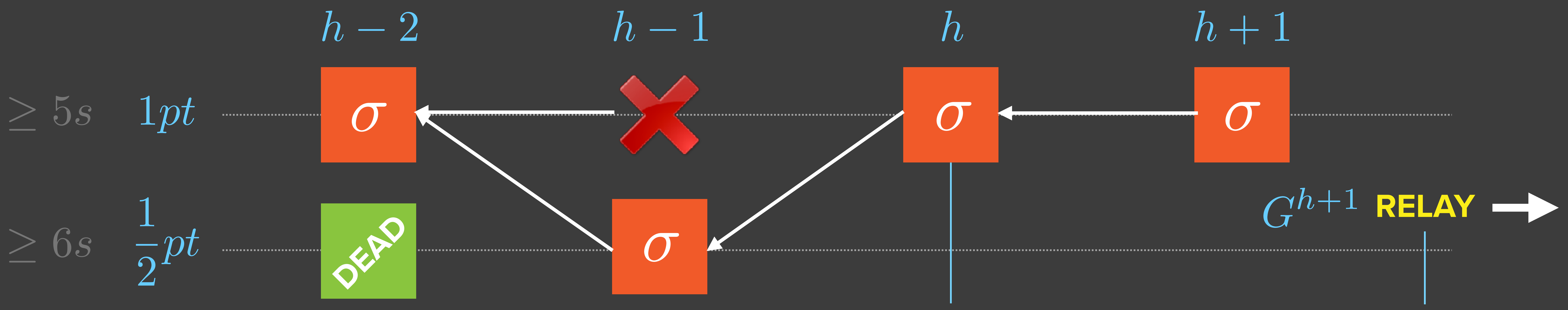
Relay and halt

Fair mining and very fast convergence



Publish sucker!

Optimal case. Overwhelming finality in 2 blocks + relay



No alternative chain head

or even partially signed chain head is visible. Yet, for a viable chain head to exist, it must have been shared with some correct processes to collect signatures, and they would have propagated (broadcast) it…

The trap shuts!

Now group ***h+1*** has relayed it will not notarize/sign any more blocks. Too late for any alternative chain head at ***h*** to “appear” and get notarized…

Gains from Notarization

Fast Optimal Avg. Finality

$$BLOCK\_TIME = 5s$$

$$\Longrightarrow$$

$$7.5s$$

Addresses Key Challenges

- Selfish Mining
- Nothing At Stake
- Equivocation

Quantifiable risk

Hooks make possible
calculate probabilities more
meaningfully

SPV

Light client needs only
Merkle root of groups

Relative Performance Copper Release

	₿	ethereum	DFINITY
Block Time	Average 10 mins *varies wildly*	Average 20 secs *varies wildly*	Average 5 secs *low variance*
“TX finality” (speed)	6 confirmations avg. 1 hr	37 confirmations avg. 10 mins	2 confirmations+relay ***avg. 7.5 secs*** ***Optimal case normal operation***
Gas available	- - -	Low due to Poisson distribution	**50X+ Ethereum** ***Unlimited scale-out achieved by applying randomness in following techniques…***

3

Miscellanea

Death By Poisson Process

The Simplest Flaws Are The Worst…

50% of Ethereum blocks are empty !

Miners prefer to build on empty blocks

since no need validate/delay

= more profitable

An empty block has more chance being

confirmed….

Duh !

Bitcoin Could Consume as

Much Electricity as Denmark

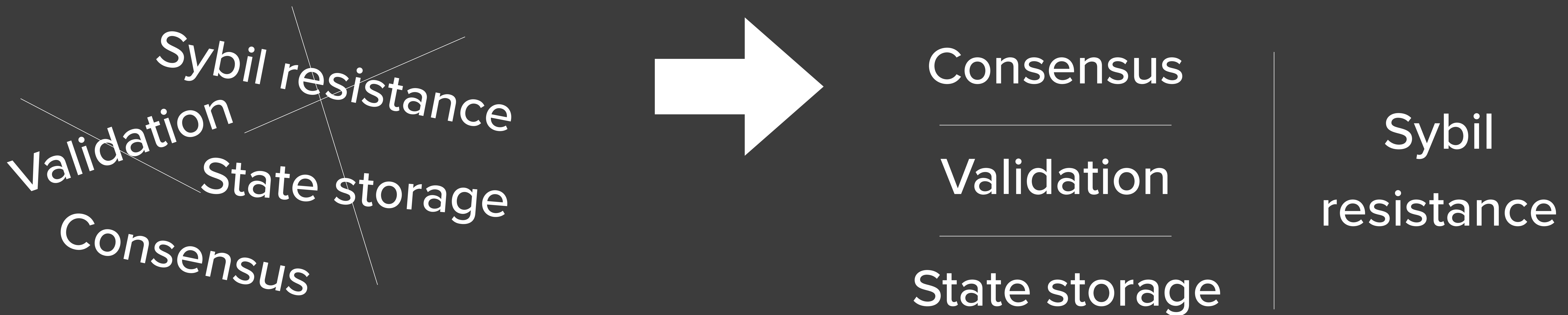
by 2020, Motherboard

3/29/2016

Separate and decouple concerns

Proof-of-Work Blockchain

DFINITY



Sybil resistance

Validation

State storage

Consensus

Consensus

Validation

State storage

Sybil resistance

Computer Science should not go out of fashion

TCP/IP

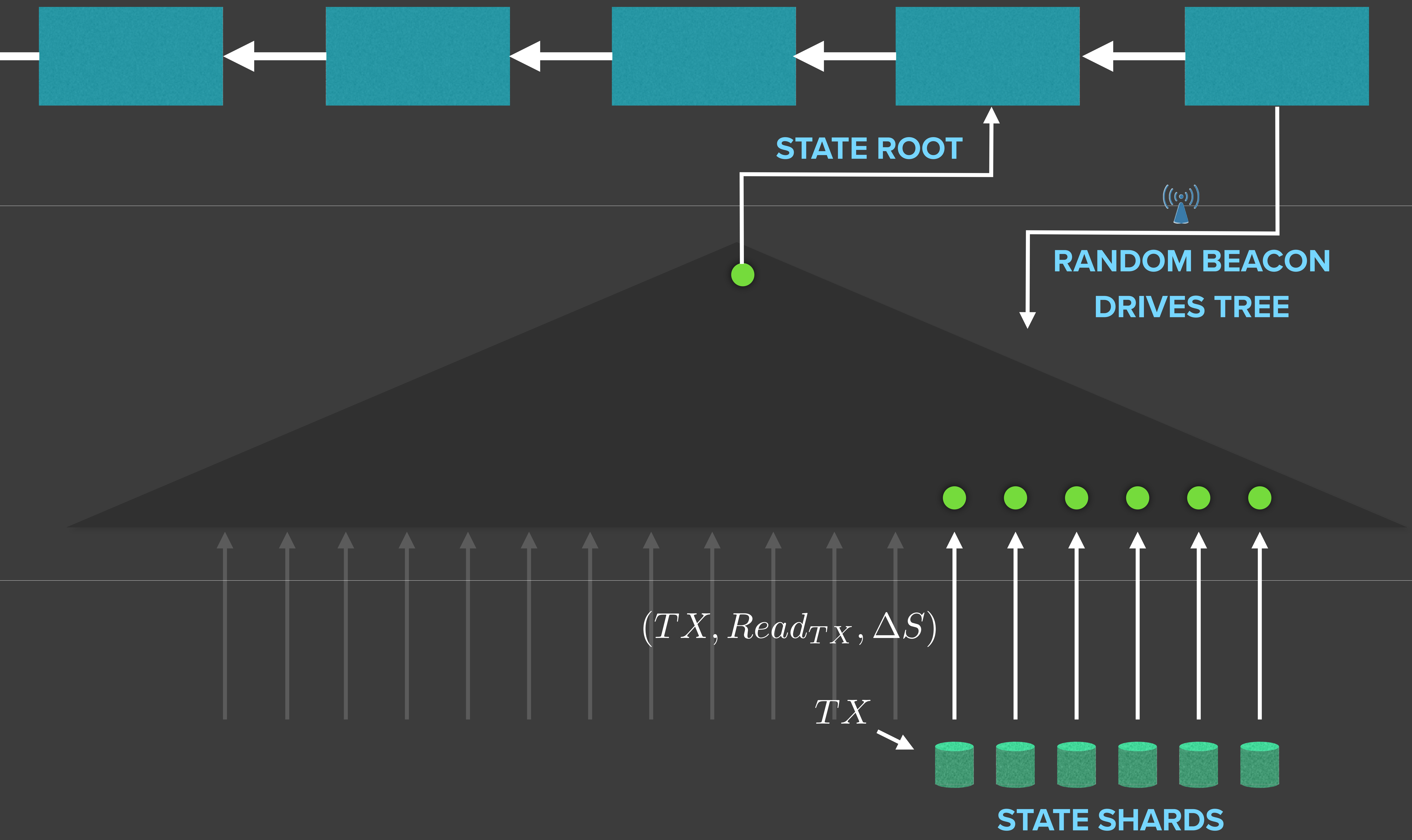
Application

Transport

Internet

Network Access

3 Layer “Scale-out” Architecture



CONSENSUS

Threshold relay chain
generates randomness,
records network metadata &
validation tree “state root”.

VALIDATION

Scalable “Validation Tree”
composed “Validation Towers”.
Does for validation what
Merkle does for data.

STORAGE

State and updates to state
stored on shards. State
transitions passed to
Validation Tree.

BLS Implementation

BLS Signature based on optimal Ate-pairing, C++/ASM

Shigeo Mitsunari, https://github.com/herumi/bls

Distributed Key Generation via Joint-Feldman Verifiable Secret Sharing, Go

Timo Hanke [about to be released, follow my Twitter @timothanke]

Threshold-Relay Simulator, Go

Timo Hanke [about to be released, follow my Twitter @timothanke]