

A black and white photograph of a city street at night. The scene is filled with tall, dark buildings that create a dense, vertical pattern. In the foreground, a car is parked on the left side of the street. The street itself is mostly empty, with a few small, indistinct figures of people visible in the distance. The overall atmosphere is moody and urban.

GET THE SLIDES NOW ➔



Build Web3 Applications on a Next Generation “Blockchain”

Introduction to Hedera

Ed Marquez – Hedera

Viv Diwakar – The HBAR Foundation



[/ed-marquez](https://www.linkedin.com/in/ed-marquez)

| [/in/vivdiwakar](https://www.linkedin.com/in/vivdiwakar)

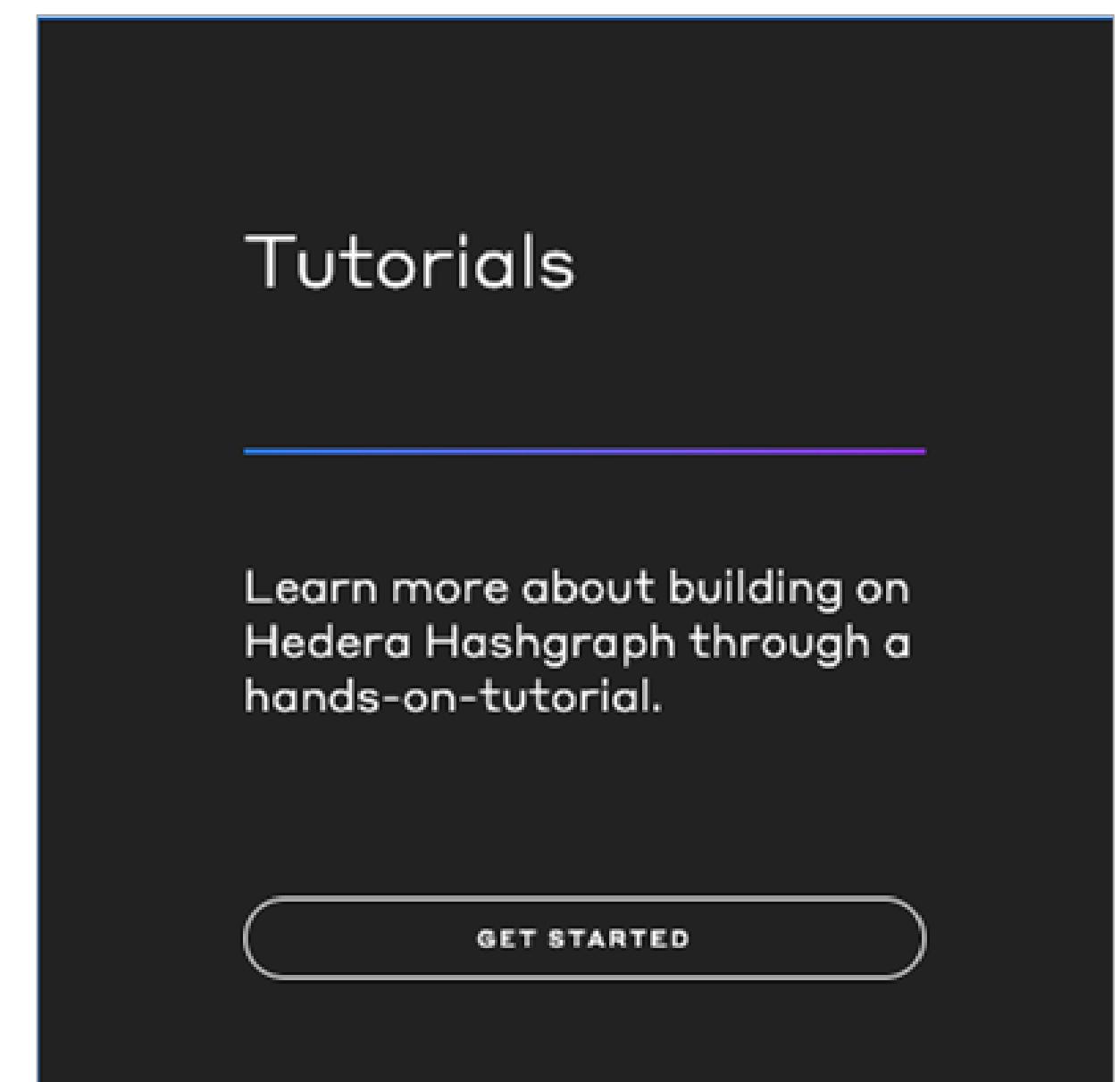
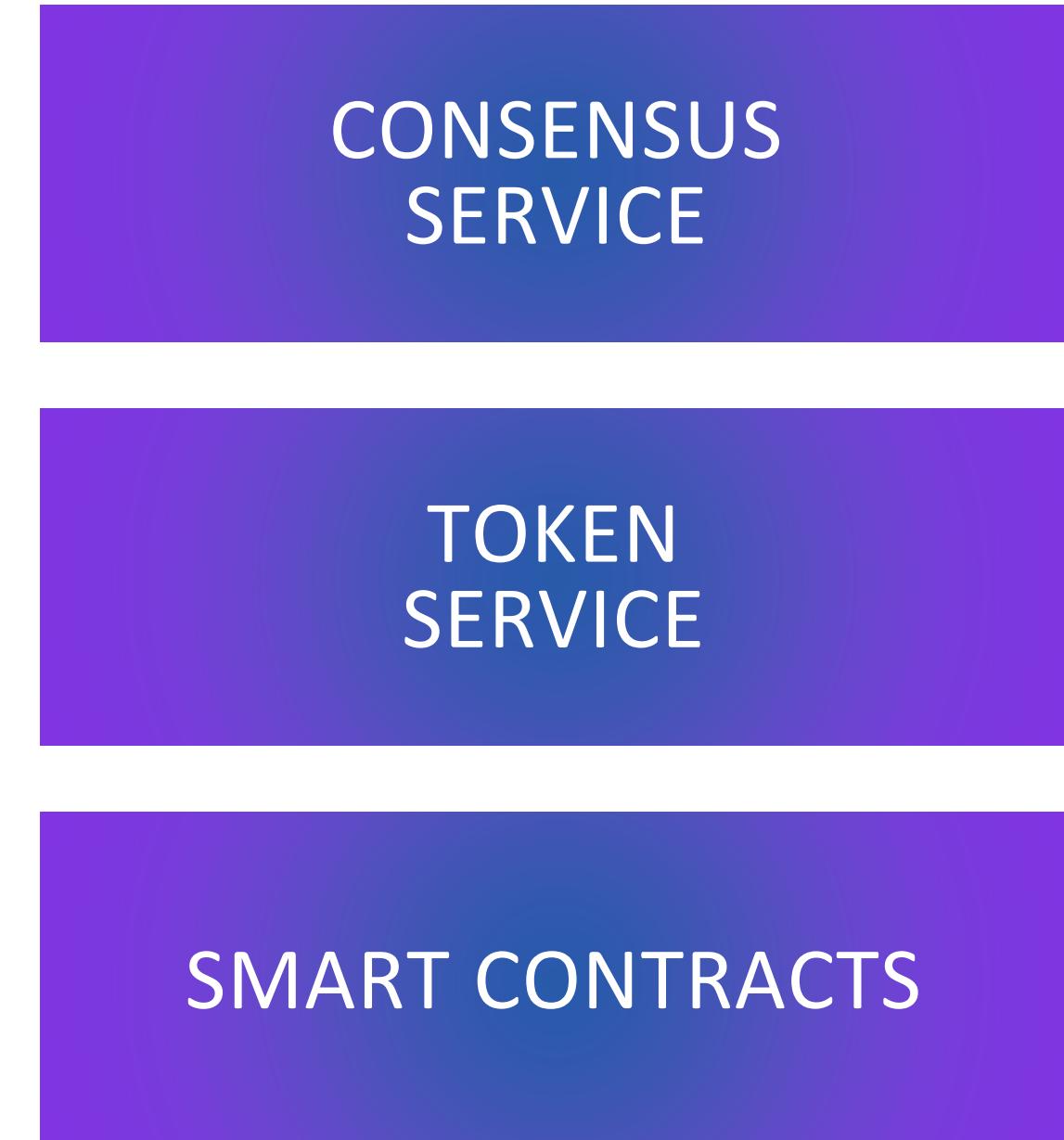
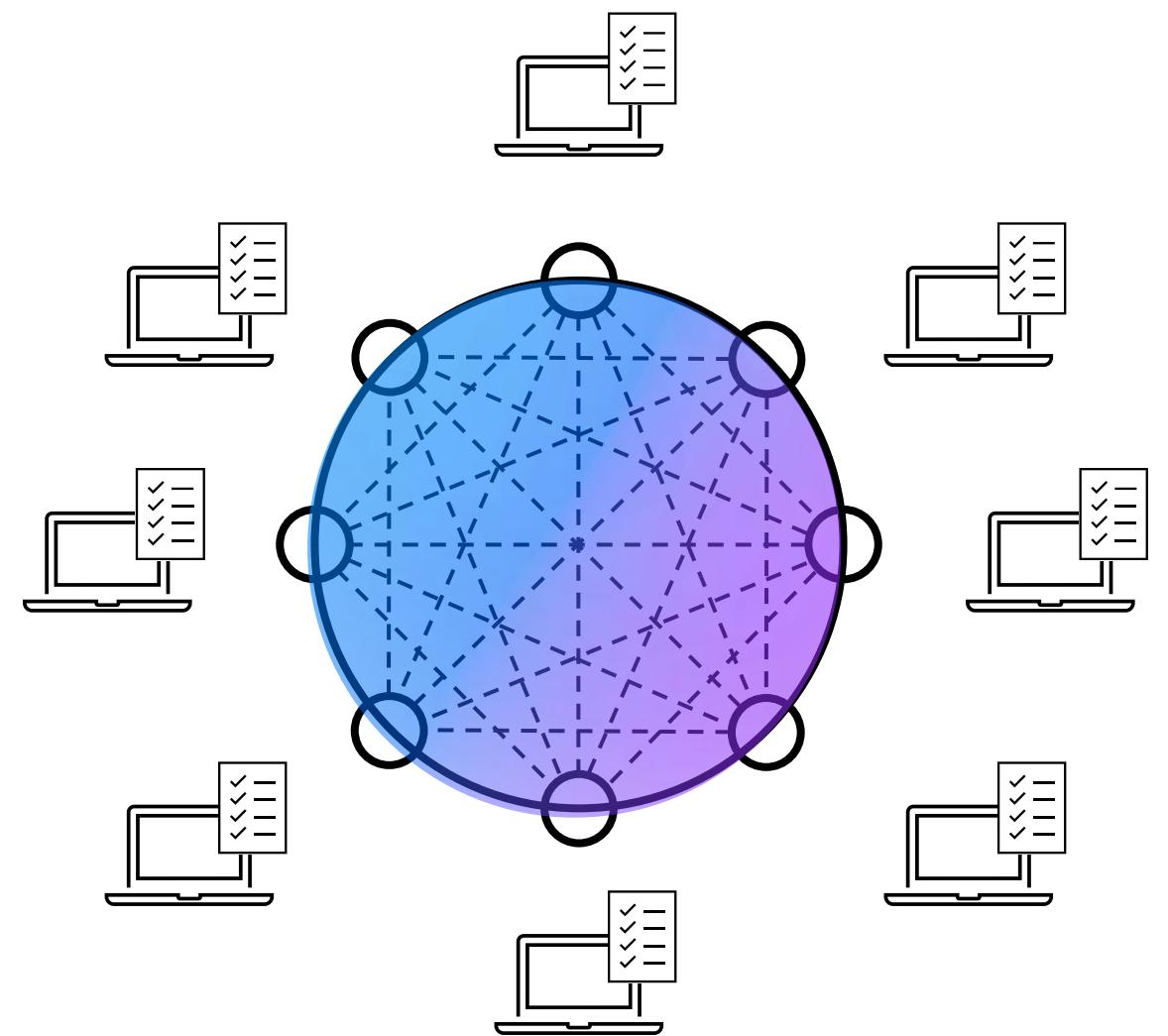


[@ed_marquez](https://twitter.com/ed_marquez)

| [@HBAR_foundation](https://twitter.com/HBAR_foundation)



In this session, you will learn how the **Hedera network functions**, how you can **start development**, and **where to learn more**



Introduction

Understand
the Hedera Network

Start Developing on
Hedera

Get Resources and
Learn More

What Is Web 3.0?

Forbes

Charles Silver Forbes Councils Member
Forbes Technology Council

Gartner

Blockchain's Big Bang: Web 3.0

By [Avivah Litan](#) | August 08, 2019 | 7 Comments

NEWSLETTERS • THE LEDGER

We live in an age of discovery at the dawn of Web 3.0

BY ROBERT HACKETT AND DECLAN HARTY

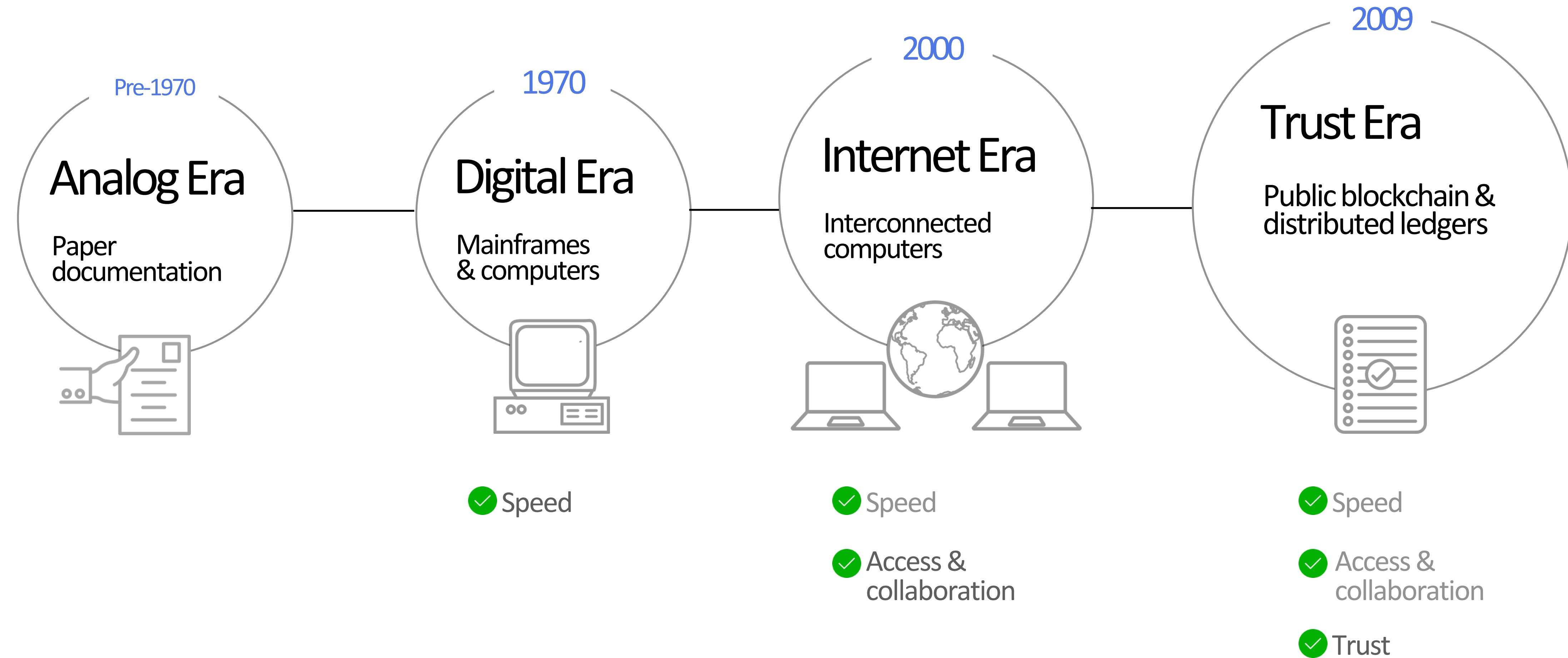
November 5, 2021 7:01 PM EDT

Web 3.0 can repair the attention-driven digital economy

Doug Petkanics @petkanics / 4:17 PM EST • November 8, 2021

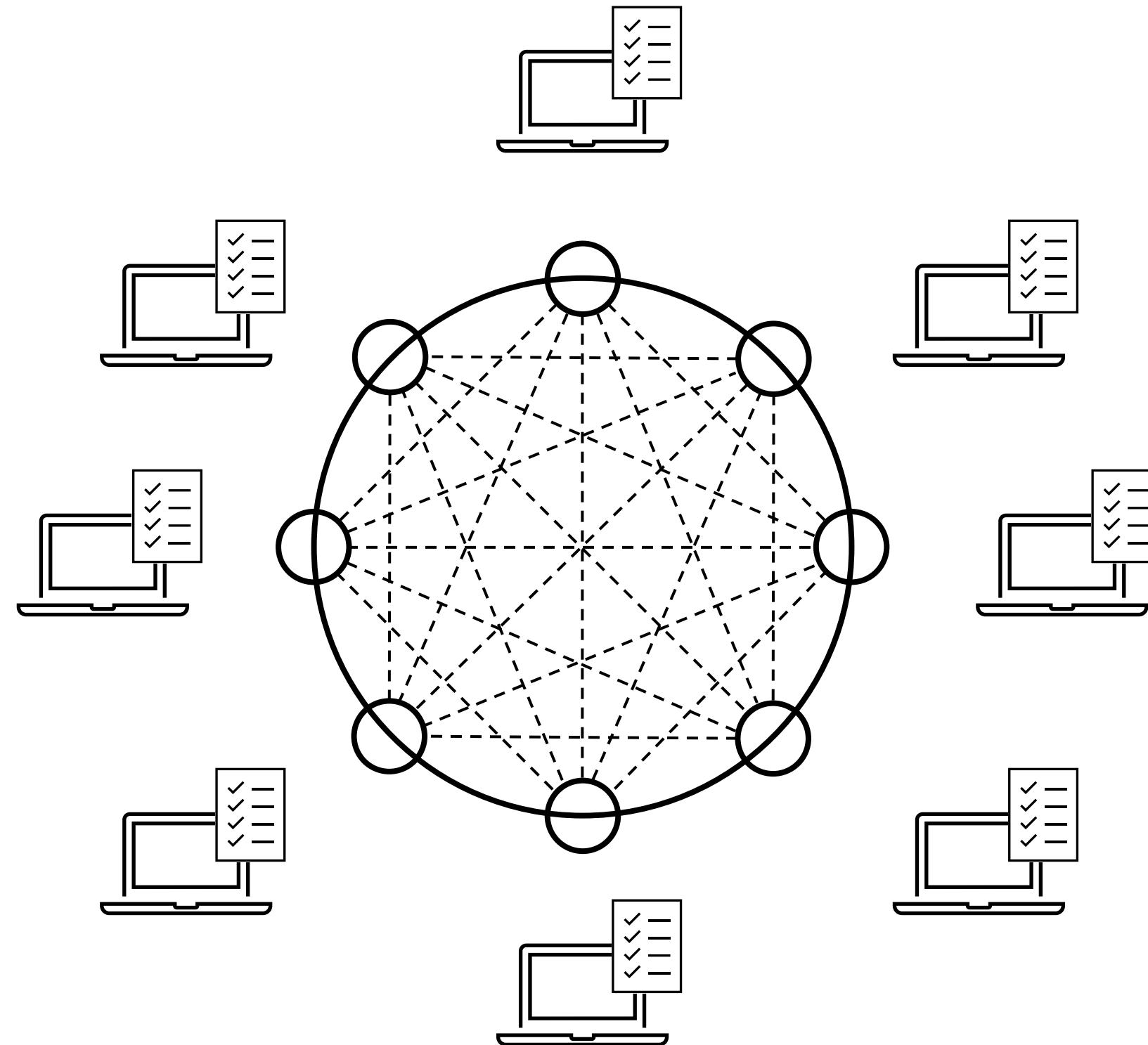
 Comment

The web evolves continuously and now it is becoming more decentralized



Distributed ledgers are a key component of Web 3.0 because of their qualities

DISTRIBUTED LEDGER



Entire network records and validates each transaction

CENTRALIZED LEDGER



Single authority verifies, records, and executes transactions

Distributed ledgers are a key component of Web 3.0 because of their qualities

No central point of failure to attack

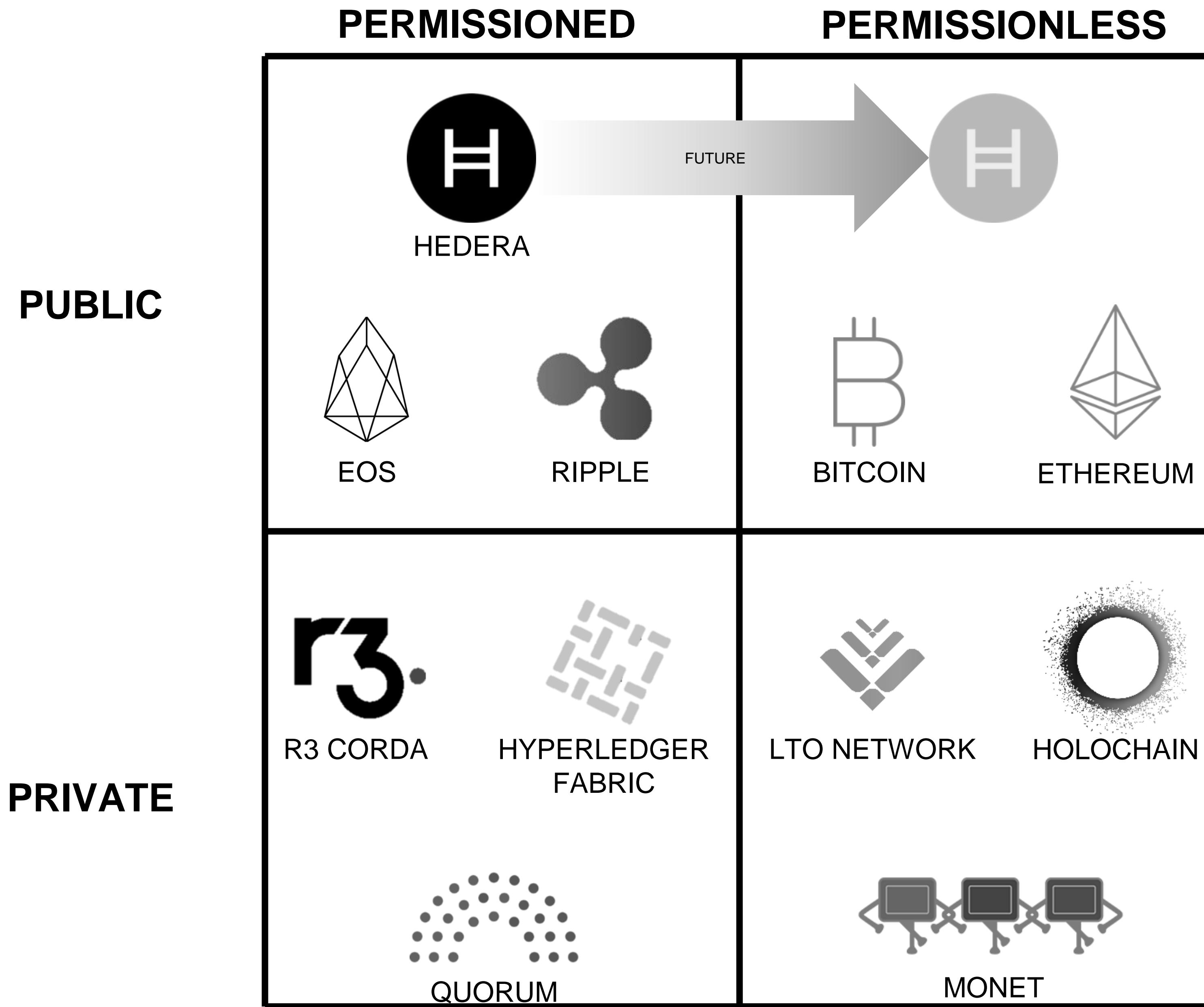


Reliant on strong cryptography to prove data integrity and tamper resistance

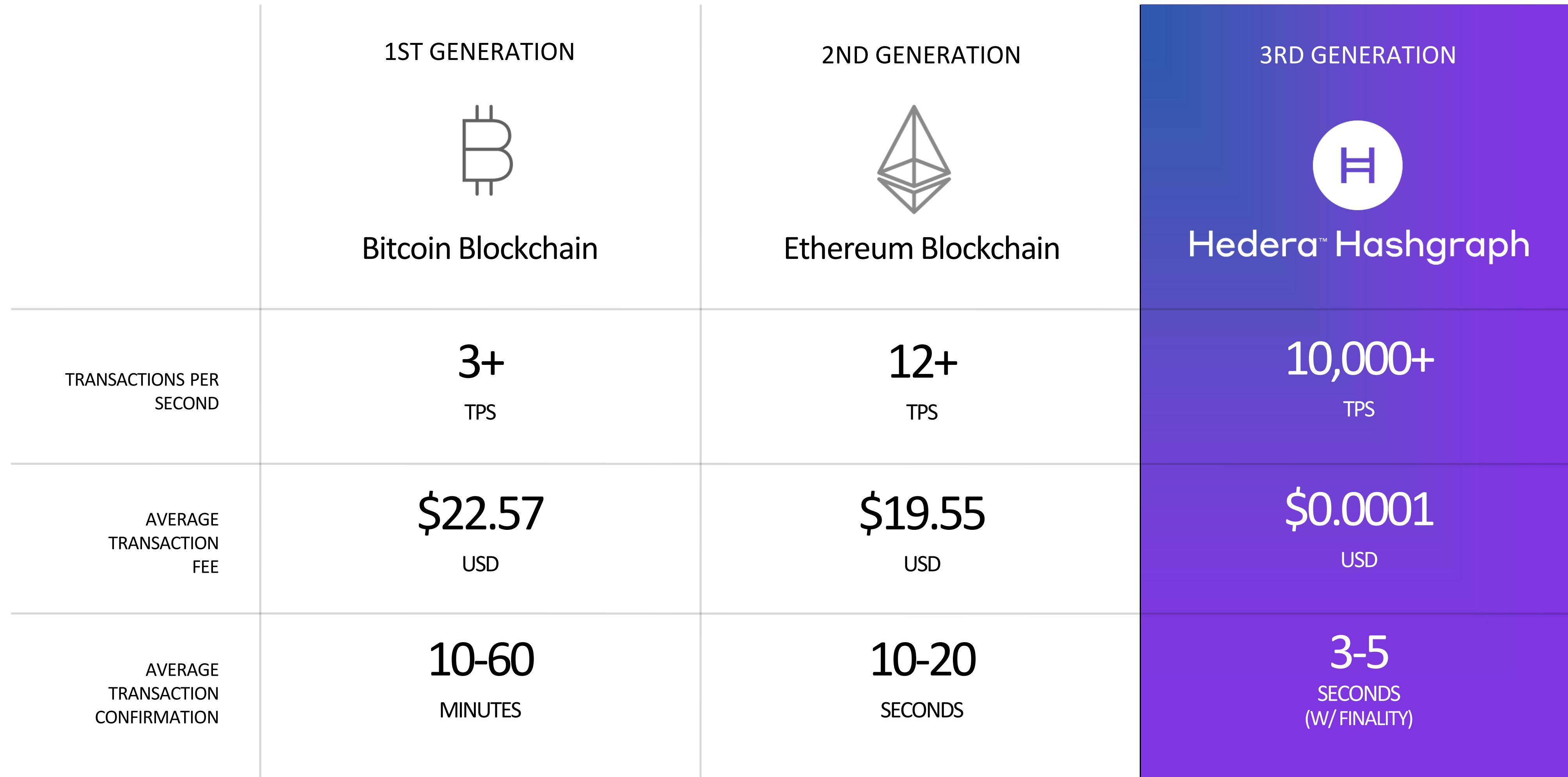


Rules for the ledger are determined by a governing concept of some sort

Require a consensus mechanism to determine the rules for adding new transactions to the state of the ledger



Hedera is a third-generation public distributed ledger



• Cryptocurrency transactions. For Hedera, range shown for transactions not requiring a transaction record, but can receive a transaction receipt.

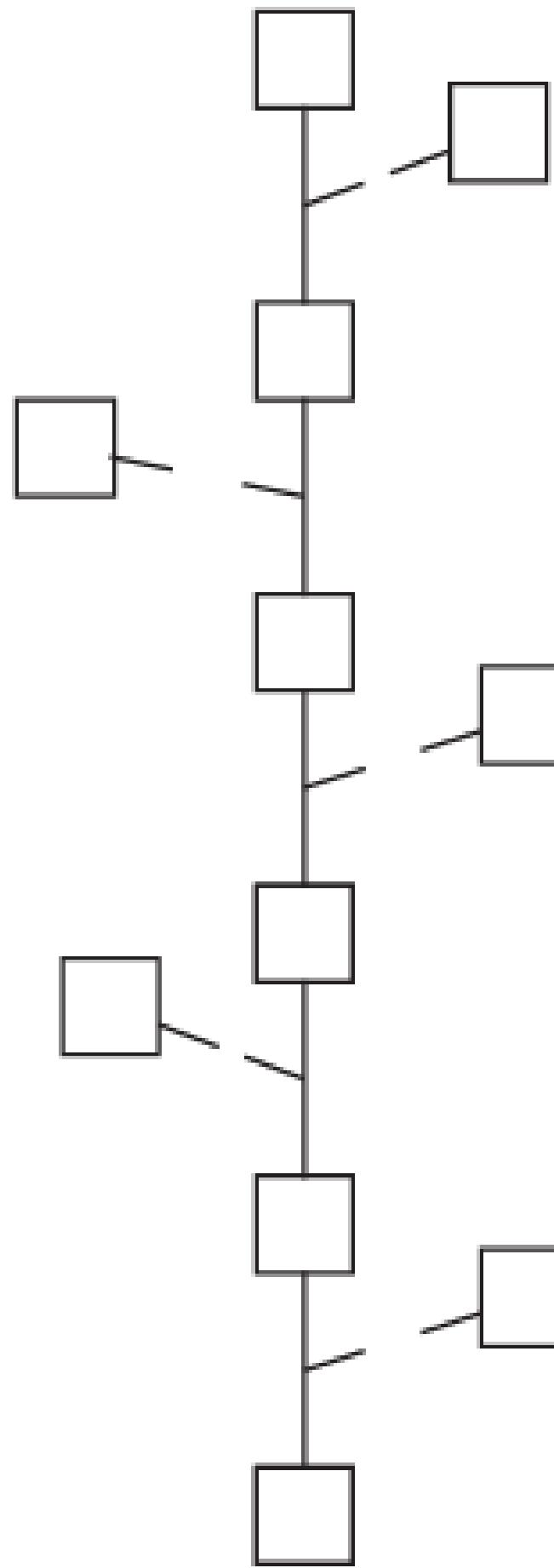
• Avg. Bitcoin tx fee from 6/26/20 - 9/24/20 from <https://blockchair.com/bitcoin/charts/average-transaction-fee-usd?interval=3m>

• Avg. Ethereum tx fee from 6/26/20 - 9/24/20 from <https://blockchair.com/ethereum/charts/average-transaction-fee-usd?interval=3m>

3rd Generation Public Ledgers

	 polygon	 Algorand	 Ethereum	 Hedera
Transactions per second	6,500* (claimed)	1,000† (approximate)	12+	10,000+
Average Fee (USD)	\$0.0020 (variable)	0.001 Algo (fixed)	\$19.55 (variable)	\$0.0001^ (fixed)
Time to Confirmation	5 - 10 seconds (leader block creation)	5 seconds (leader block creation)	10 - 20 seconds (leader block creation)	3-5 seconds (with finality)
Energy use per transaction	90+ kWh	0.00534 kWh	102+ kWh	0.00017 kWh

TRADITIONAL BLOCKCHAINS



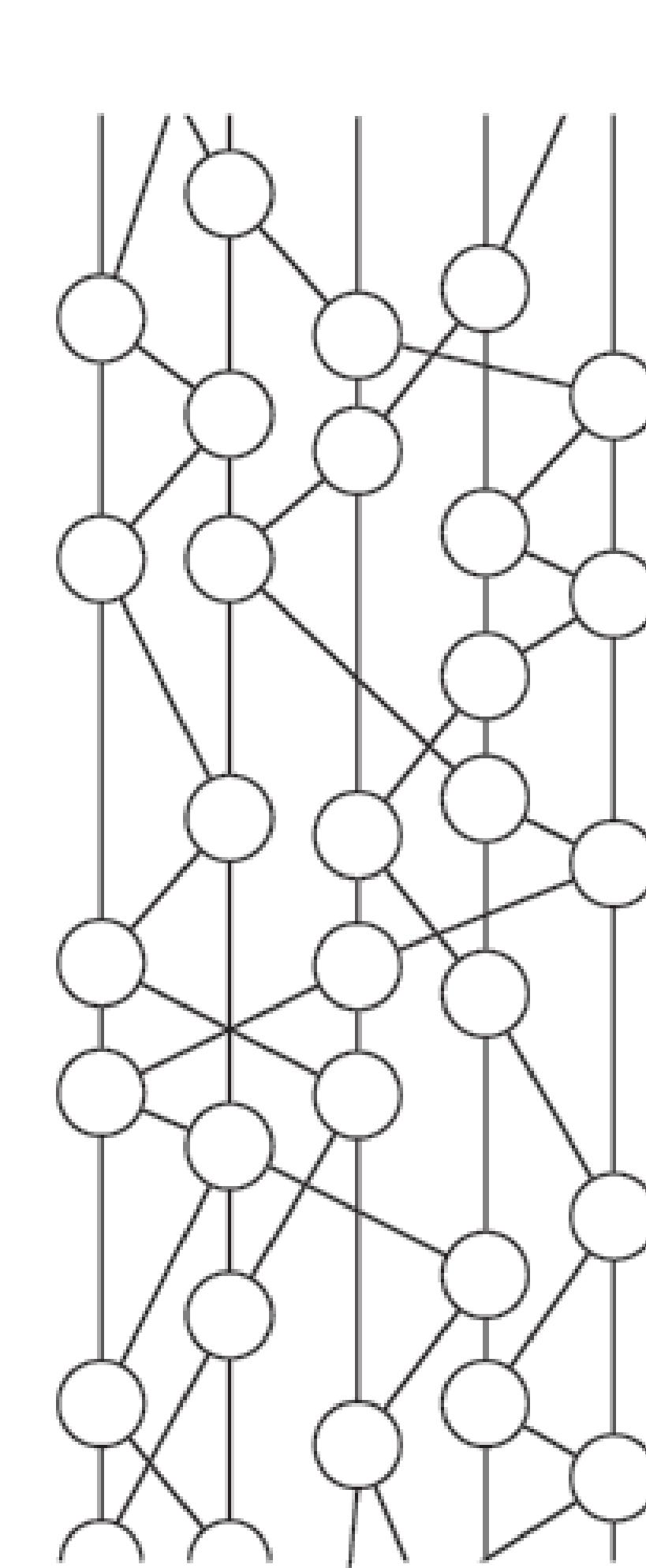
Proof-of-Work (PoW) typically has heavy electrical/computational requirements

No guaranteed consensus, just a “good enough” probability

Inefficient - “stale” blocks are pruned

Can be difficult to scale beyond ~1,000 tps

HASHGRAPH



Hashgraph is an **open-source** consensus algorithm and data structure

Uses a directed acyclic graph (DAG) and novel inventions

- Gossip about gossip
- Virtual voting

Hedera currently supports 10,000+ cryptocurrency transactions per second

Finality within 3-5 seconds

TRADITIONAL BLOCKCHAINS



HASHGRAPH





14,000+ DEVELOPERS

Attending global hackathons, meetups, and active in Discord.

100+ APPLICATIONS

In production on Hedera Mainnet, since open access on Sept. 2019

>4,000,000+ TXS/DAY

Far surpassing the daily transaction volume of Ethereum.

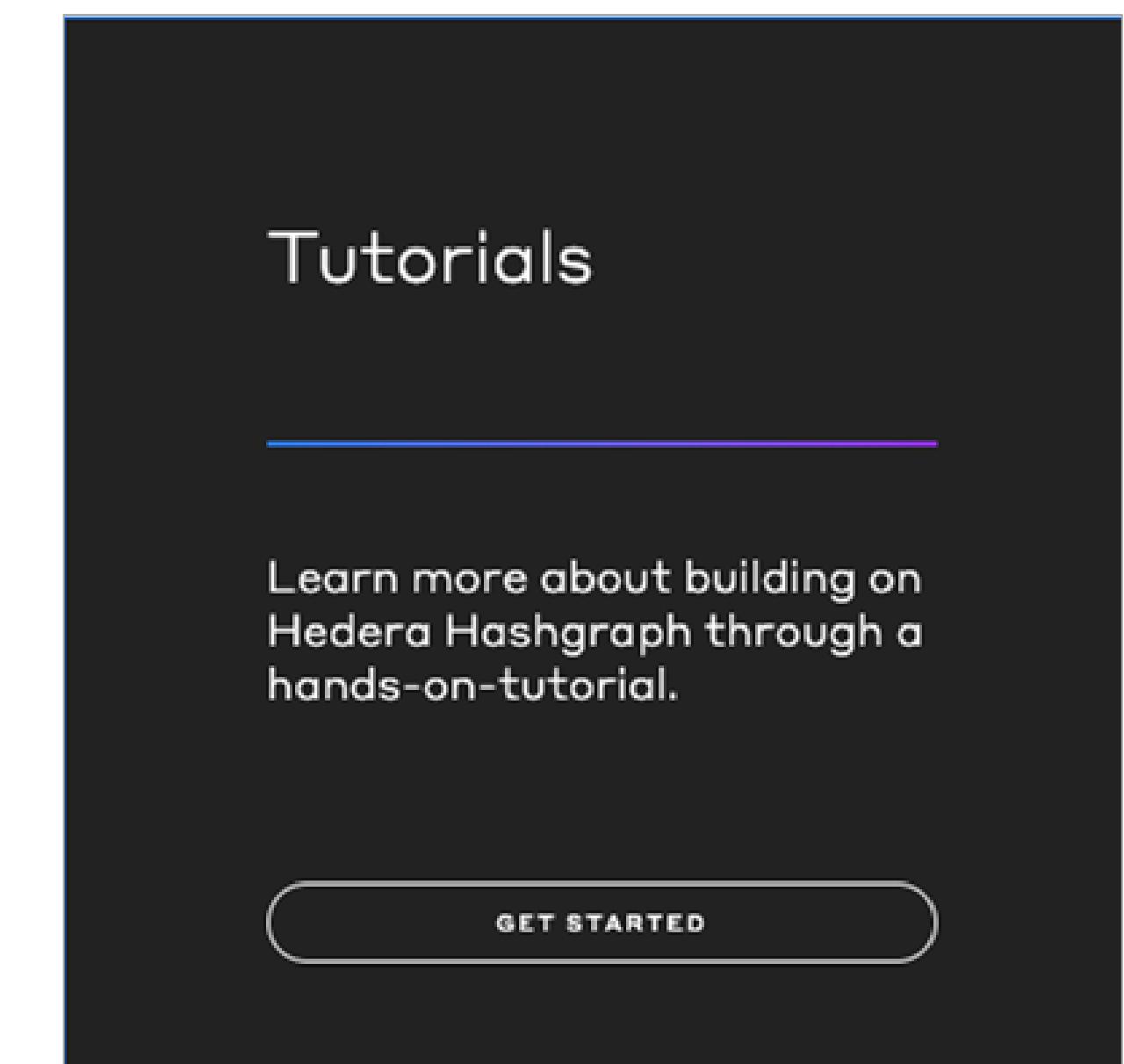
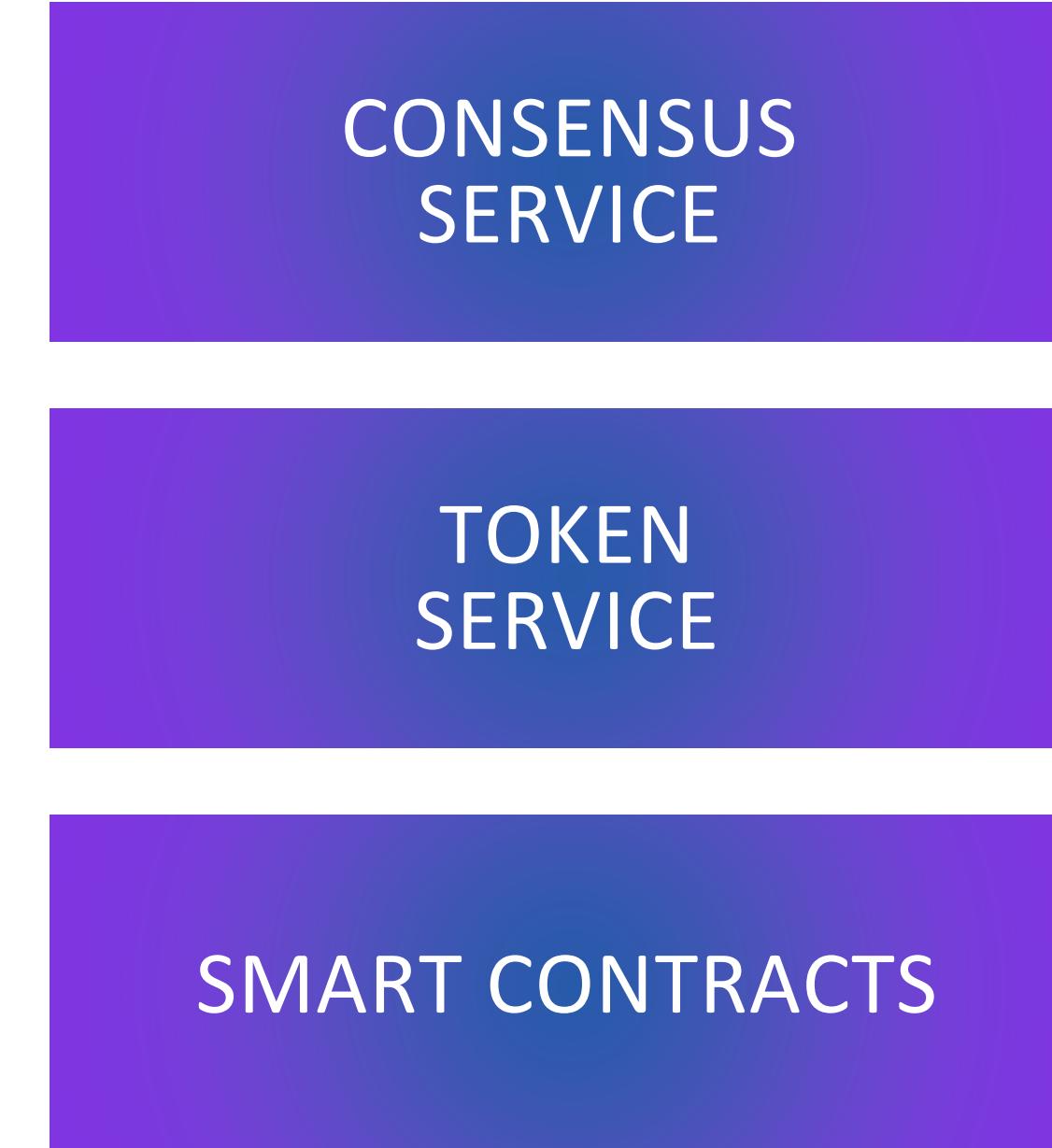
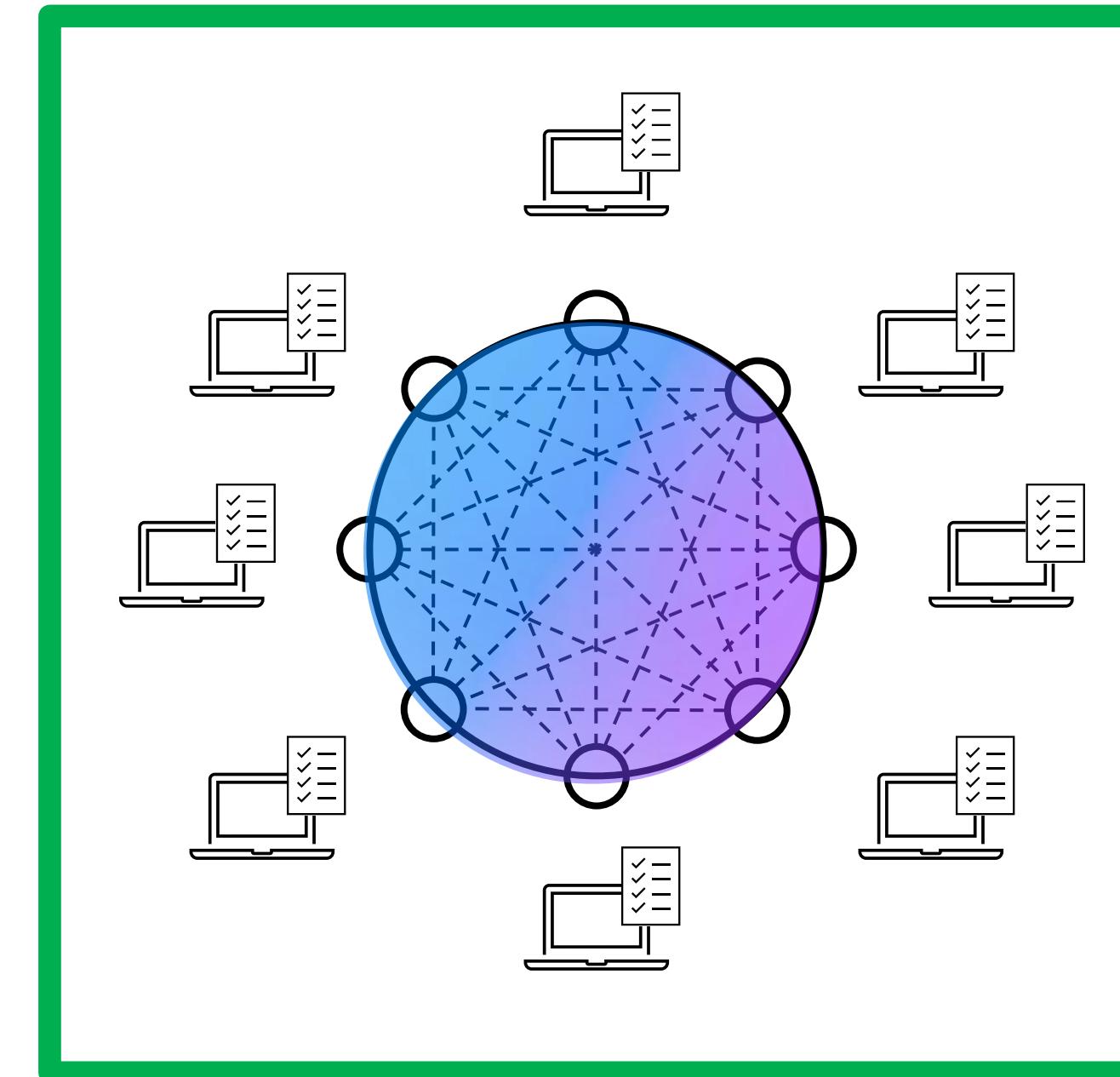
>2,300,000,000+ TXS TO DATE

Most used public DLT surpassing total transaction volume of Ethereum.

>700,000+ TOTAL ACCOUNTS ON MAINNET

49,500 accounts created in February 2022 alone.

In this session, you will learn how the **Hedera network functions**, how you can **start development**, and **where to learn more**

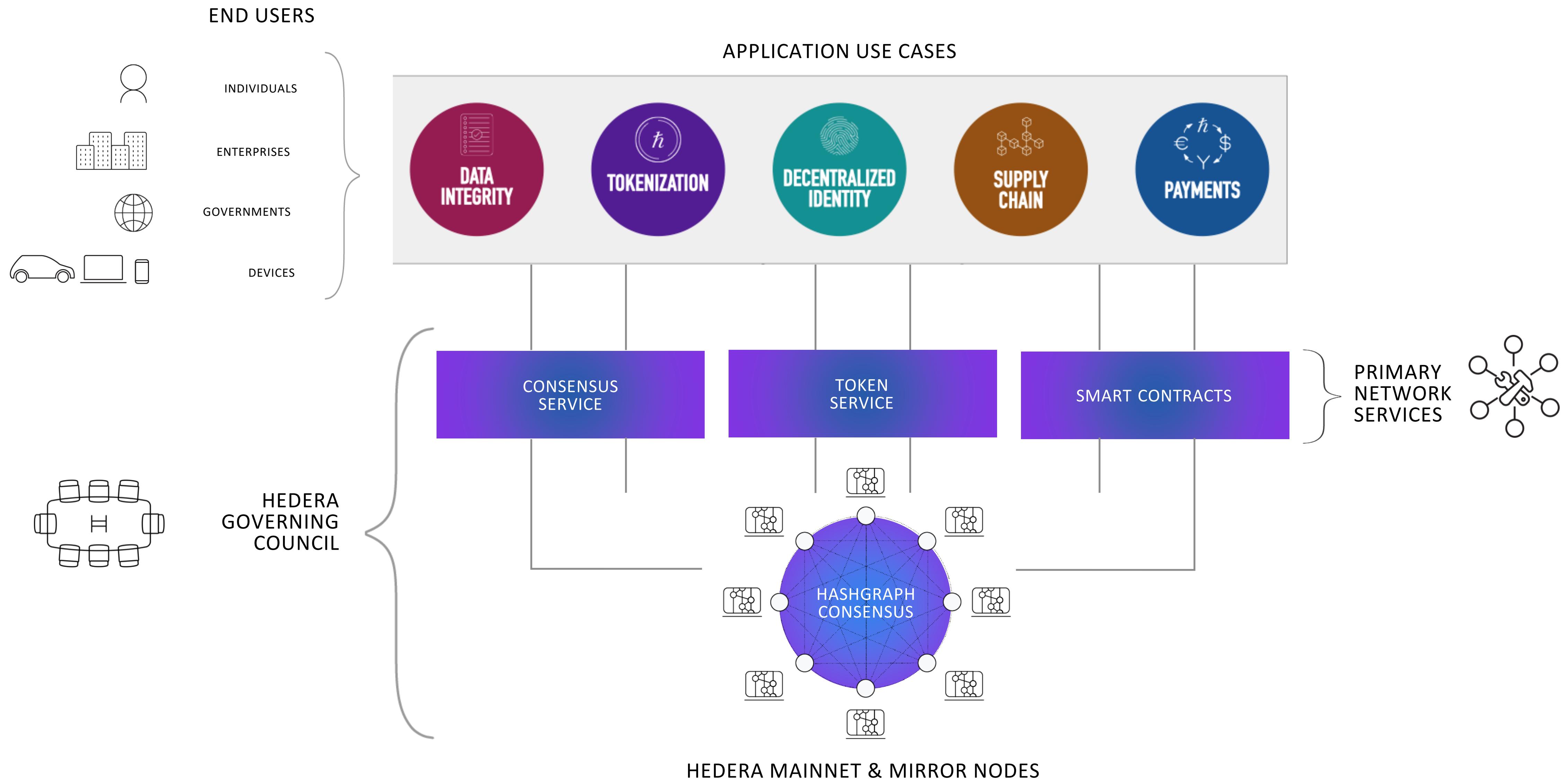


Introduction

Understand
the Hedera Network

Start Developing on
Hedera

Get Resources and
Learn More



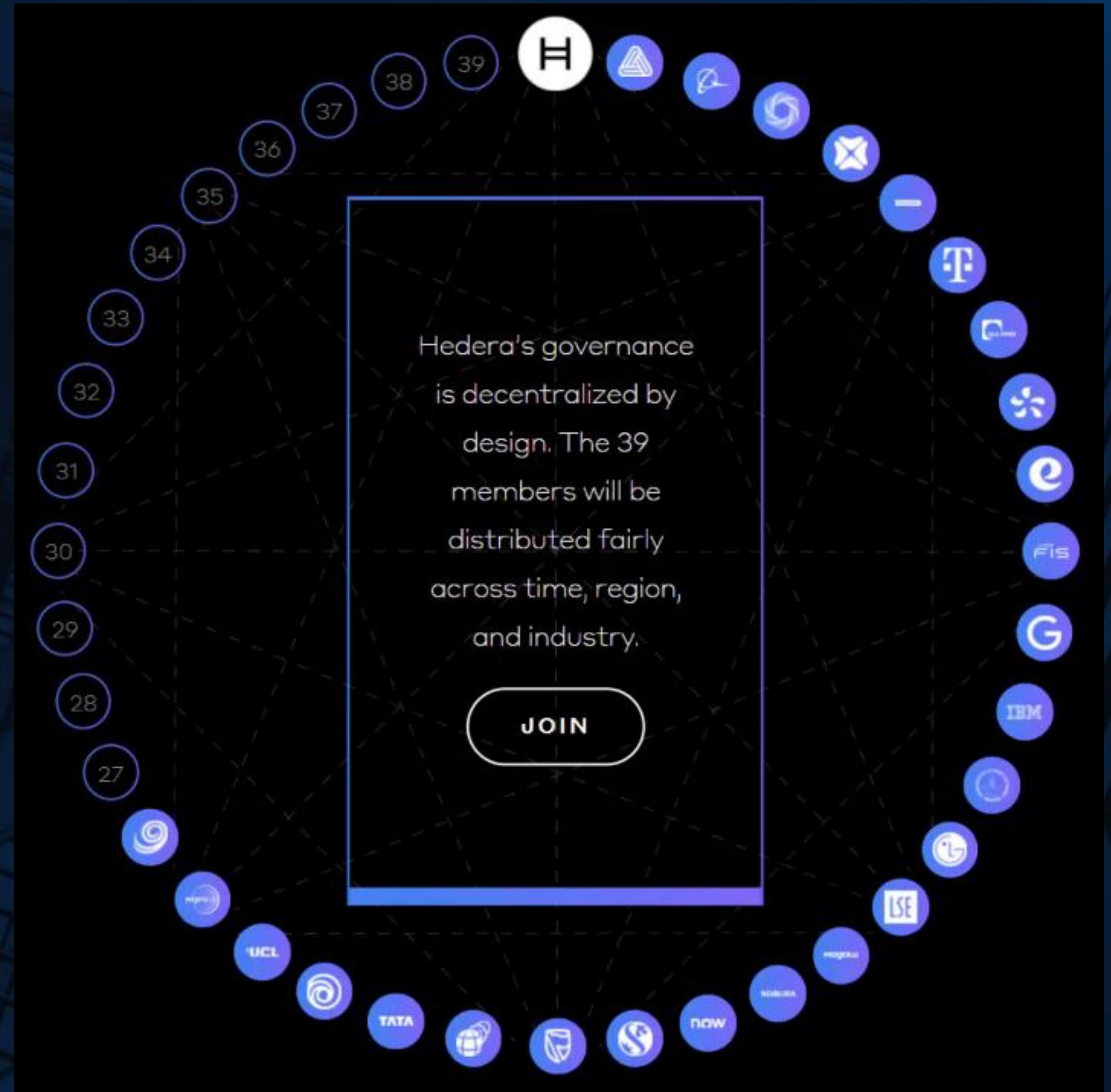
Hedera Governing Council



Up to **39** leading global organizations

20+ unique industries across **5** continents, touching all major markets

3-year maximum term, with up to 2 consecutive terms



2.6% influence per member (equal vote)

7 committees:
Membership, Marketing, Finance, Audit, Technical Steering/Product Pricing, Legal & Regulatory

Every member required to run a network node



Chainlink Labs



DENTONS



Deutsche
Telekom

NOMURA

magalu

eftpos



SHINHAN BANK

FiS Google

IBM



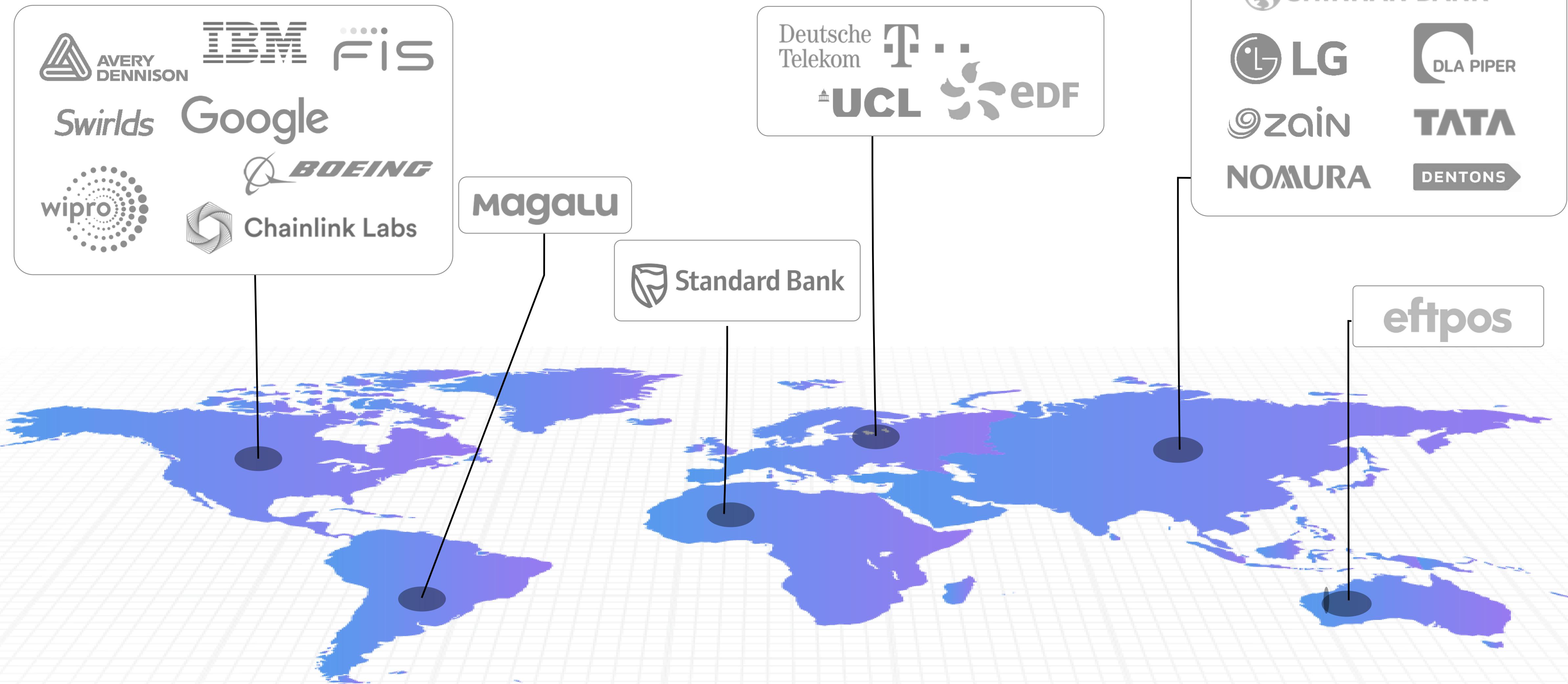
**HEDERA
GOVERNING COUNCIL**

Google

Building the future together

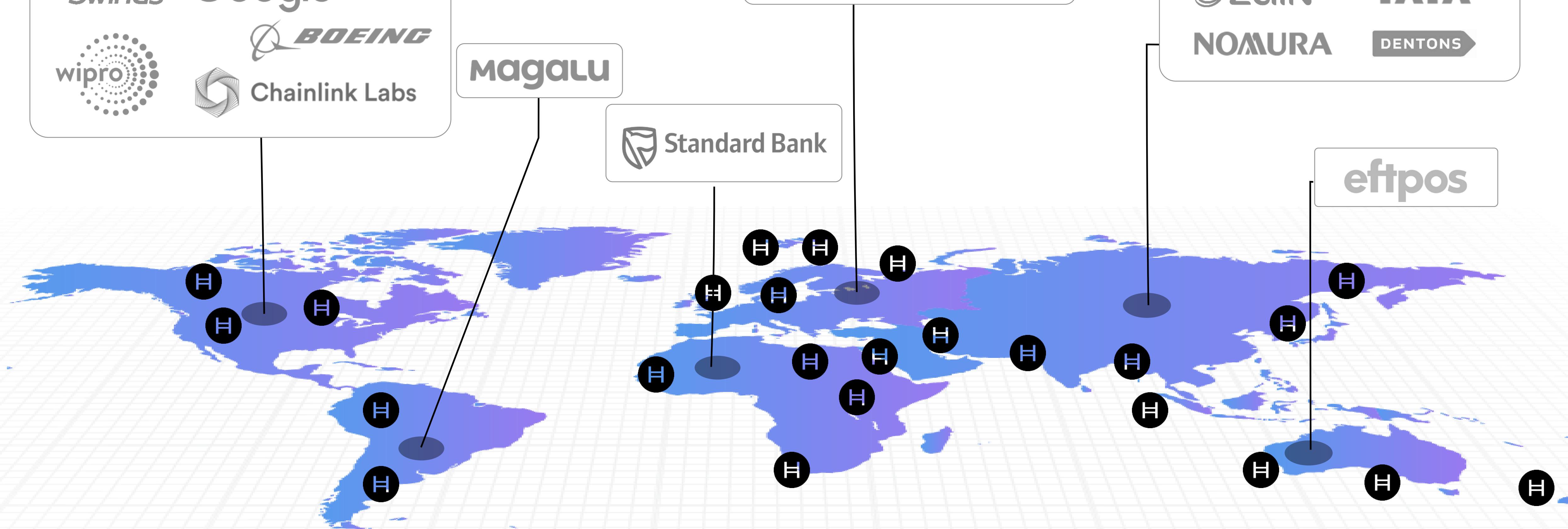
NETWORK GROWTH OVER TIME

PHASE 1: Up to 39 council members



NETWORK GROWTH OVER TIME

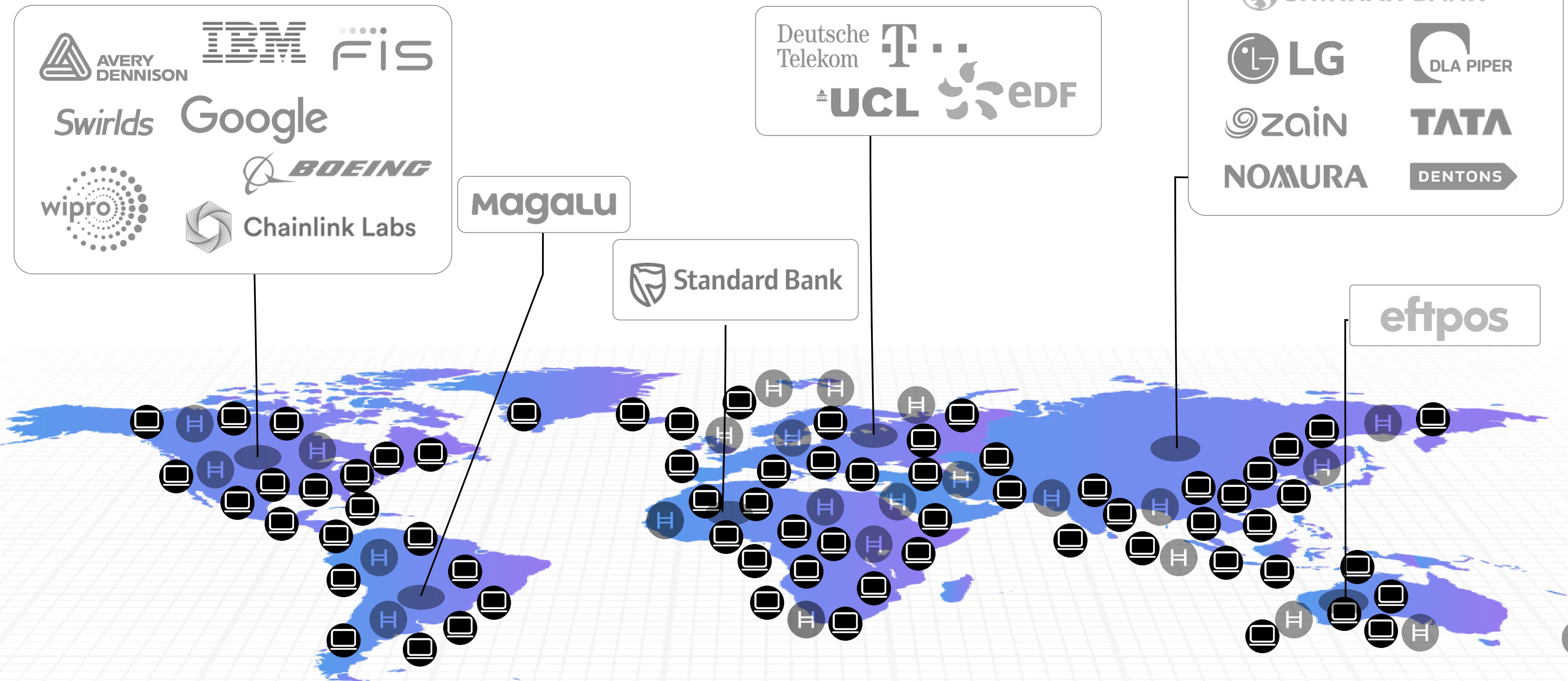
PHASE 2: 100s of KYC'd permissioned nodes



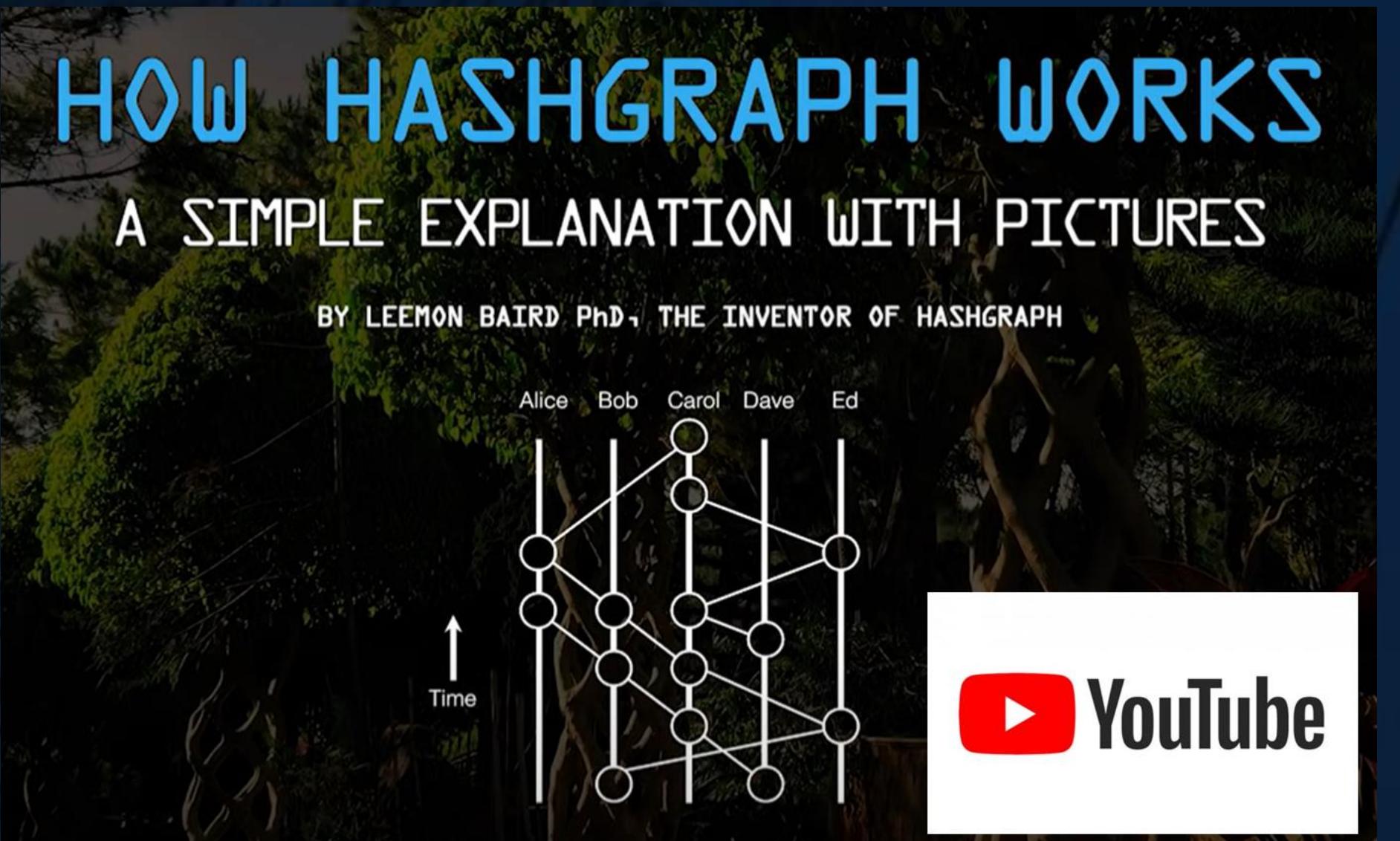
Hedera™ Hashgraph

NETWORK GROWTH OVER TIME

PHASE 3: 1000s of permissionless nodes

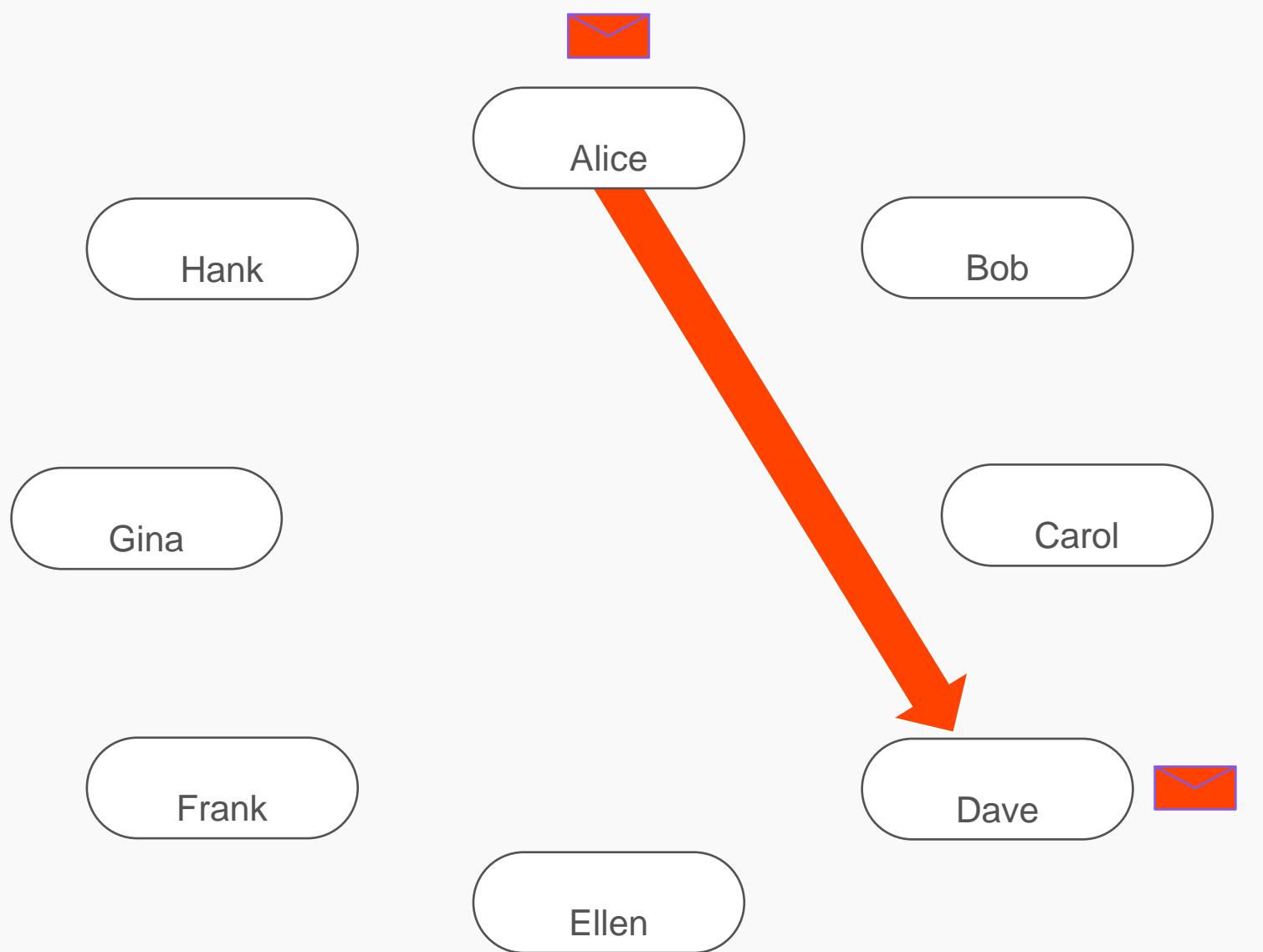


A brief overview of how hashgraph works

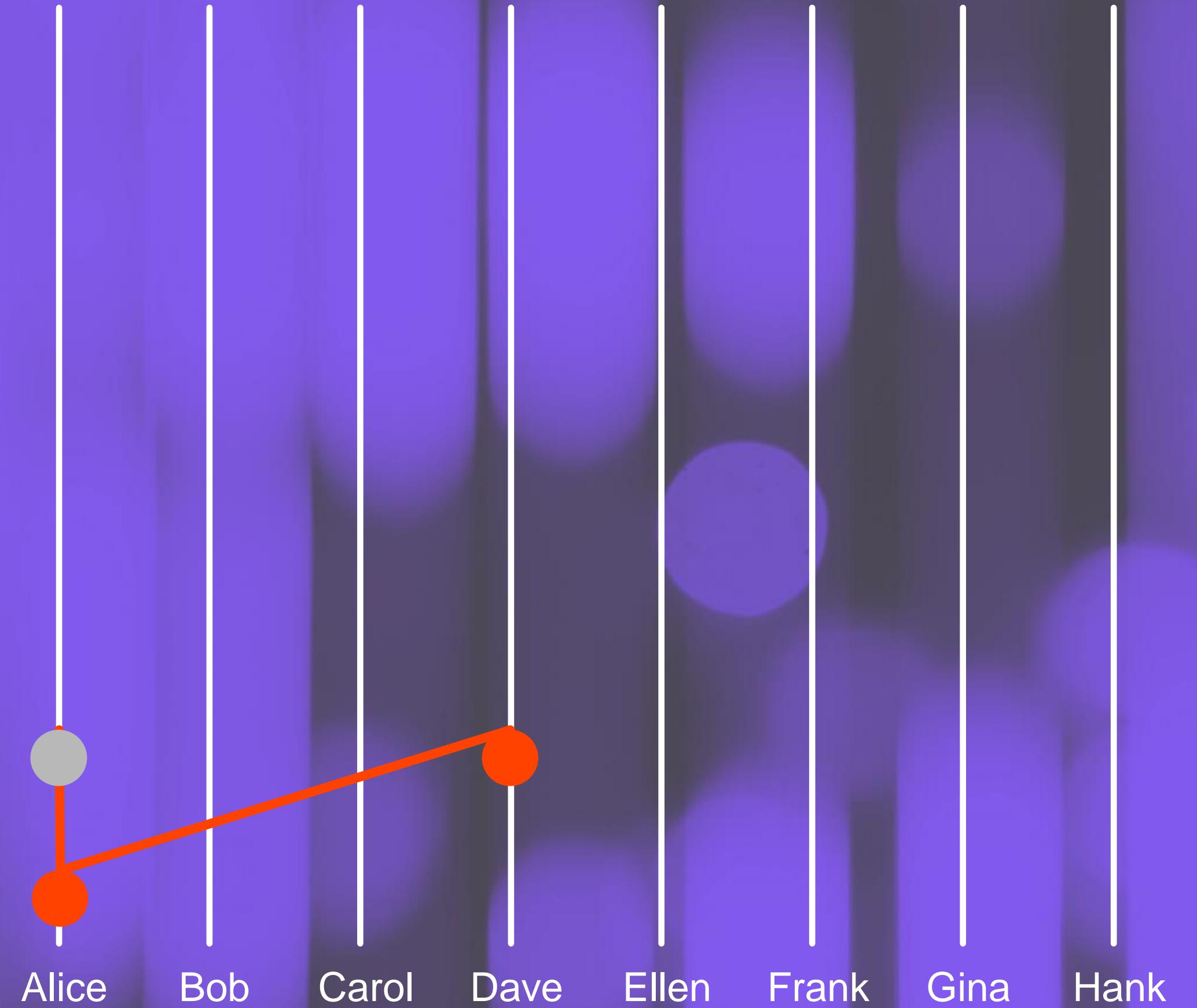


hedera.com

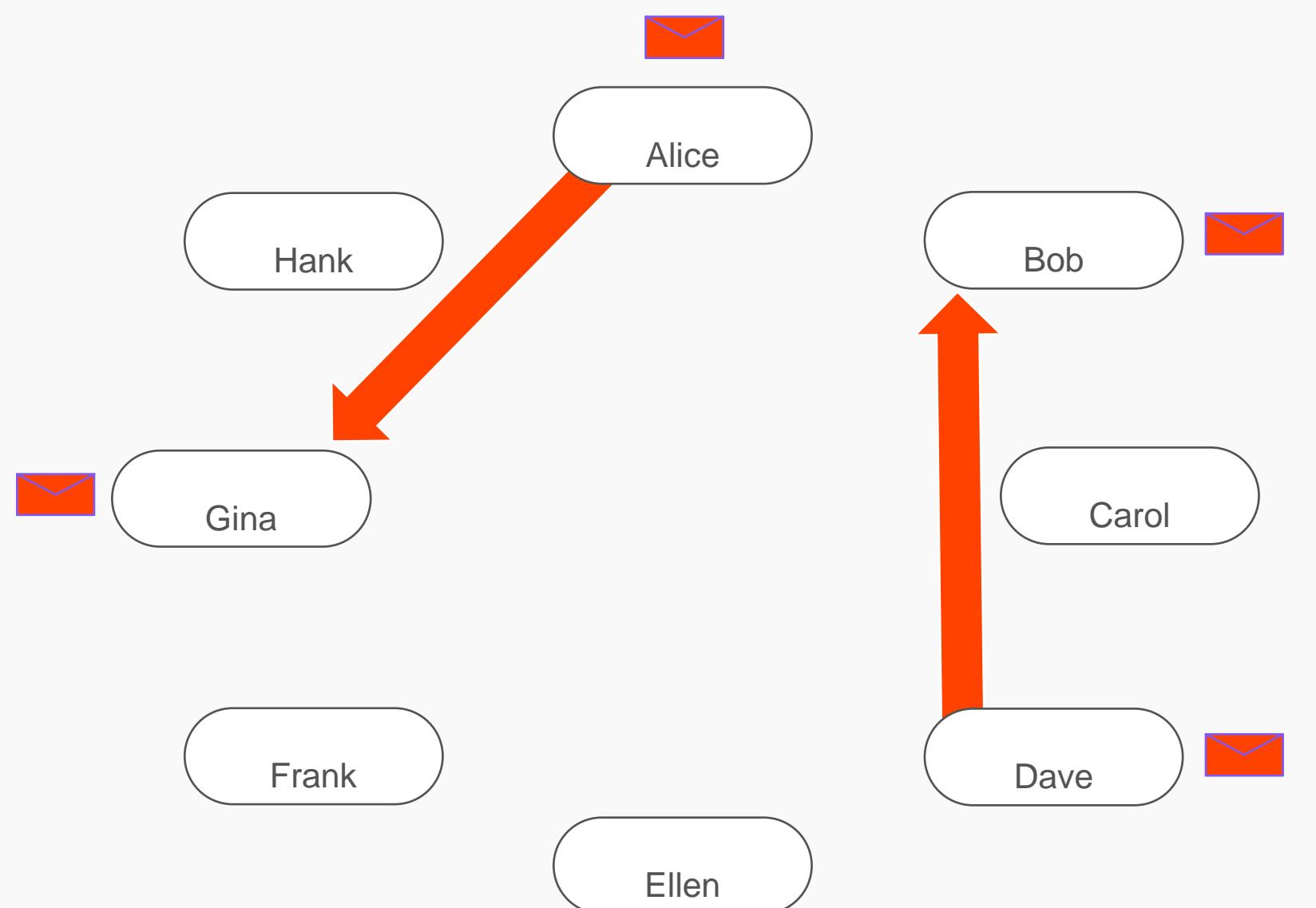
Gossip about gossip



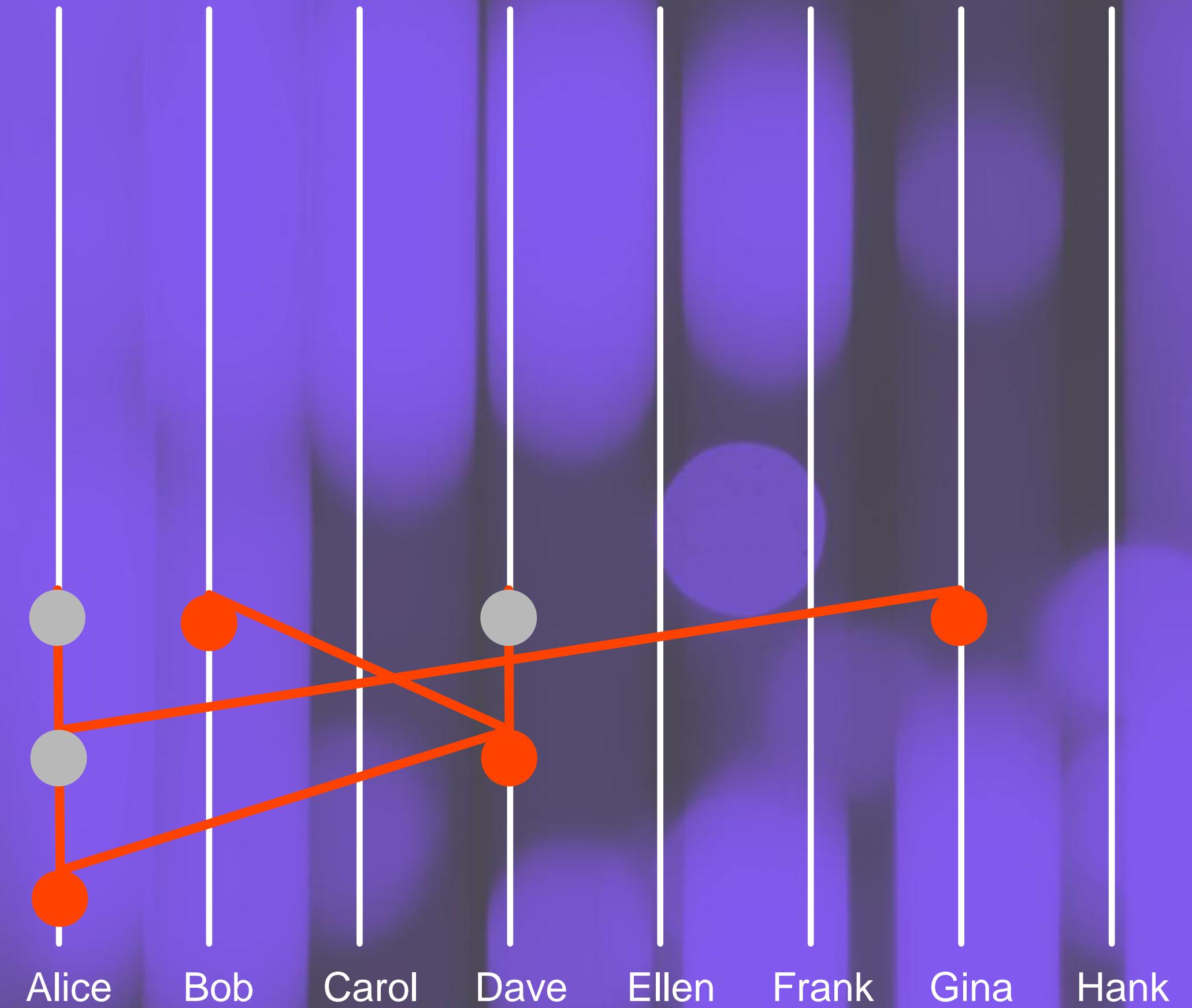
Gossip about Gossip



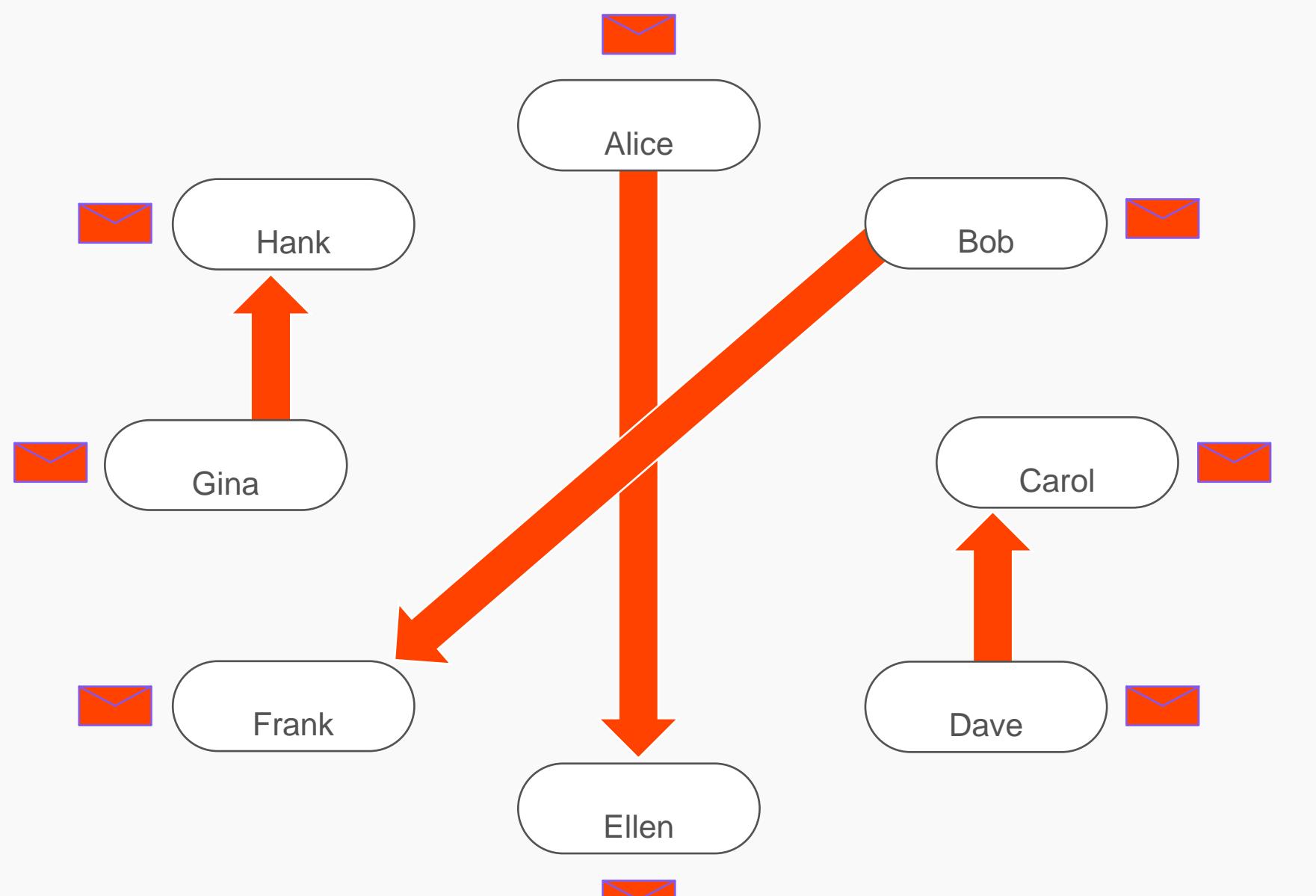
Gossip about gossip



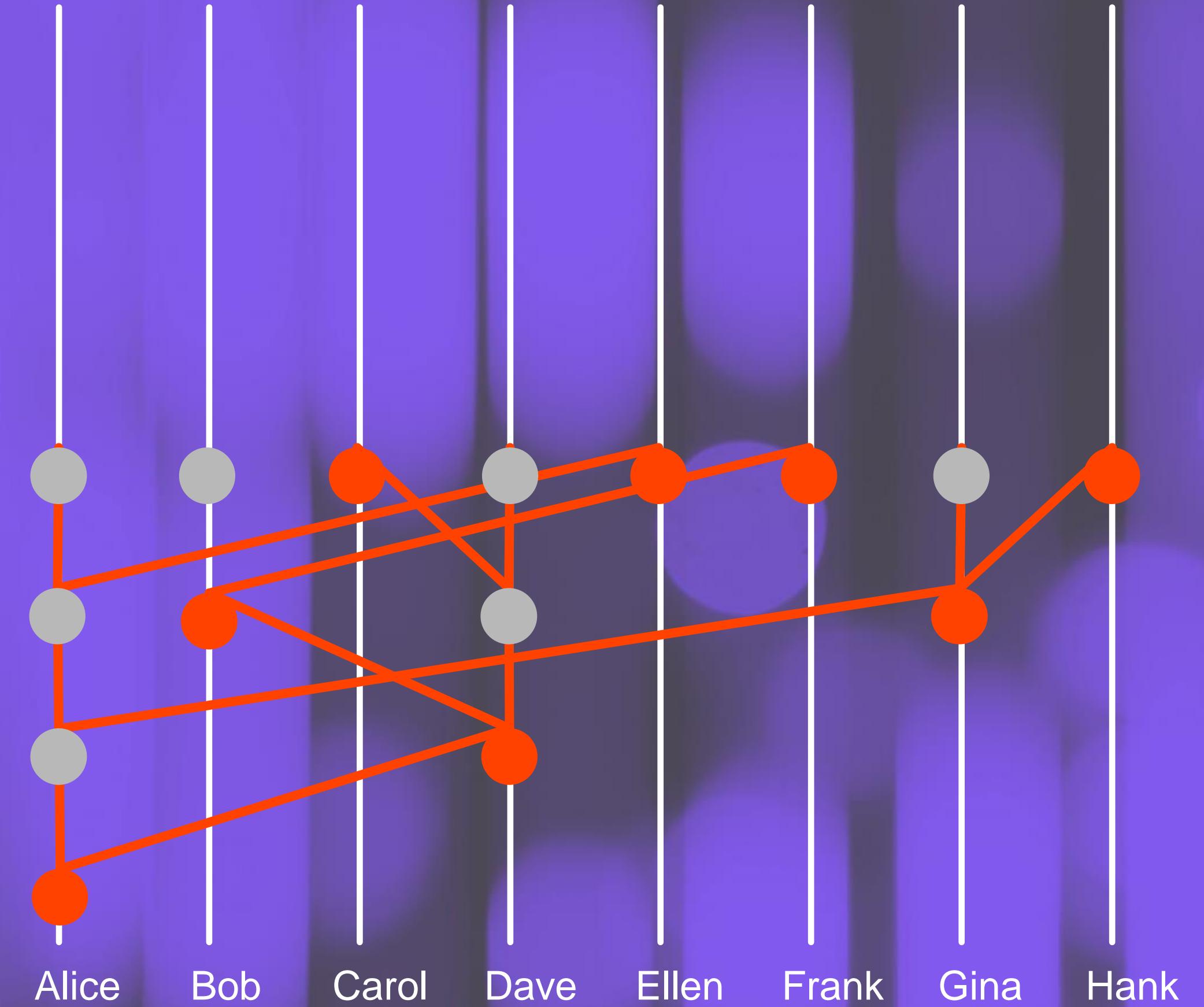
Gossip about Gossip



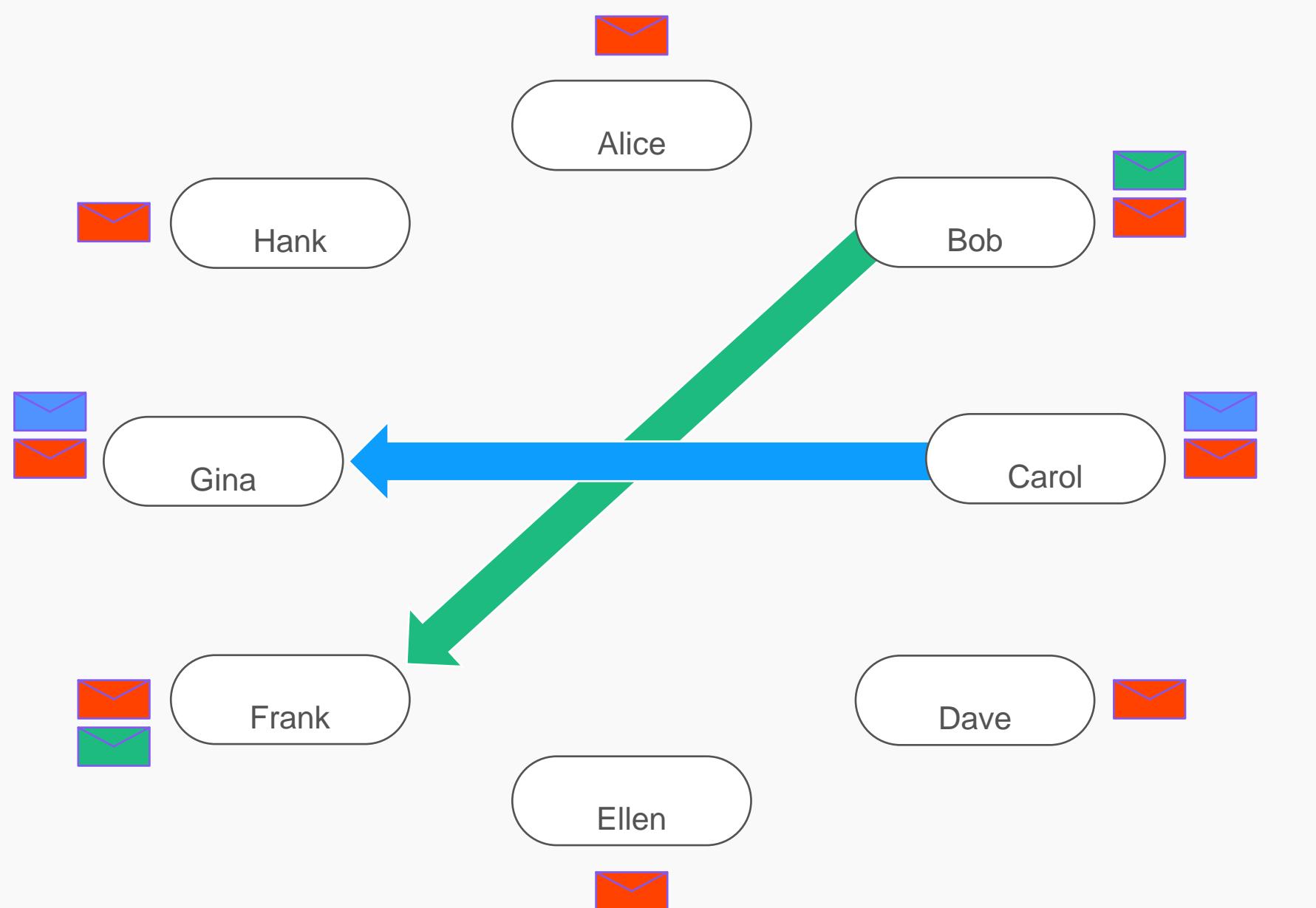
Gossip about gossip



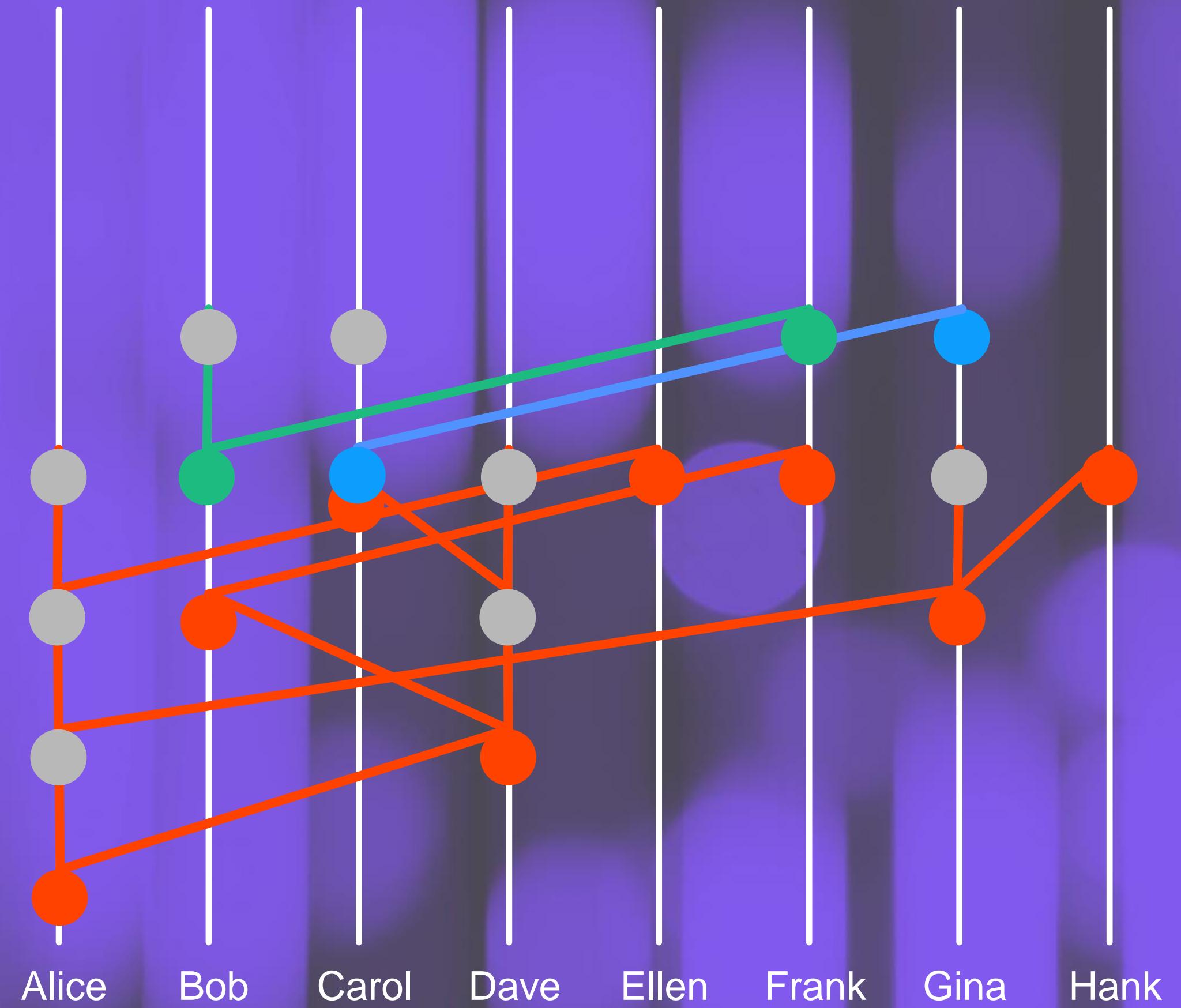
Gossip about Gossip



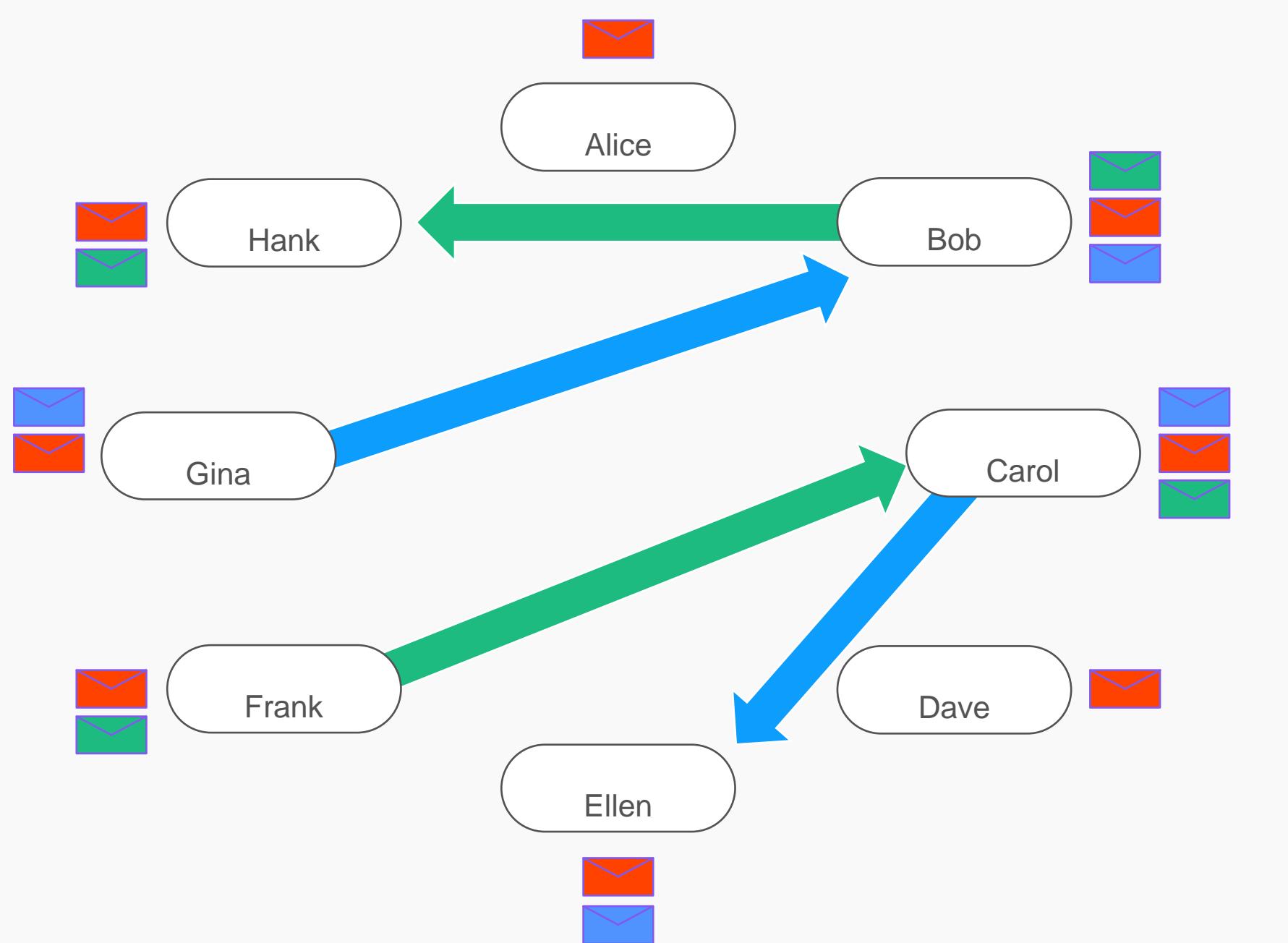
Gossip about gossip



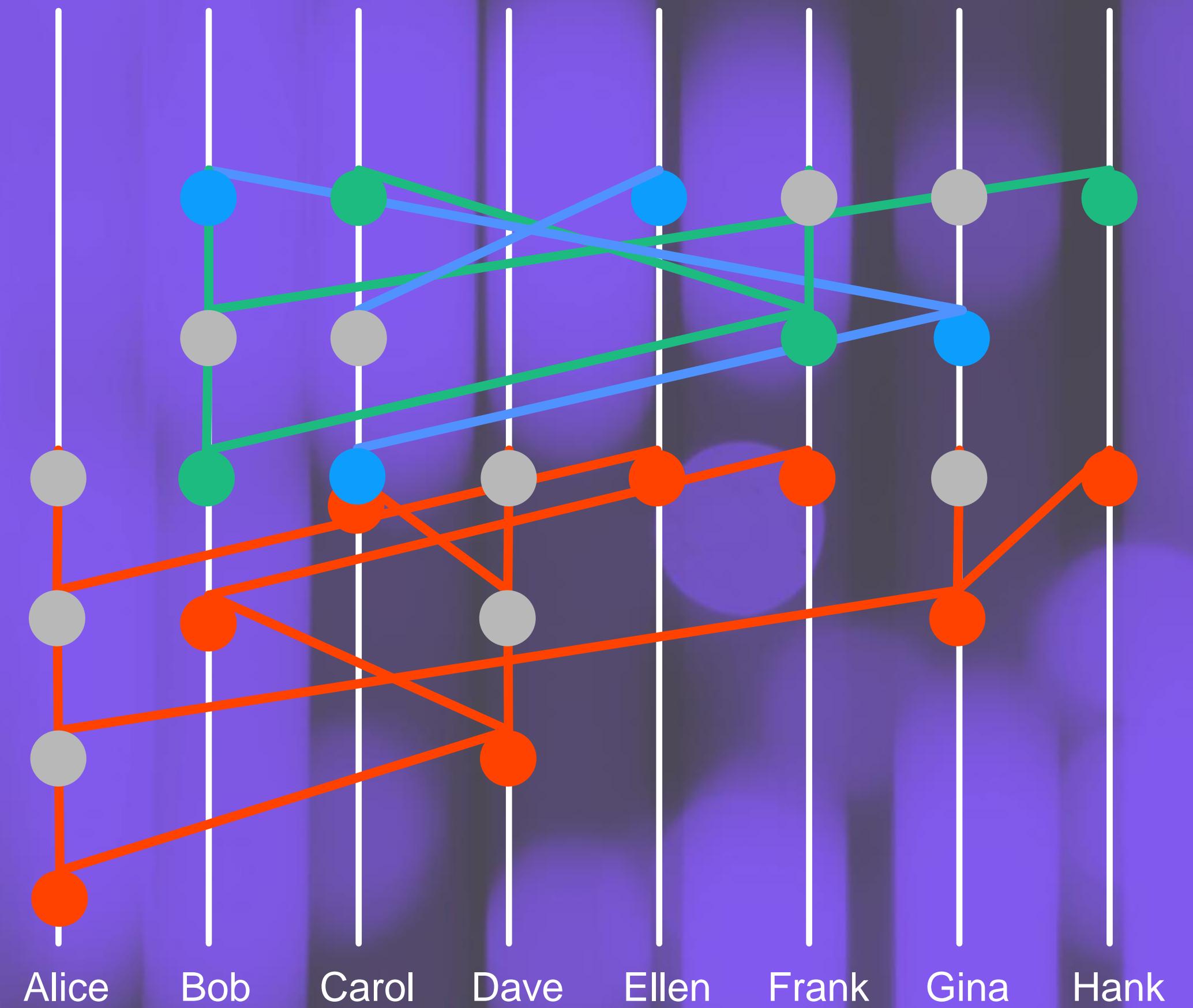
Gossip about Gossip



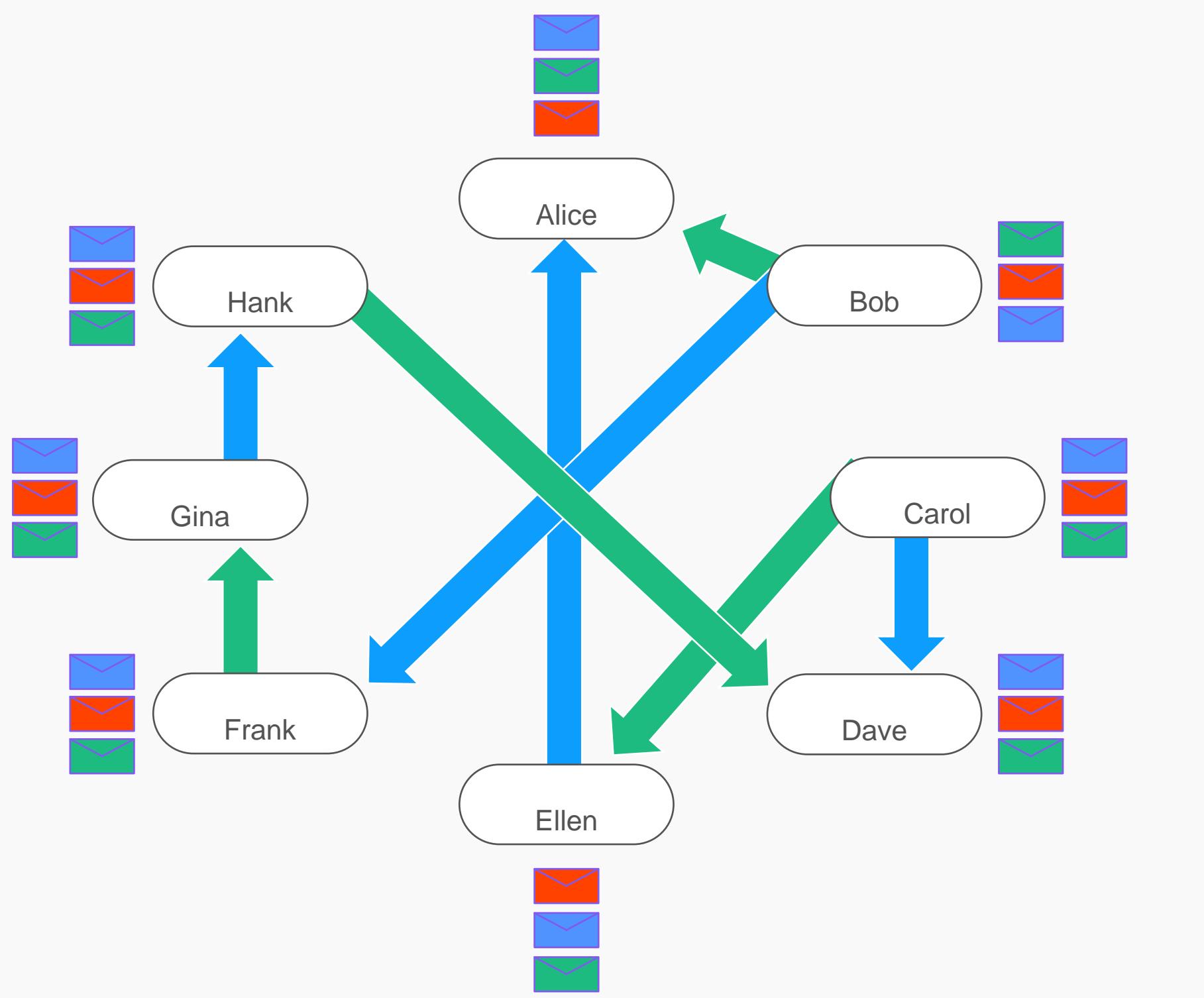
Gossip about gossip



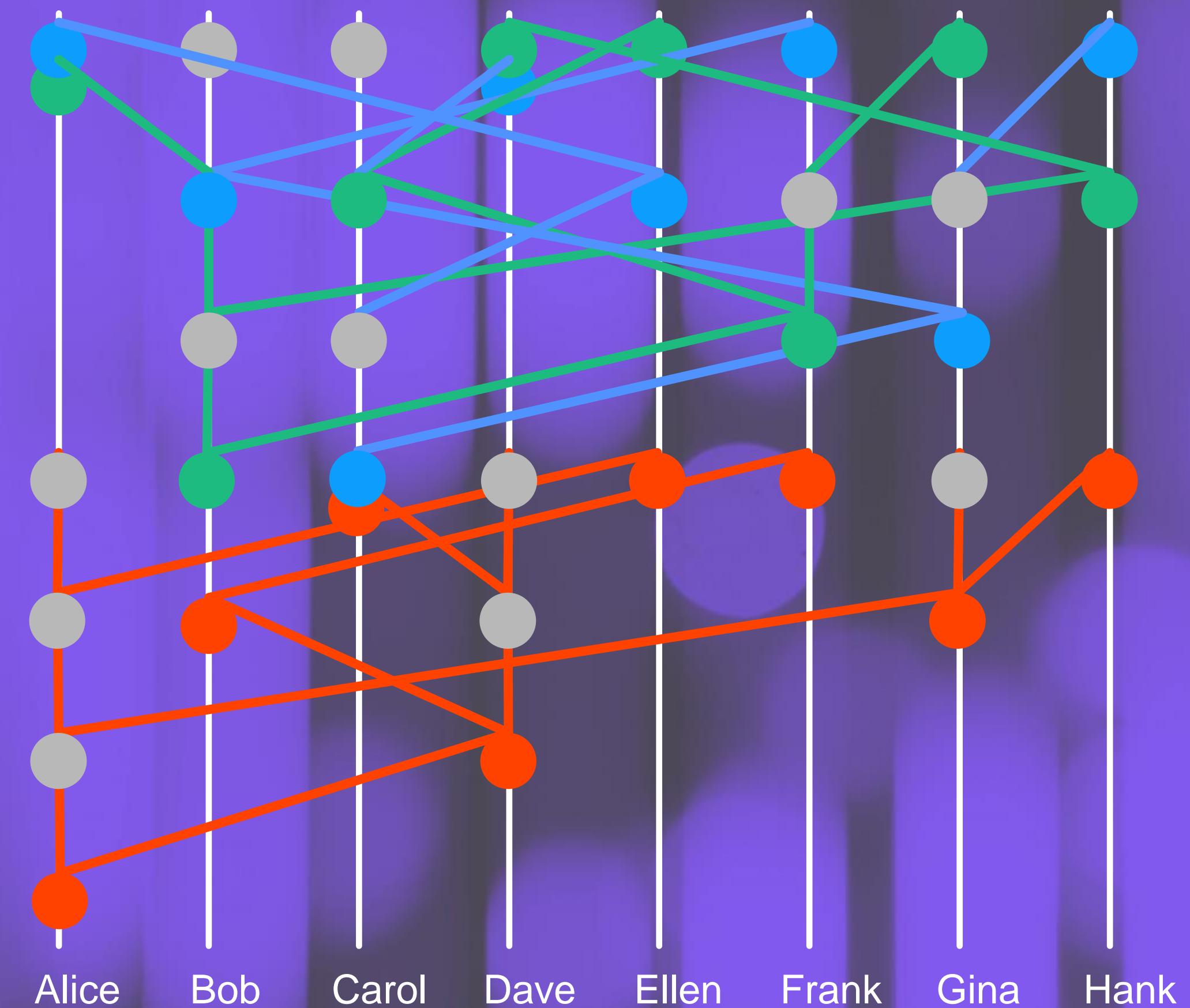
Gossip about Gossip



Gossip about gossip

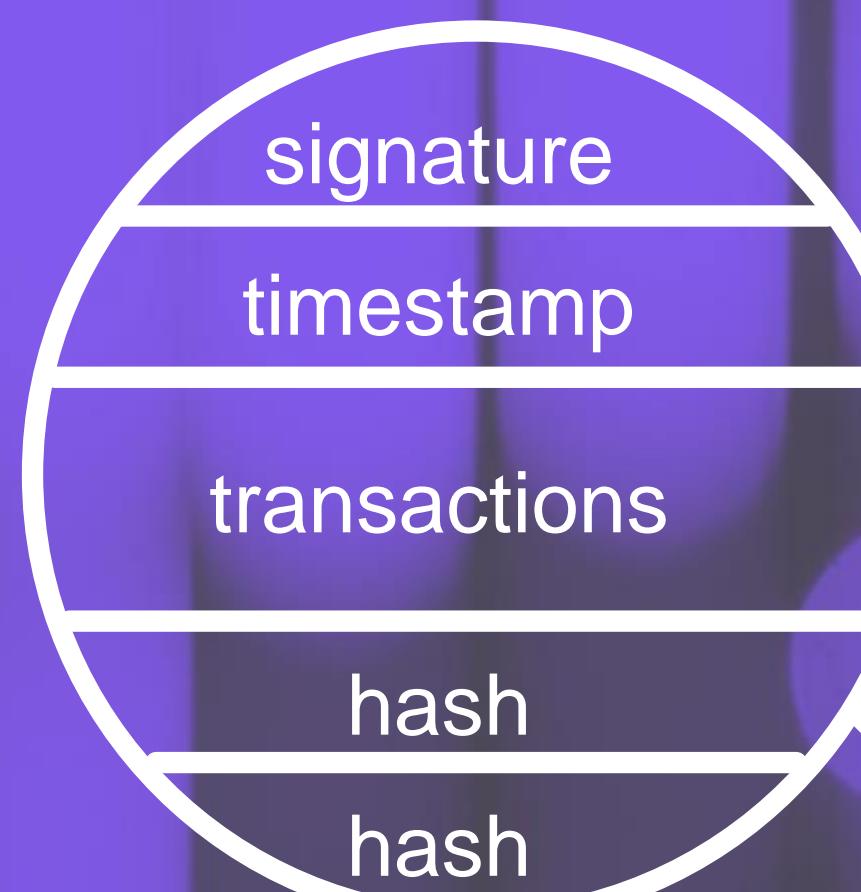


Gossip about Gossip



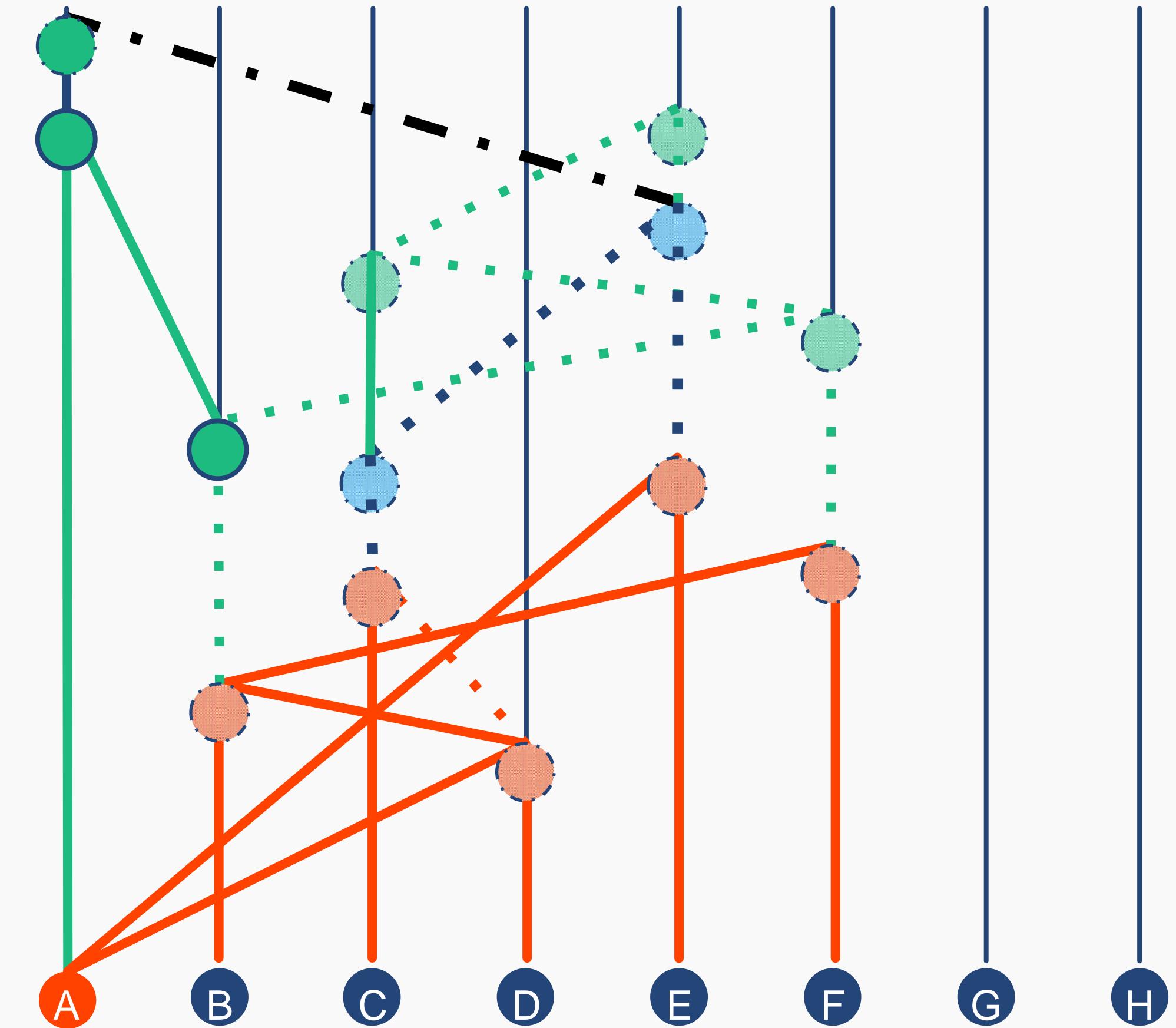


GOSSIP ABOUT GOSSIP



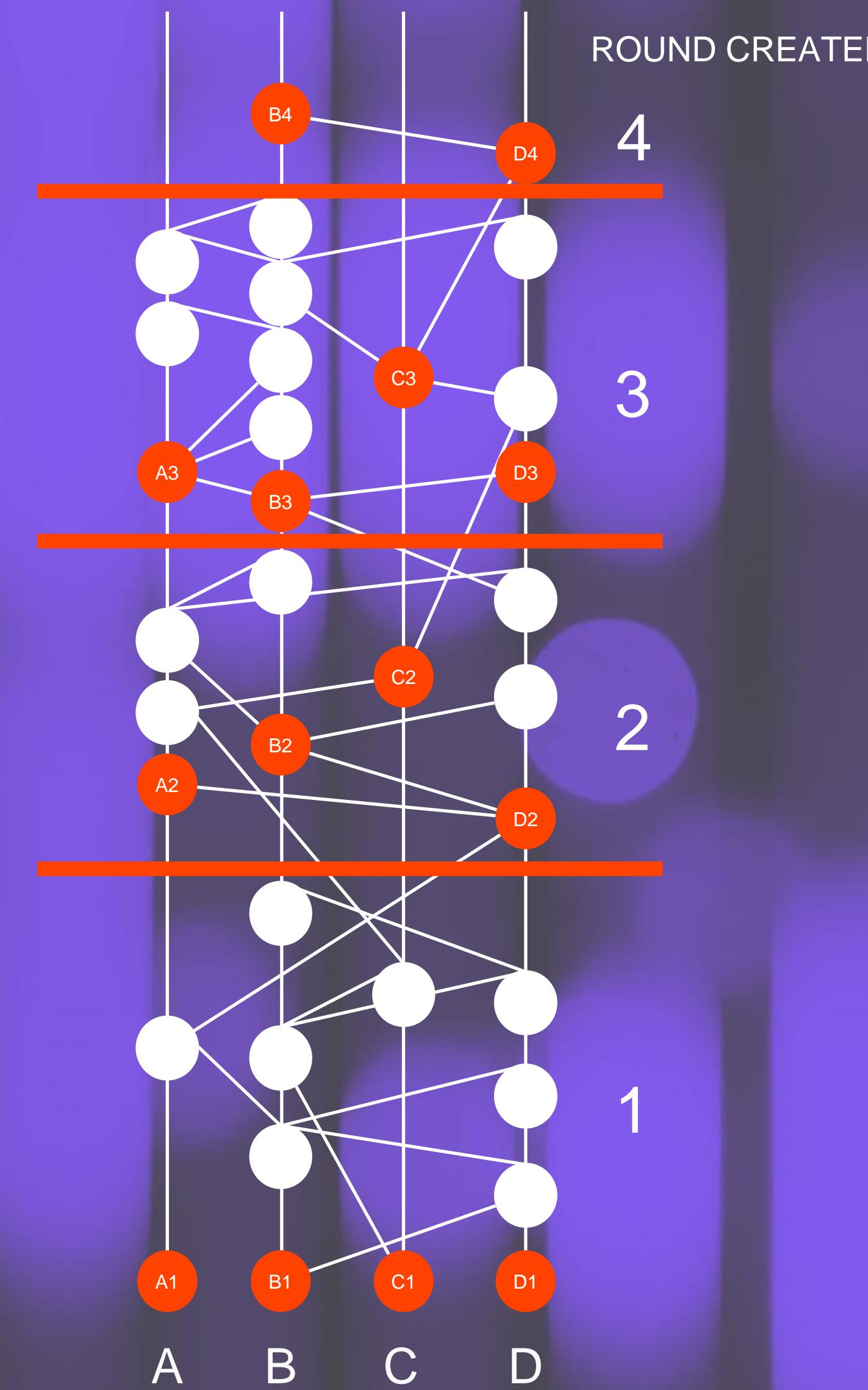
Share events unknown to each other
during gossip

Gossip about Gossip





VIRTUAL VOTING

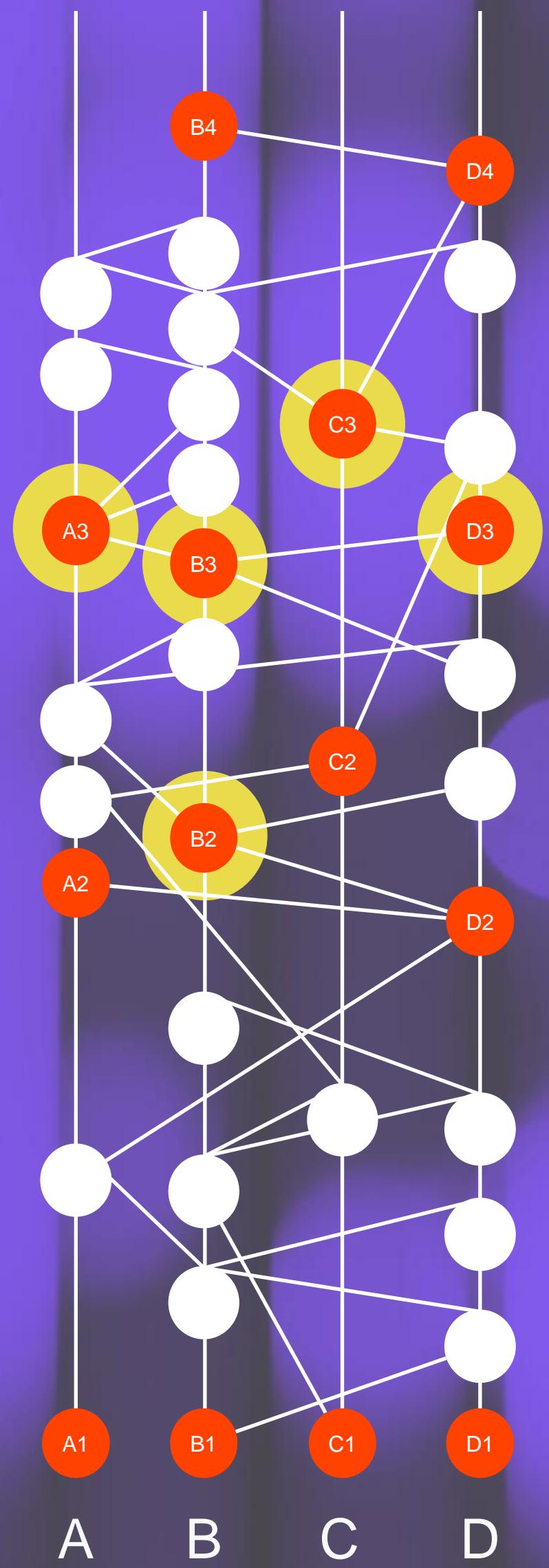


Rounds and witnesses

- Round: Created when the supermajority of witnesses in the previous round can be strongly seen
- Witness: First event in a round for a given node



VIRTUAL VOTING

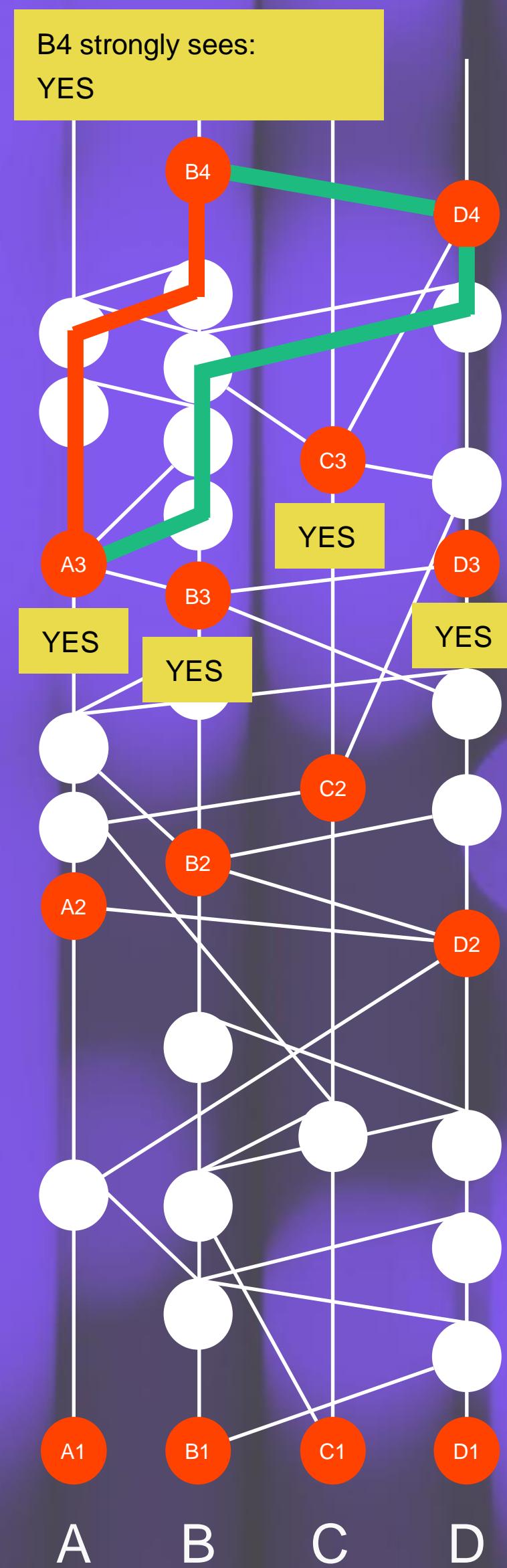


Famous witnesses

- Famous witnesses: means lots of people see it in the next round



VIRTUAL VOTING



Counting votes

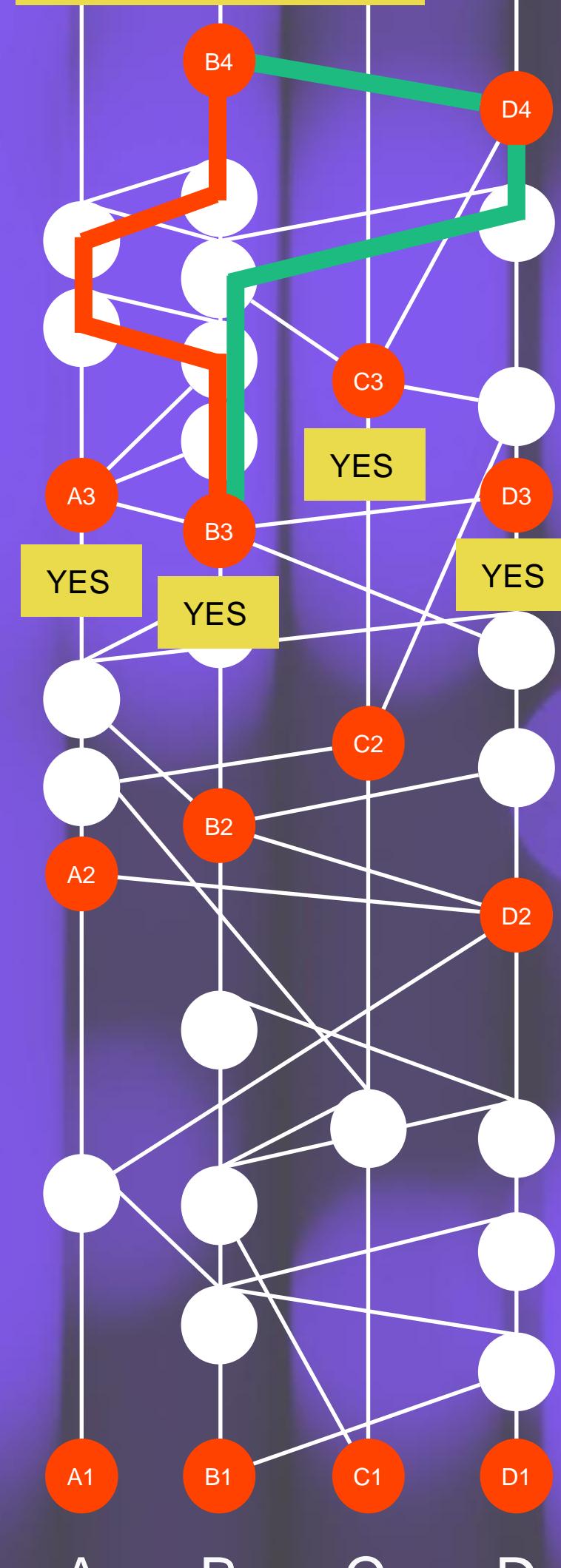
- Strongly seen: To strongly see someone, you have to see them through a supermajority. That means more than two-thirds of the population (or stake as we will see later).





VIRTUAL VOTING

B4 strongly sees:
YES, YES



Counting votes

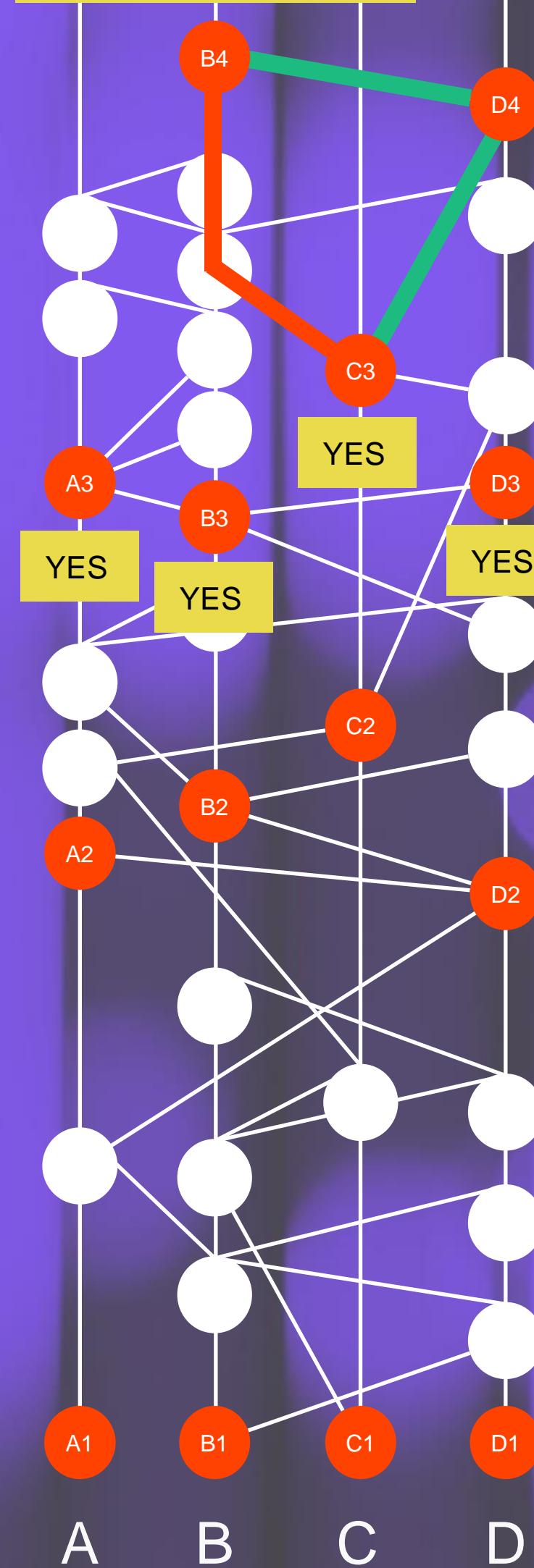
- Strongly seen: To strongly see someone, you have to see them through a supermajority. That means more than two-thirds of the population (or stake as we will see later).





VIRTUAL VOTING

B4 strongly sees:
YES, YES, YES



Counting votes

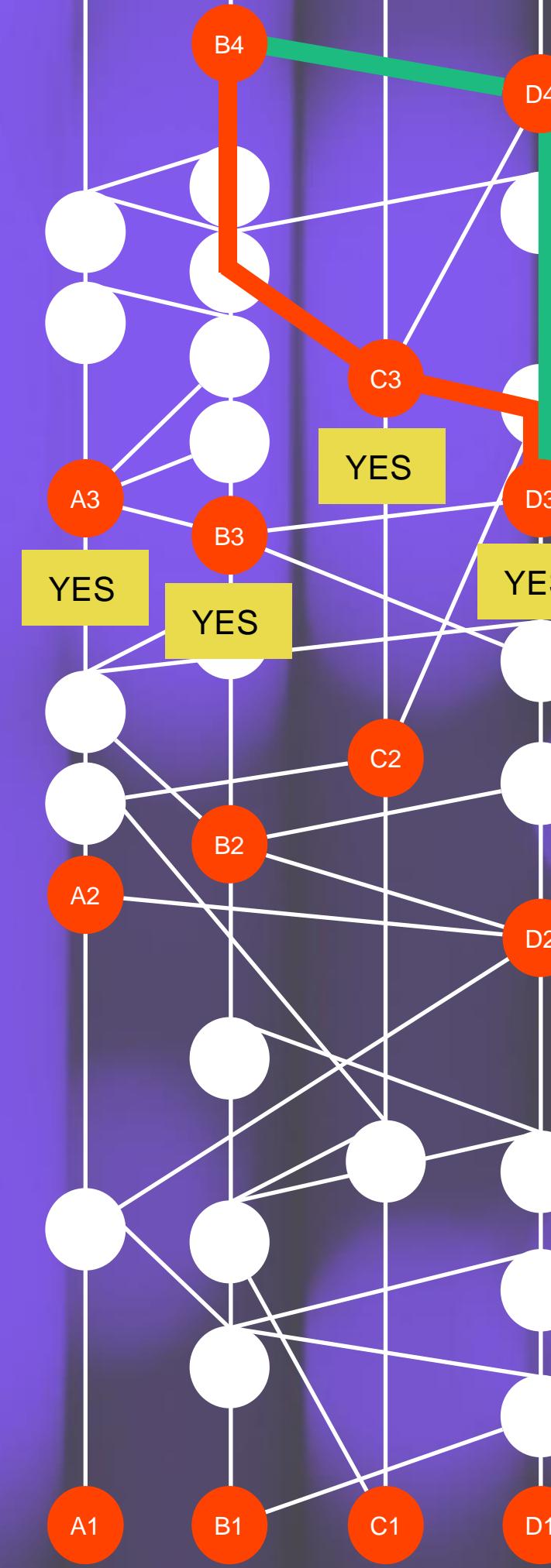
- Strongly seen: To strongly see someone, you have to see them through a supermajority. That means more than two-thirds of the population (or stake as we will see later).





VIRTUAL VOTING

B4 strongly sees:
YES, YES, YES, YES



A B C D

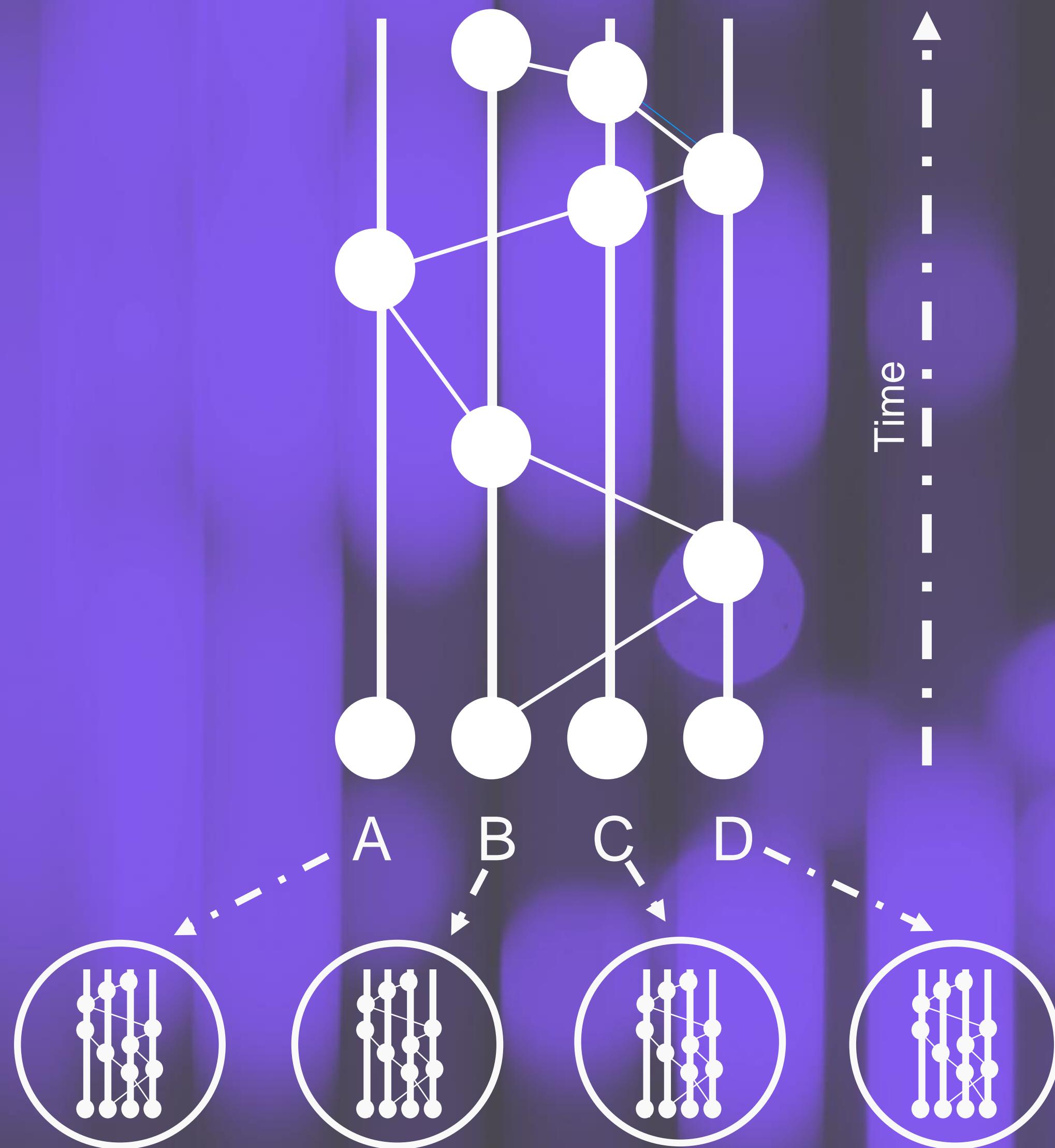


Counting votes

- Strongly seen: To strongly see someone, you have to see them through a supermajority; that means more than two-thirds of hbar staked.

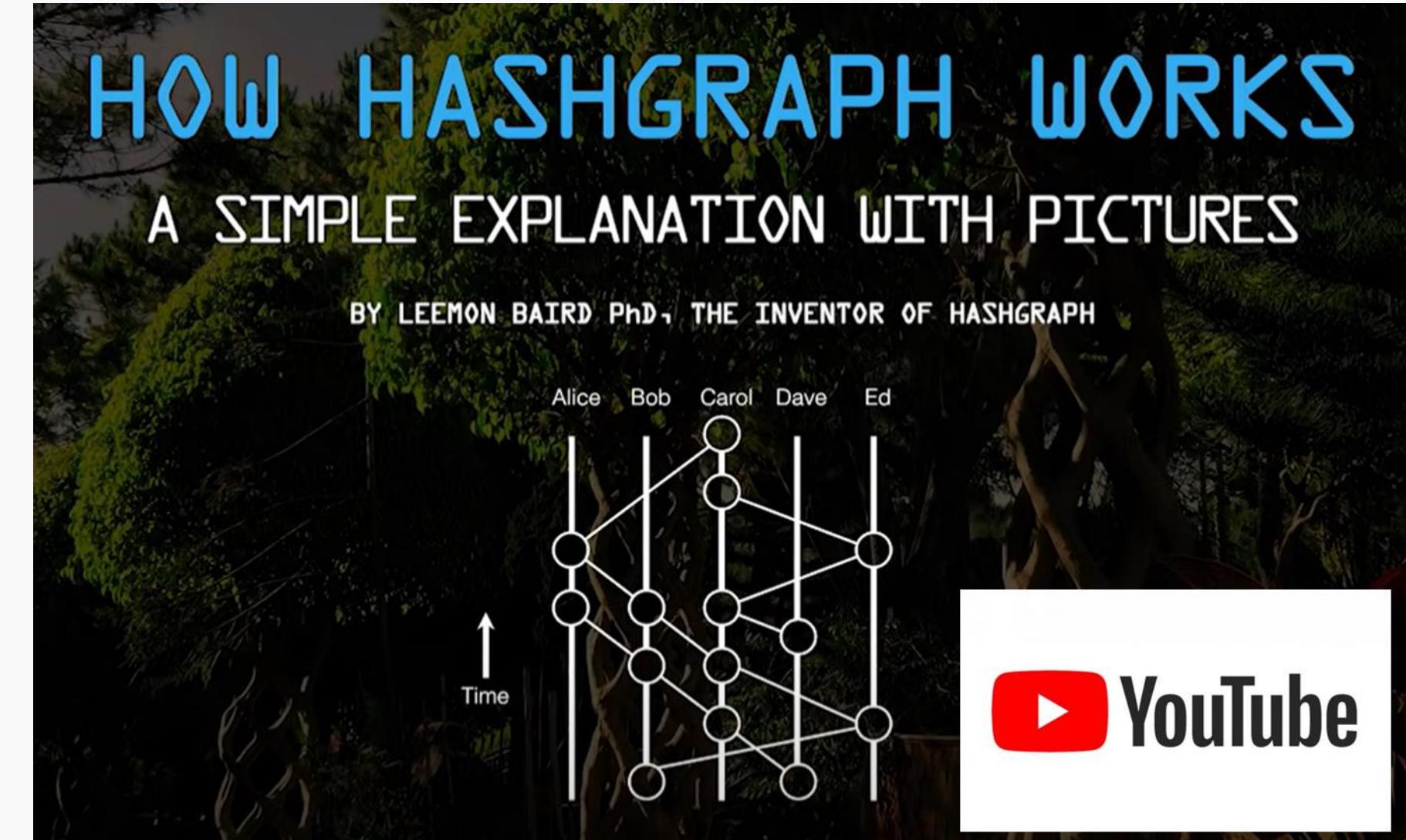


GOSSIP ABOUT GOSSIP



The hashgraph gives each node the entire history of who has talked to who and when

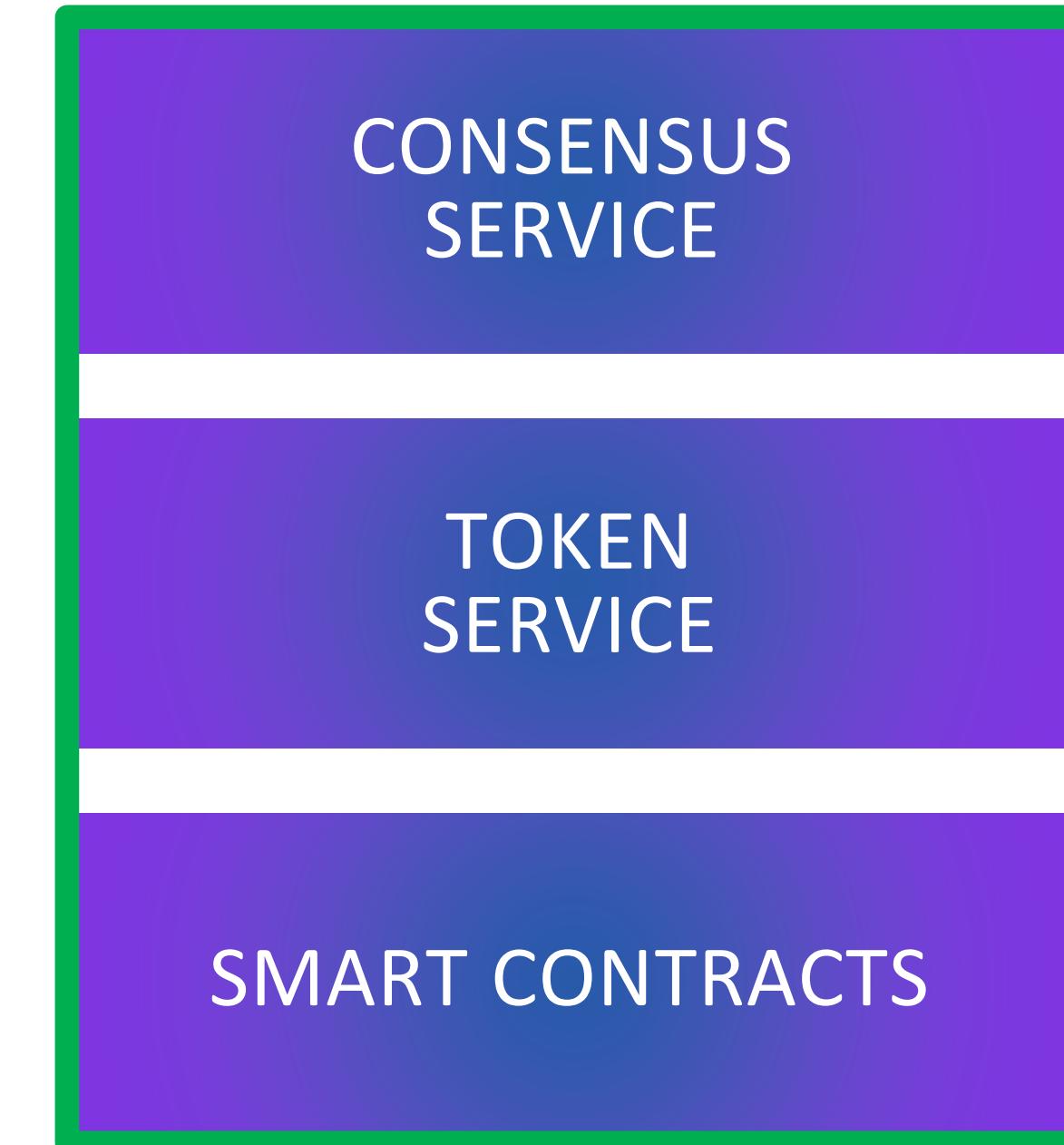
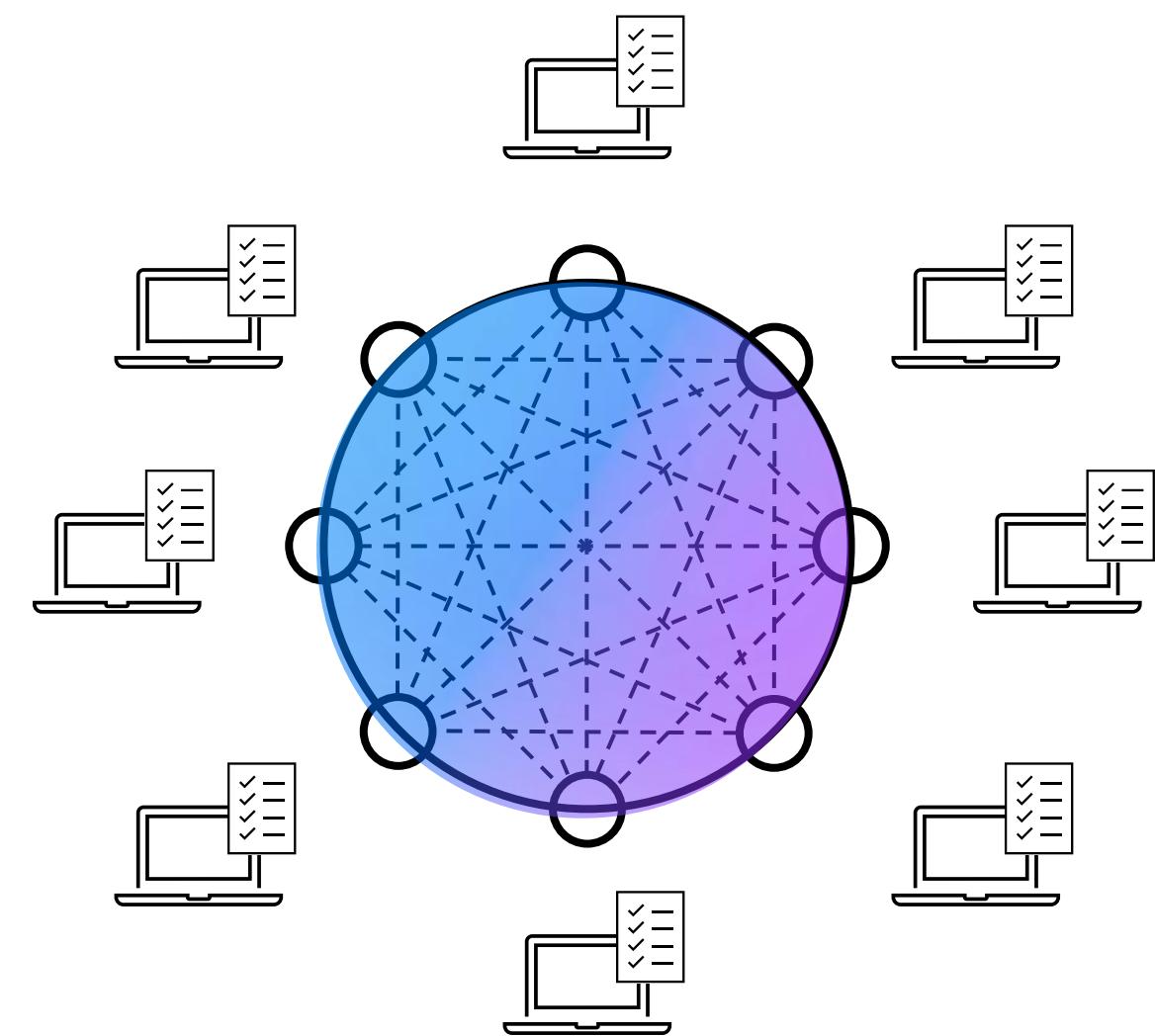
No need to send multiple ballots and receipts over the network as in historical voting models



All nodes have the same picture of the hashgraph, except for the very latest events which will eventually sync



In this session, you will learn how the **Hedera network functions**, how you can **start development**, and **where to learn more**

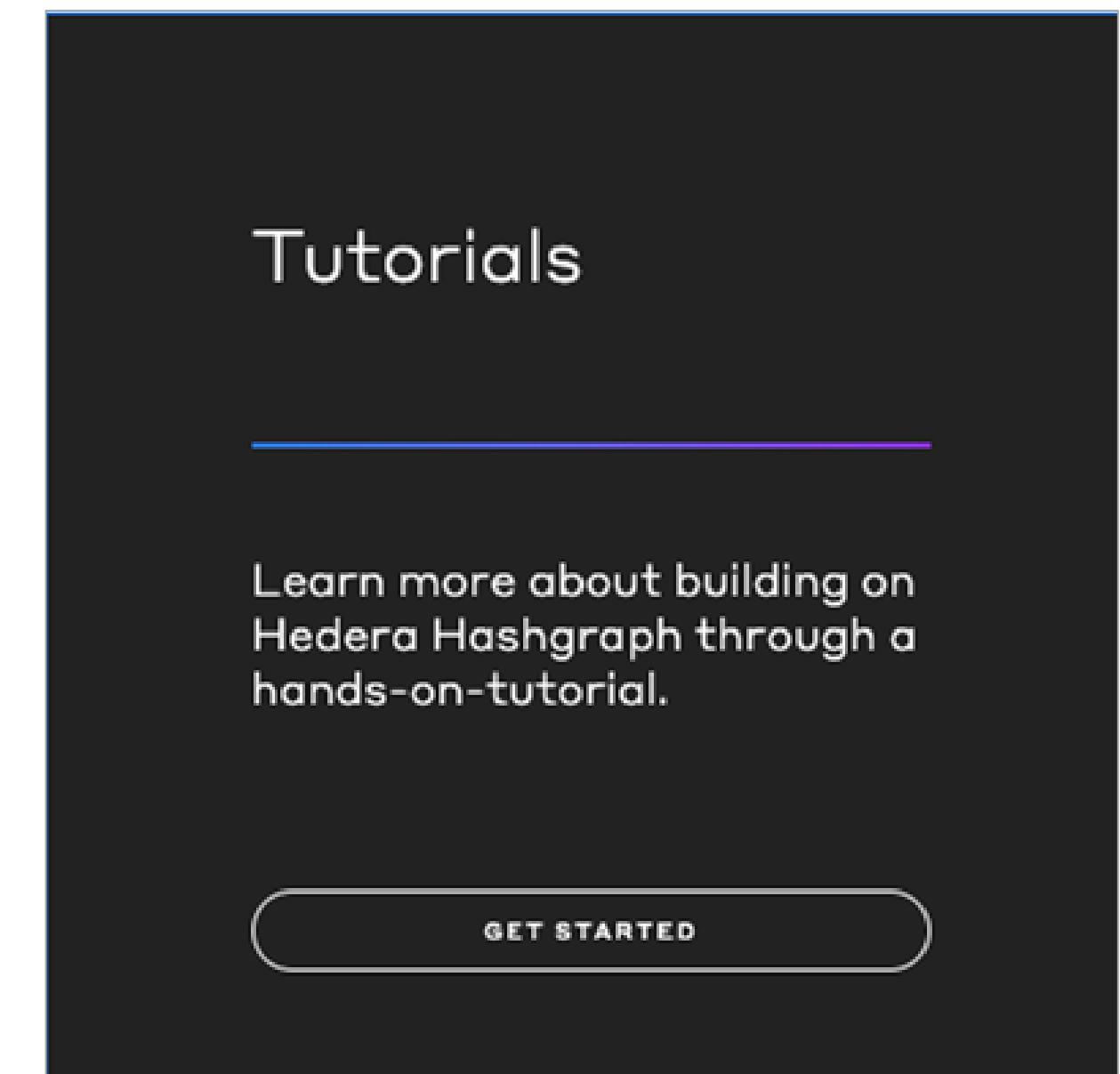


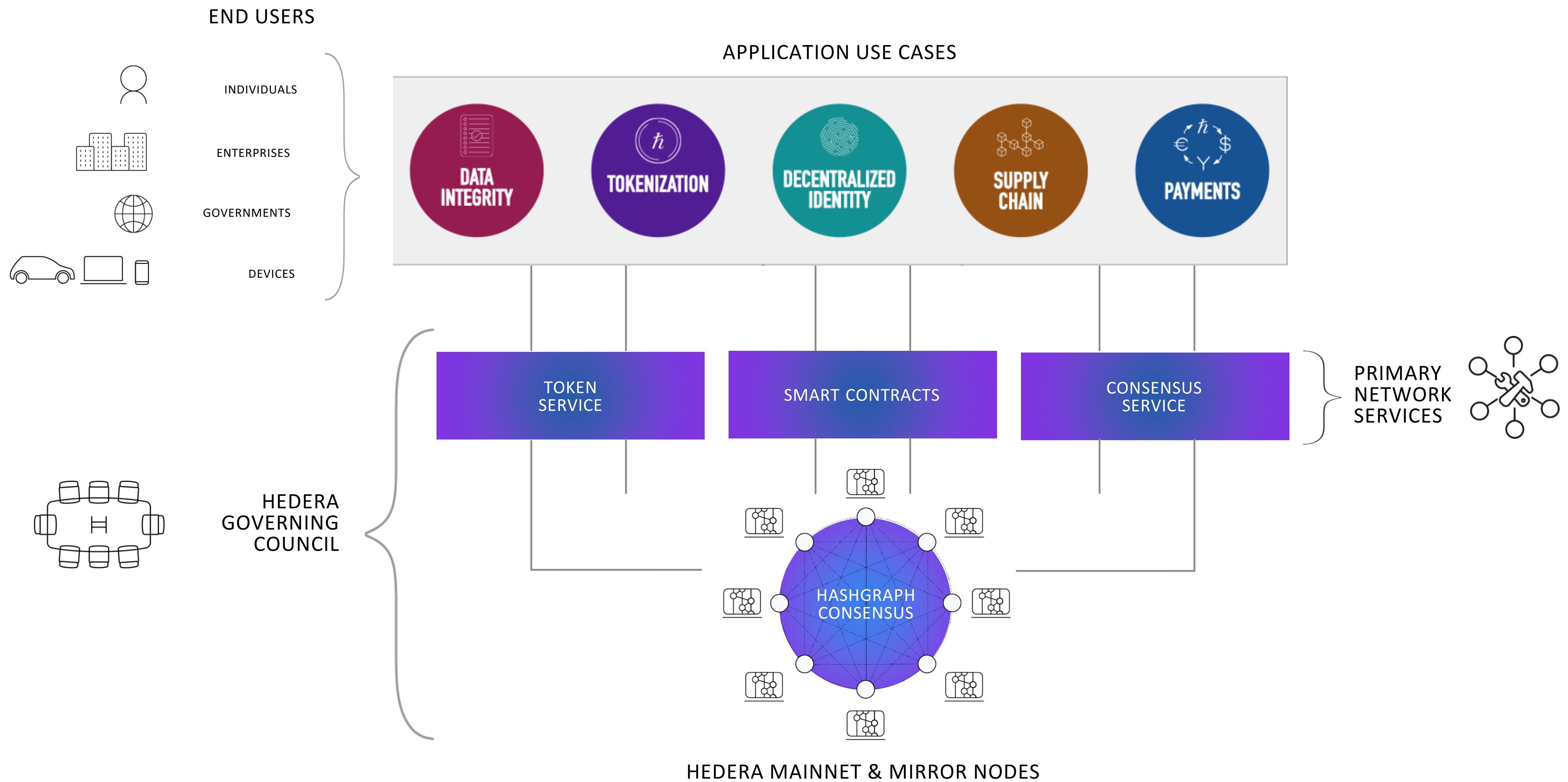
Introduction

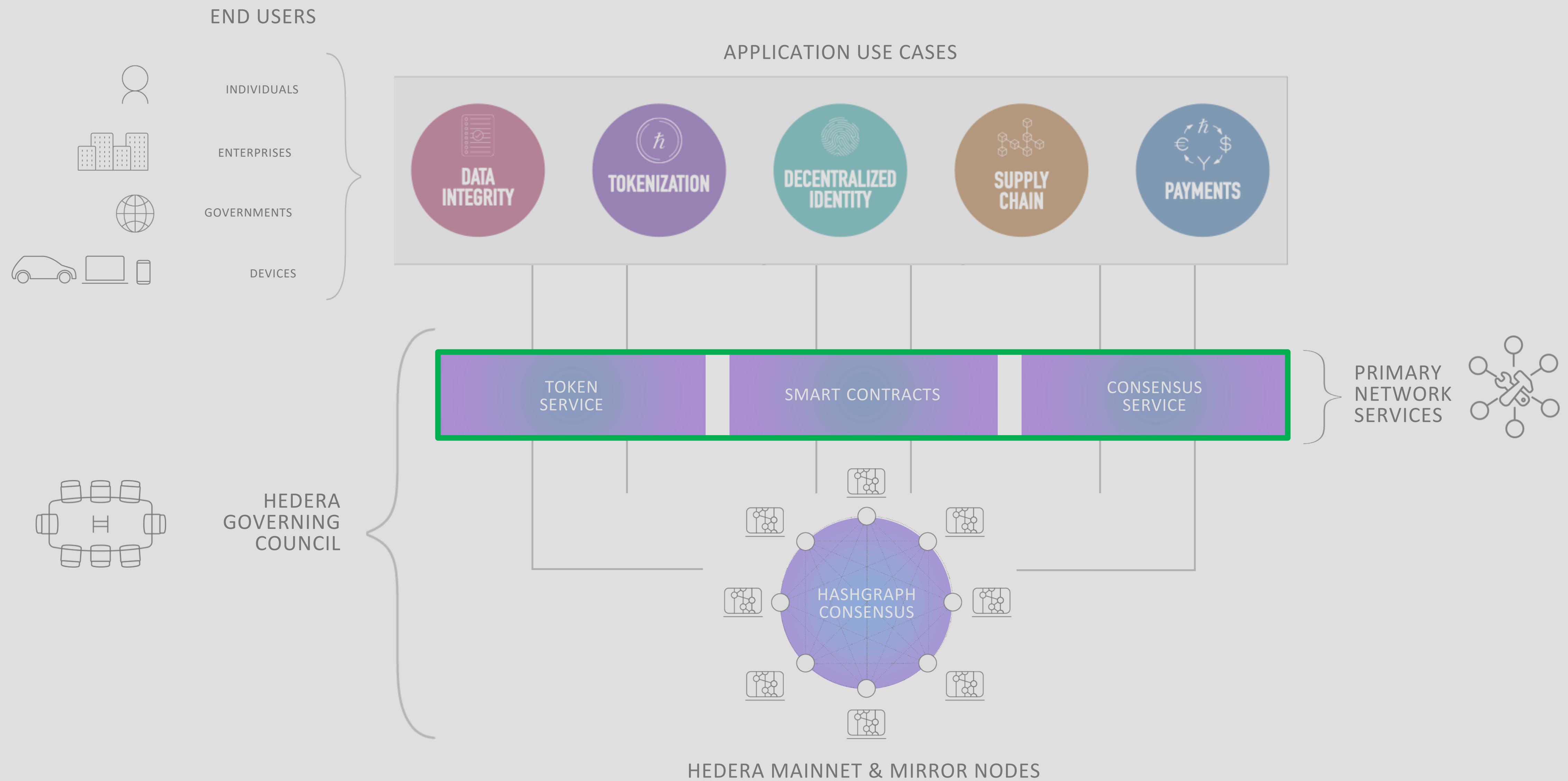
Understand
the Hedera Network

Start Developing on
Hedera

Get Resources and
Learn More







Why build on Hedera?

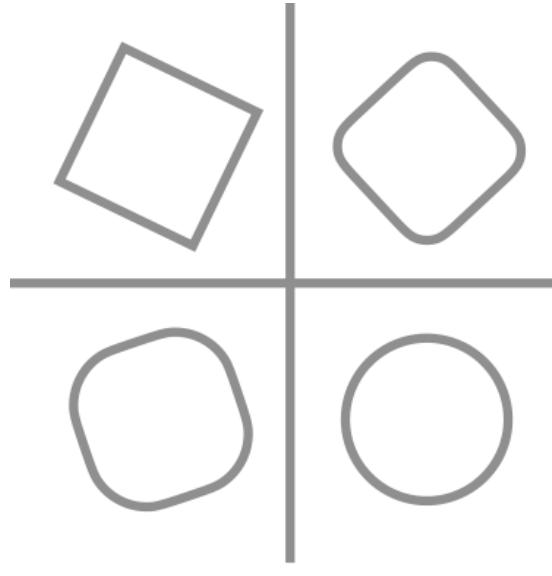
Hedera helps you meet strict requirements in the following areas...



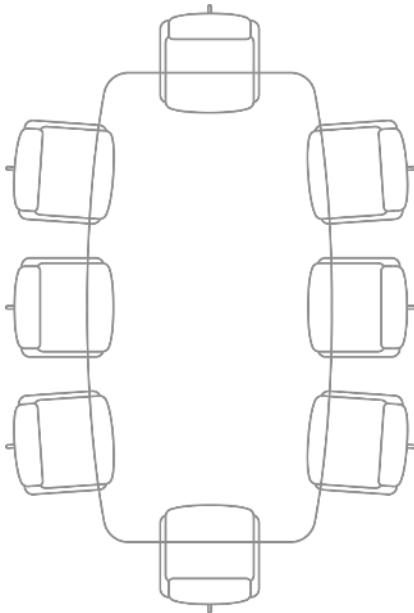
Performance



Security



Stability



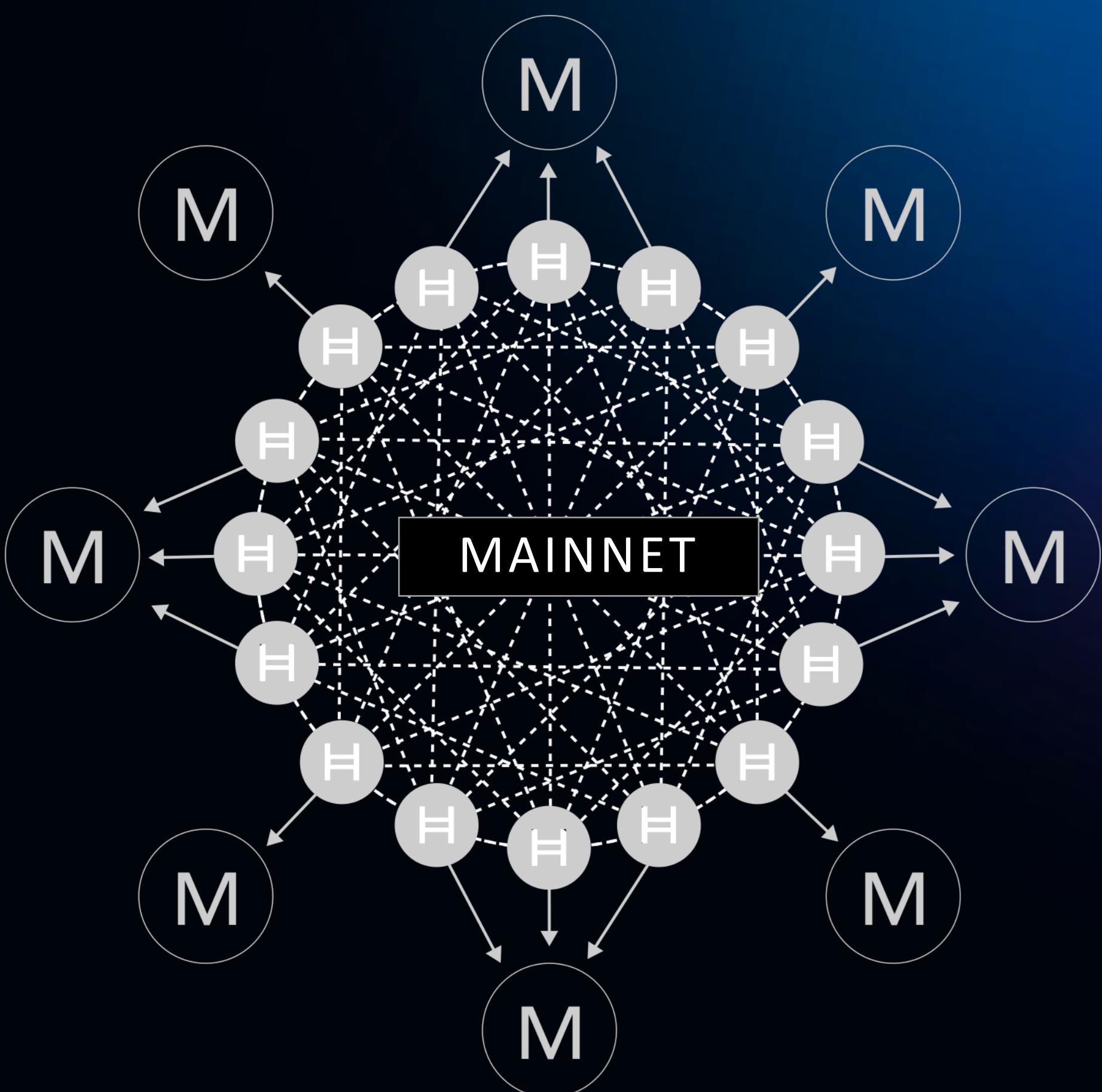
Governance

HEDERA MAINNET & MIRRORNET



MAINNET

- Can submit HAPI (Hedera API) transactions to the Hedera network
- Contributes to consensus on transactions
- Creates events on the Hedera network
- Requires HBAR cryptocurrency payment for transactions & queries



MIRRORNET

- Maintains a history of some or all the Hedera network state and ledger of transactions
- Value-added services (managed read-only node, etc.)
- Enables analytical insight into an application's state / transactions
- Publish and subscribe capabilities



Hedera™ Hashgraph

hedera.com



Cryptocurrency

“Hedera is the only platform we’ve seen that can cope with the volume of split-second transactions that need to take place.”

Jiro Olcott | Director | Power Transition

Scalable transactions

Guaranteed finality in 2-5 seconds

10,000 tps, in a single shard

Used to pay for network fees

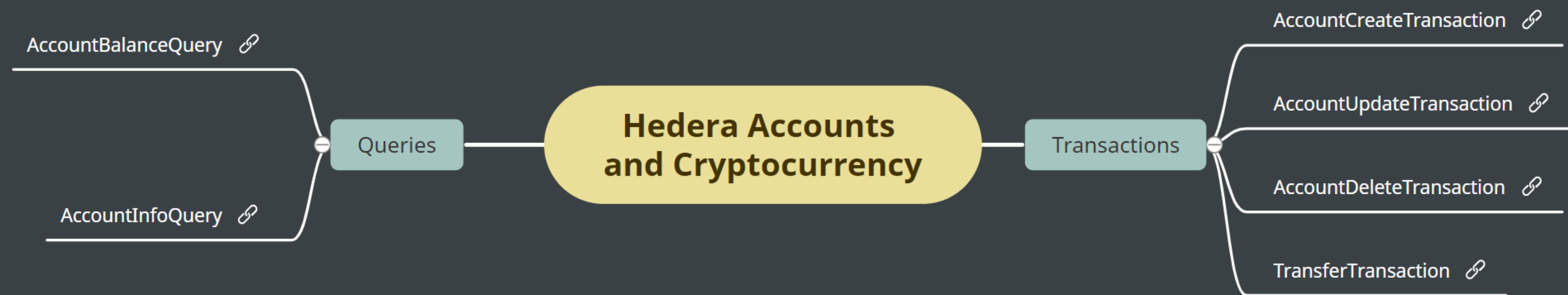
HBAR is Hedera’s native coin

Used to pay for each network API call

Security Mechanism

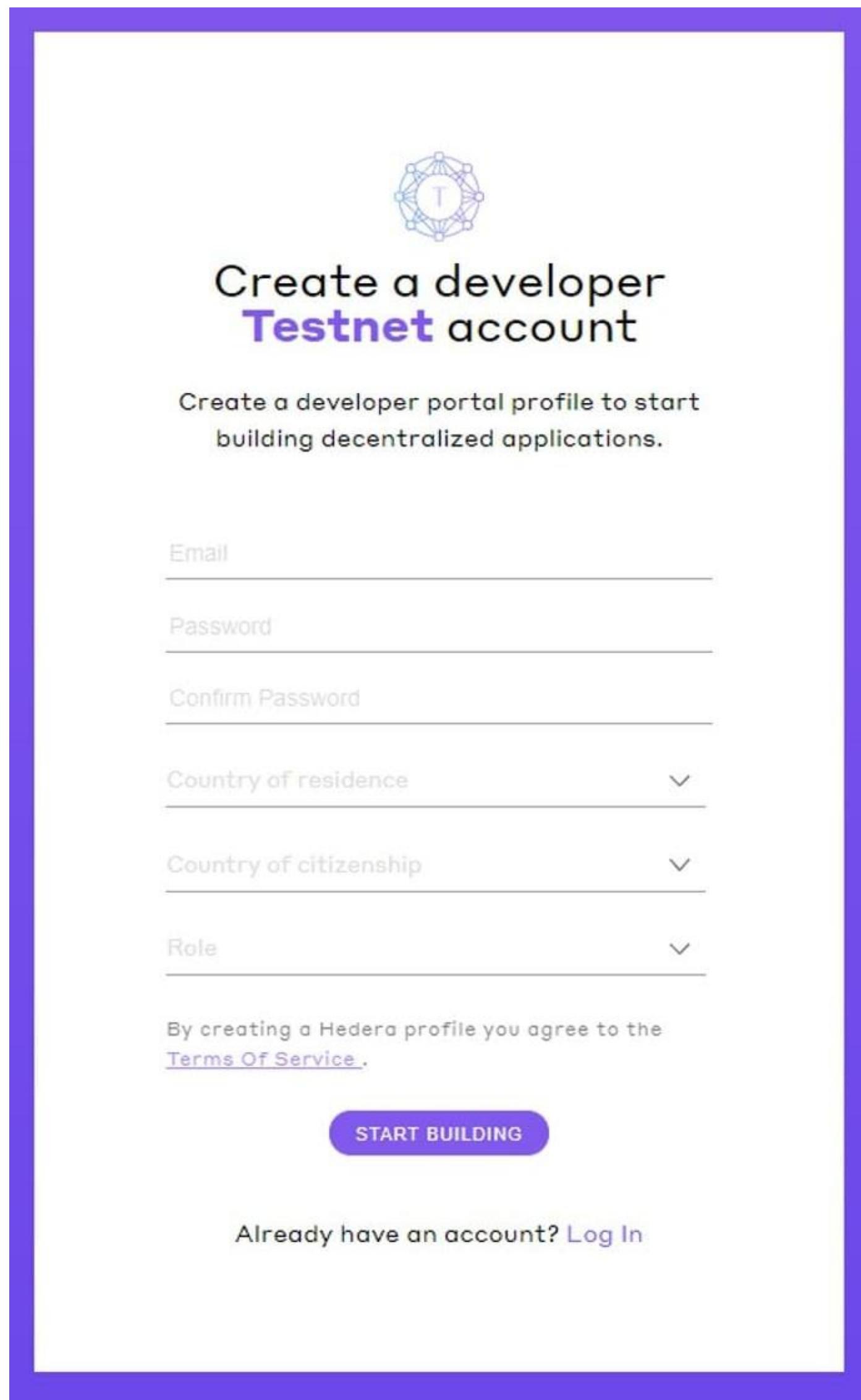
Hedera is a proof of stake network

Malicious actors would need to control 1/3rd of the total HBAR supply in order to launch an attack



Connect to a network

Get a testnet account! (<https://portal.hedera.com/register>)



Create a developer **Testnet** account

Create a developer portal profile to start building decentralized applications.

Email

Password

Confirm Password

Country of residence

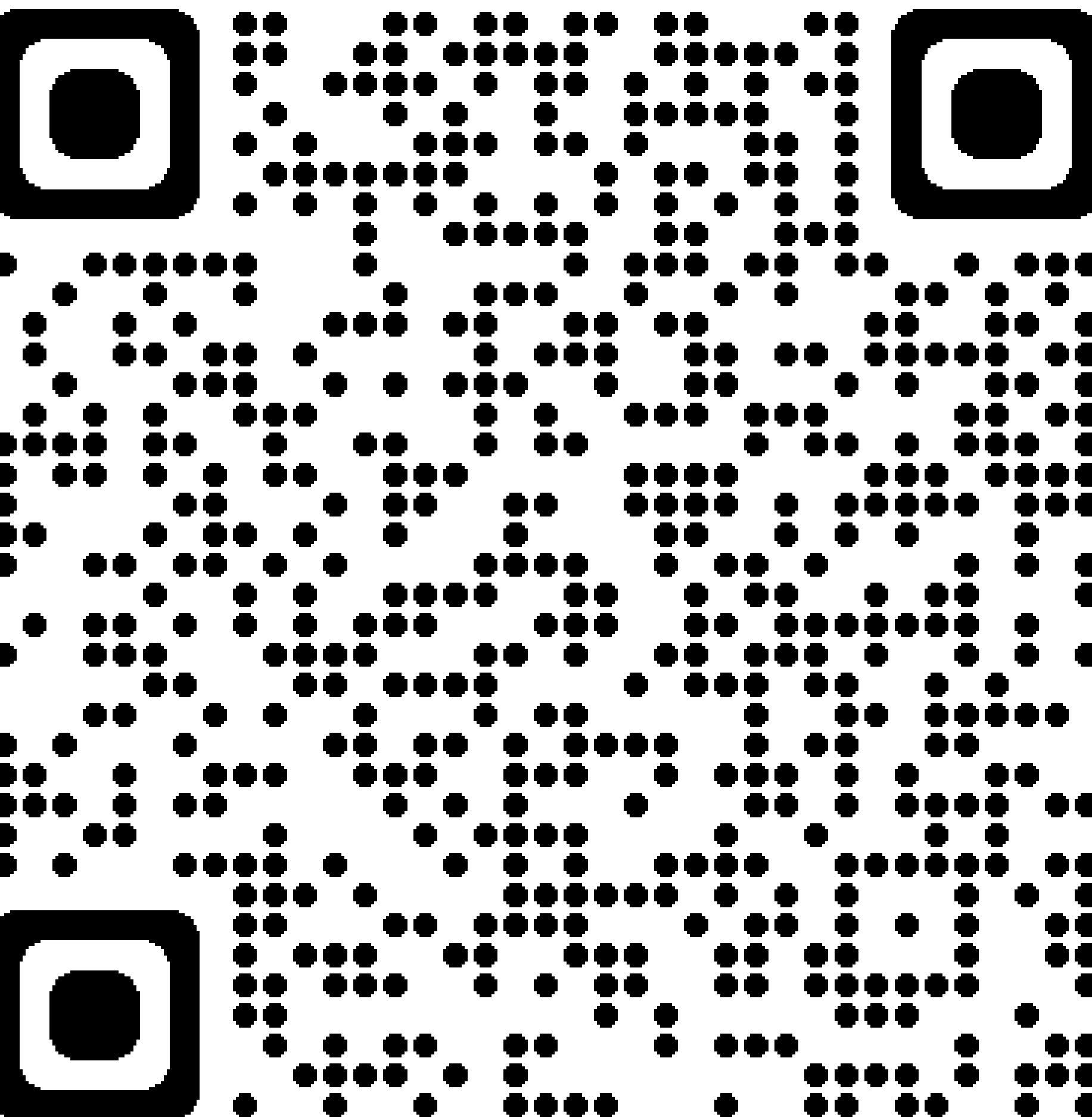
Country of citizenship

Role

By creating a Hedera profile you agree to the [Terms Of Service](#).

START BUILDING

Already have an account? [Log In](#)



Update .env file with account details

```
#TESTNET CREDENTIALS
```

```
OPERATOR_ID = 0.0.17994793
```

```
OPERATOR_PBKEY = 302a[REDACTED]
```

```
OPERATOR_PVKEY = 302e[REDACTED]
```

```
TREASURY_ID = 0.0.17994793
```

```
TREASURY_PBKEY = 302a[REDACTED]
```

```
TREASURY_PVKEY = 302e[REDACTED]
```

```
ALICE_ID = 0.0.17994793
```

```
ALICE_PBKEY = 302a[REDACTED]
```

```
ALICE_PVKEY = 302e[REDACTED]
```

portal.hedera.com

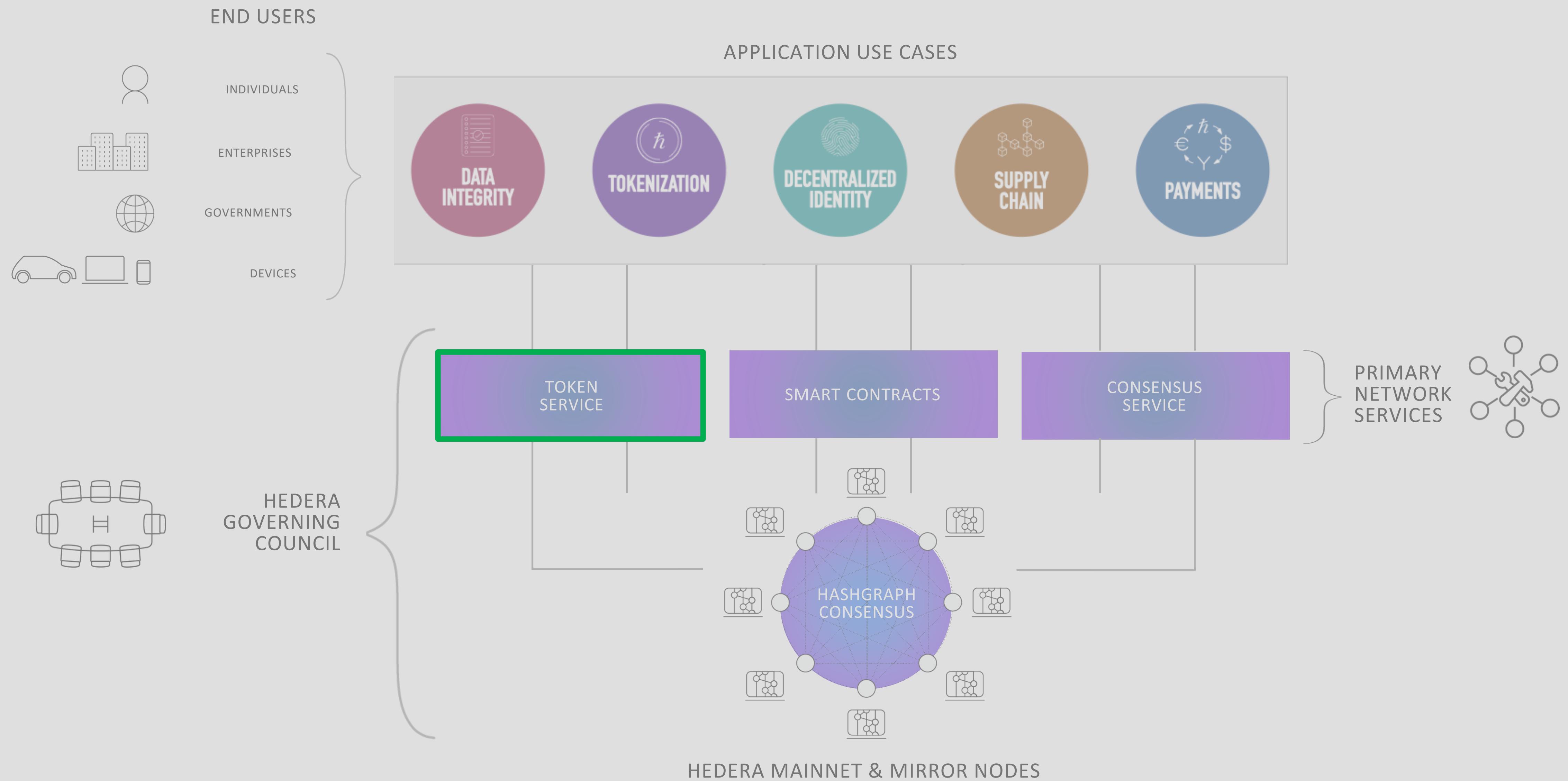
OR

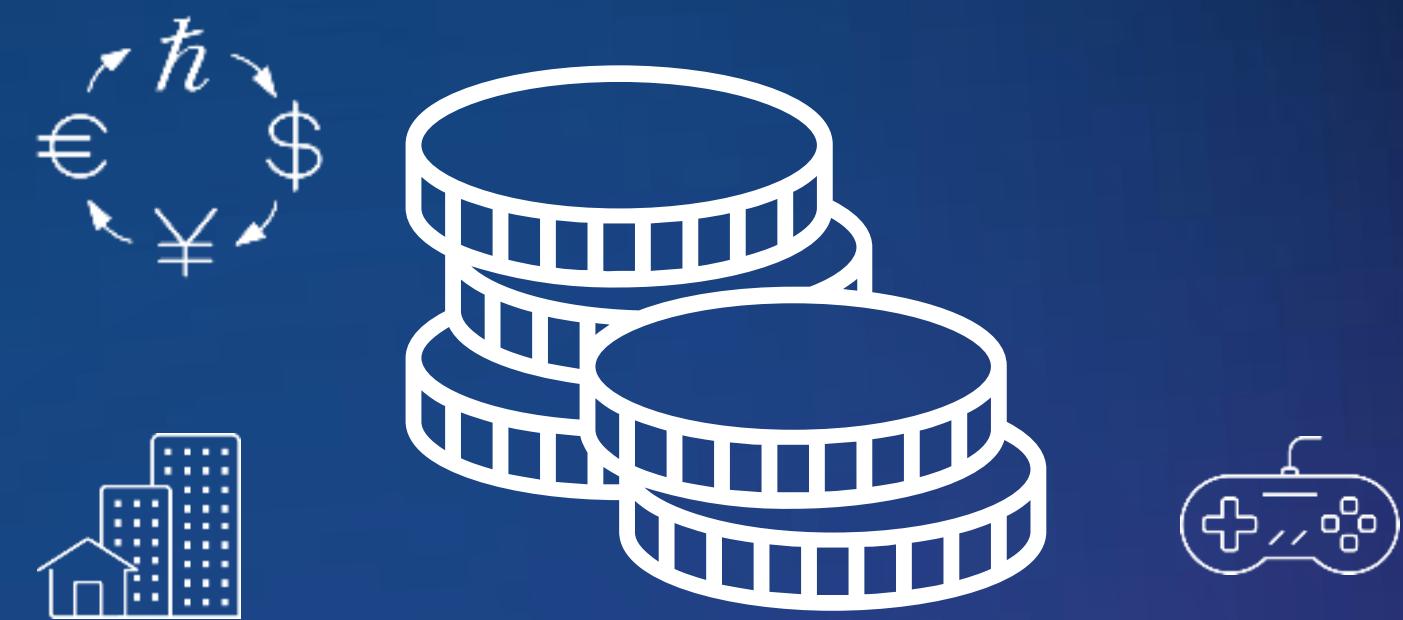
docs.hedera.com/docs/mainnet

Configure accounts and client

```
// Configure accounts and client in your main function
const operatorId = AccountId.fromString(process.env.OPERATOR_ID);
const operatorKey = PrivateKey.fromString(process.env.OPERATOR_PKEY);

const client = Client.forTestnet().setOperator(operatorId, operatorKey);
```





Token Service

“There are a number of challenges in using traditional ordering services for decentralized applications. By exposing this capability, Hedera is making a meaningful contribution to furthering the capabilities of distributed networks.”

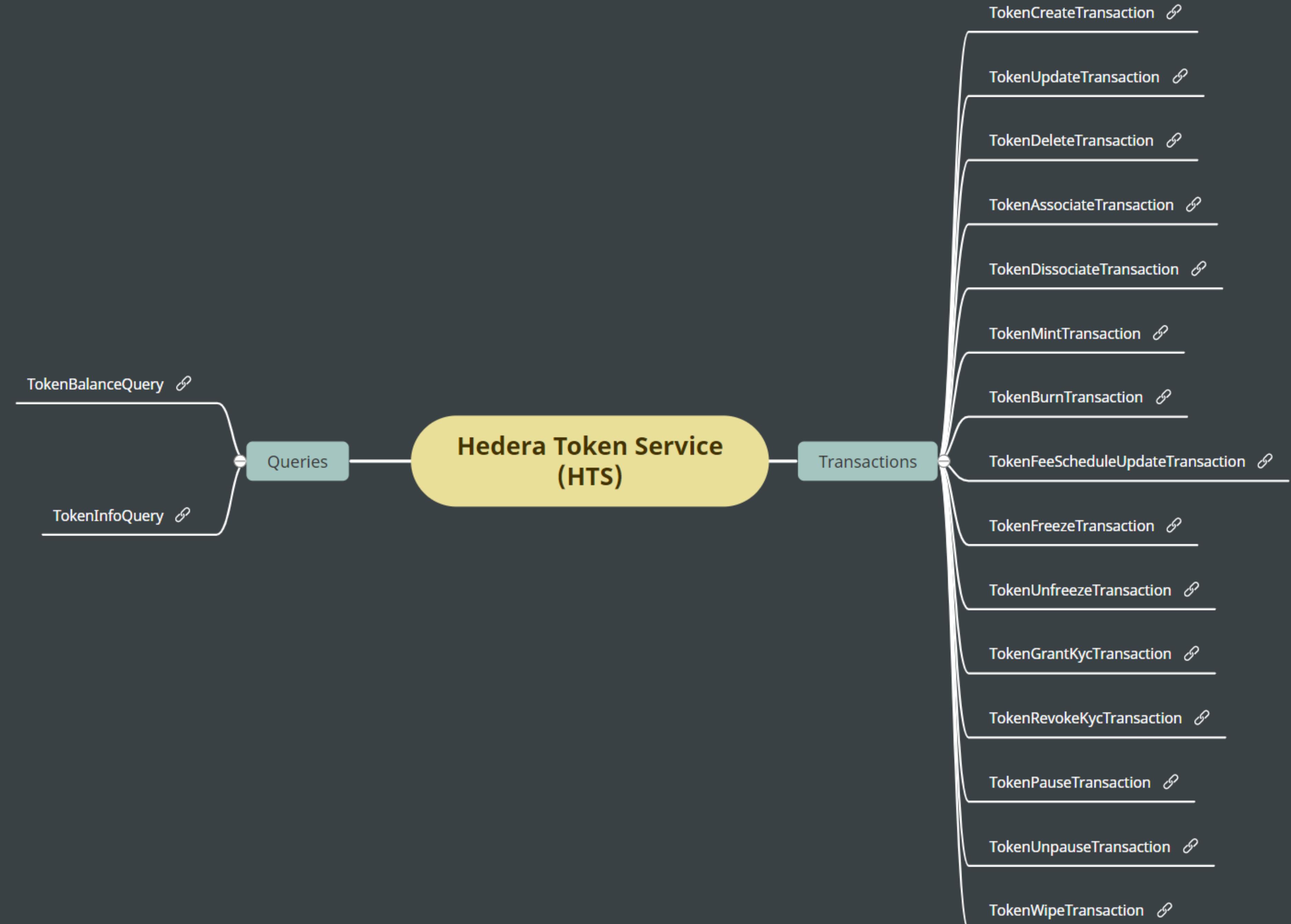
Bryan Gross | Product Manager | IBM Blockchain

Tokenization made simple

Mint and manage fungible and non-fungible tokens without needing to deploy a smart contract

Tokenize natively

High-throughput, compliance configurations, and on-chain programmability



TokenCreateTransaction() | Simple Non-fungible Token (NFT)

```
const { Client, TokenCreateTransaction, TokenType, TokenSupplyType } = require("@hashgraph/sdk");
client = Client.forTestnet().setOperator(operatorId, operatorKey);

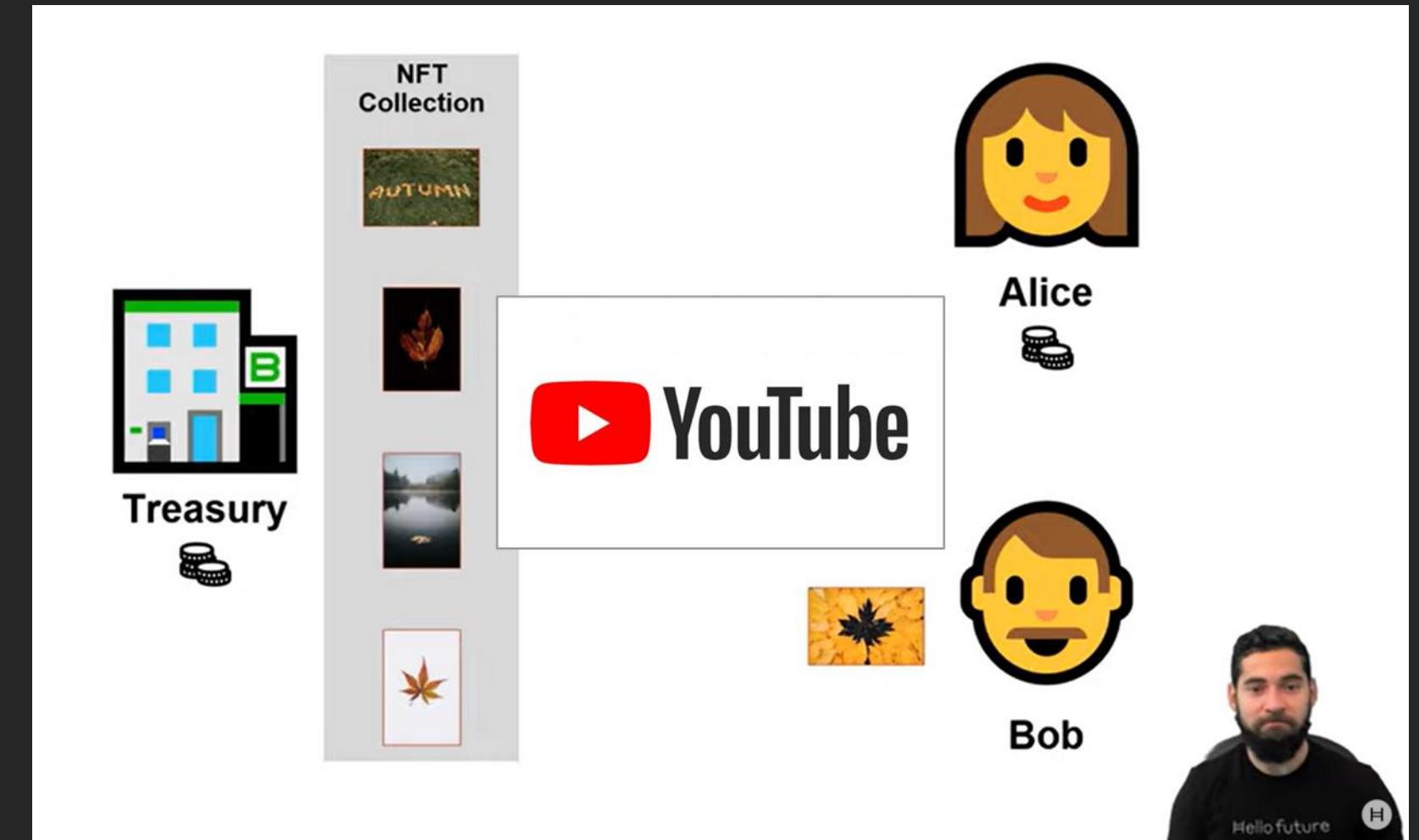
async function main() {

  const nftCreate = await new TokenCreateTransaction()
    .setTokenName("Fall Collection")
    .setTokenSymbol("LEAF")
    .setTokenType(TokenType.NonFungibleUnique)
    .setInitialSupply(0)
    .setTreasuryAccountId(treasuryId)
    .setMaxSupply(10)
    .setSupplyKey(supplyKey)
    .freezeWith(client);

  const nftCreateTxSign = await nftCreate.sign(treasuryKey);
  const nftCreateSubmit = await nftCreateTxSign.execute(client);
  const nftCreateRx = await nftCreateSubmit.getReceipt(client);
  const tokenId = nftCreateRx tokenId;
  console.log(`Created NFT with Token ID: ${tokenId}\n`);

}

main();
```



TokenCreateTransaction() | Customized Non-fungible Token (NFT)

```
const { Client, TokenCreateTransaction, TokenType, TokenSupplyType } = require("@hashgraph/sdk");
client = Client.forTestnet().setOperator(operatorId, operatorKey);

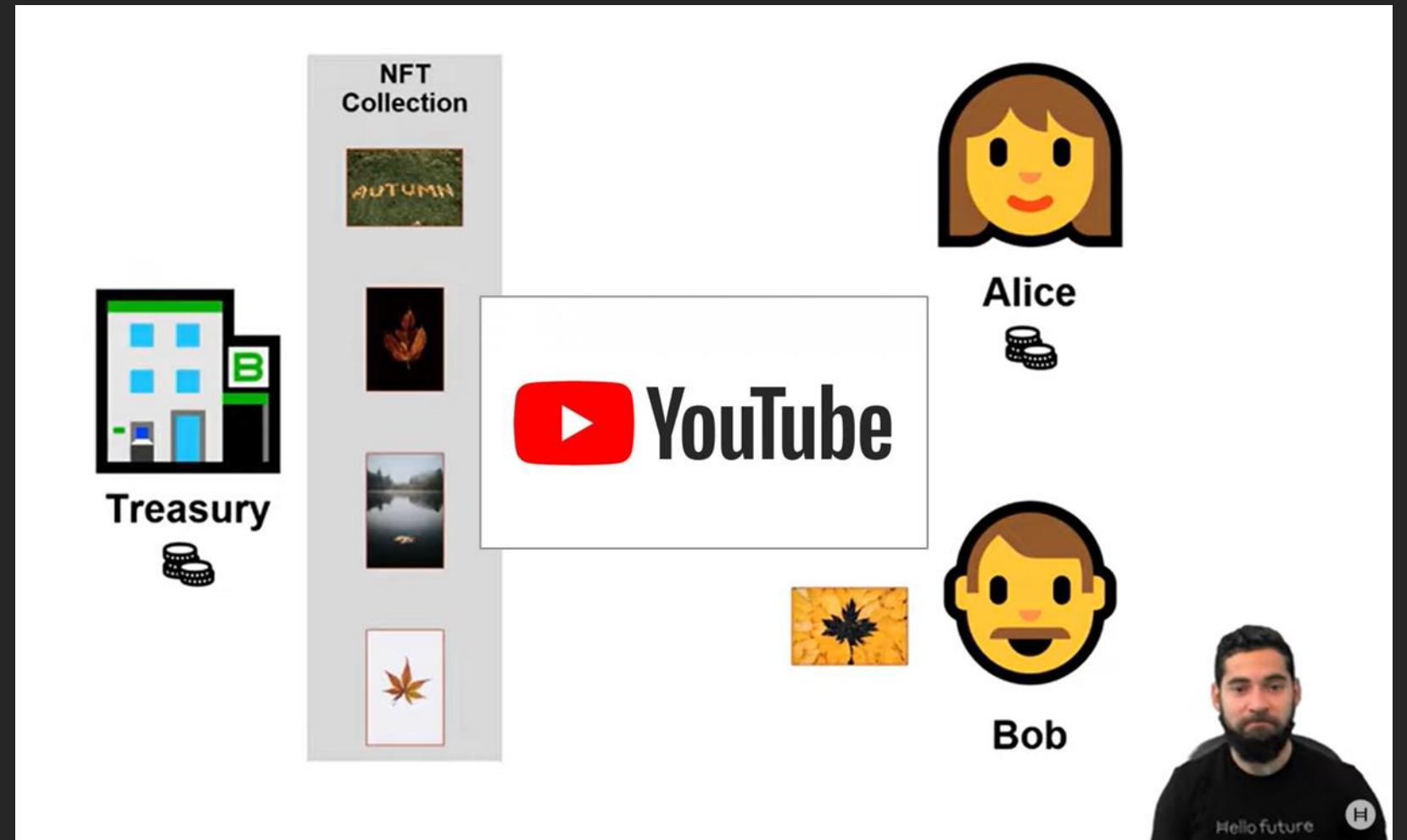
async function main() {

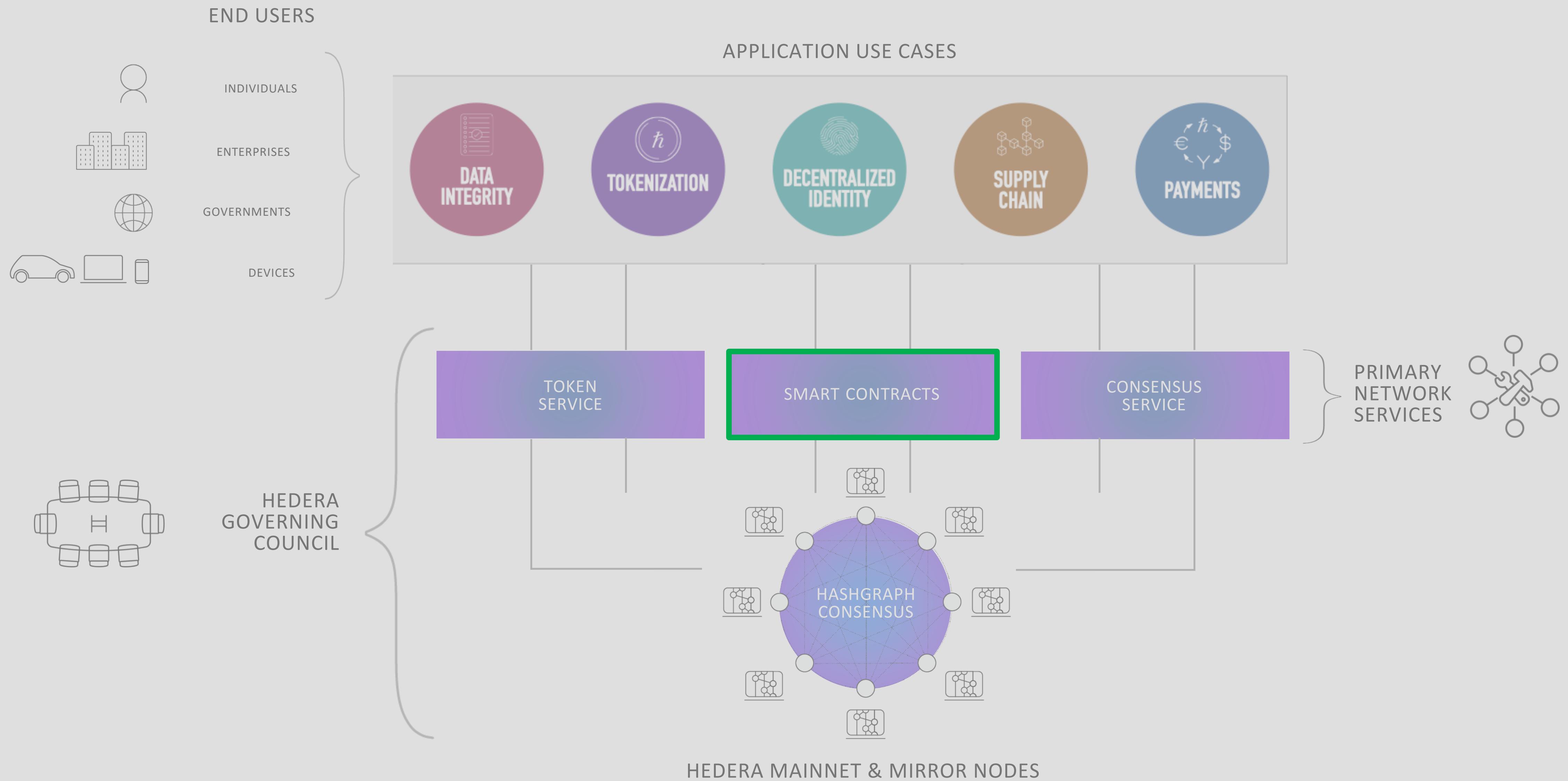
  const nftCreate = await new TokenCreateTransaction()
    .setTokenName("Fall Collection")
    .setTokenSymbol("LEAF")
    .setTokenType(TokenType.NonFungibleUnique)
    .setDecimals(0)
    .setInitialSupply(0)
    .setTreasuryAccountId(treasuryId)
    .setSupplyType(TokenSupplyType.Finite)
    .setMaxSupply(CID.length)
    .setCustomFees([nftCustomFee])
    .setAdminKey(adminKey)
    .setSupplyKey(supplyKey)
    .setPauseKey(pauseKey)
    .setFreezeKey(freezeKey)
    .setWipeKey(wipeKey)
    .freezeWith(client)
    .sign(treasuryKey);

  const nftCreateTxSign = await nftCreate.sign(adminKey);
  const nftCreateSubmit = await nftCreateTxSign.execute(client);
  const nftCreateRx = await nftCreateSubmit.getReceipt(client);
  const tokenId = nftCreateRx.tokenId;
  console.log(`Created NFT with Token ID: ${tokenId} \n`);

}

main();
```







Hedera Smart Contracts

“the one we’ve already built a use case for

[for our clients] is around tokenization of assets. The origin of that comes from speaking with our commercial real estate clients who have large, commercial real estate assets that are difficult to deal with”

Scott Thiel | Partner & Hedera Governing Council Member | DLA Piper

Build Using Solidity

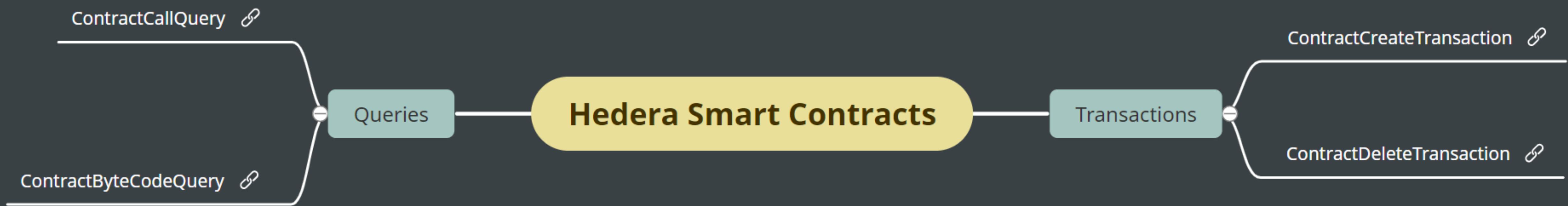
Run smart contracts written in Solidity, unchanged and using existing standards.

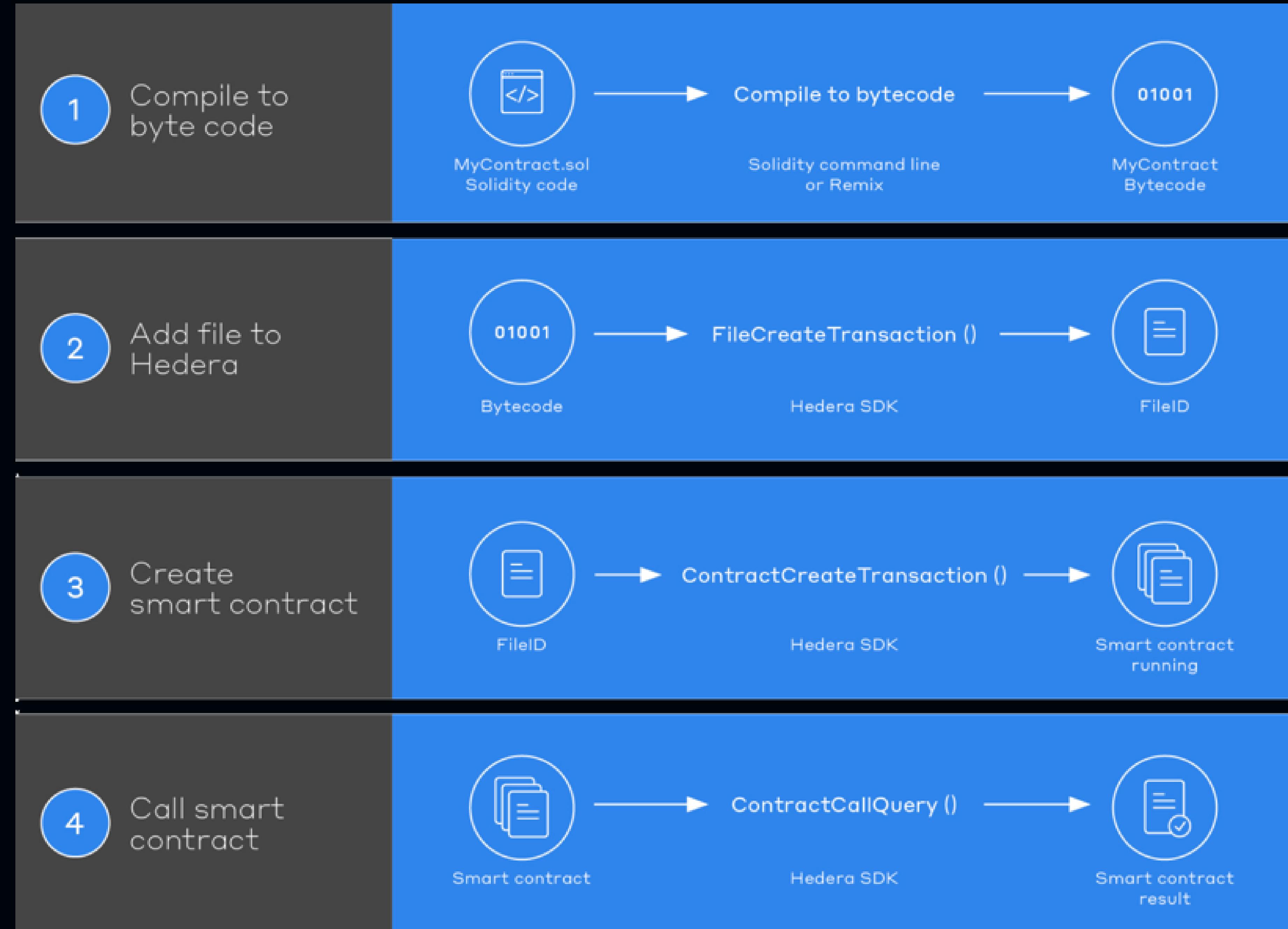
Fair Transaction Ordering

Like all services, smart contracts are executed in order received, never the amount of gas paid. No need to pay extra to be included earlier in history.

Admin privileges

Transparently define administrative keys to a contract, allowing owners to make sometimes drastically needed alterations to an otherwise immutable contract... Or not, it's up to your implementation!







LookupContract.sol

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity >=0.7.0 <0.9.0;

contract LookupContract{

mapping (string => uint) public myDirectory;

constructor (string memory _name, uint _mobileNumber) public {
    myDirectory[_name] = _mobileNumber;
}

function setMobileNumber(string memory _name, uint _mobileNumber) public{
    myDirectory[_name] = _mobileNumber;
}

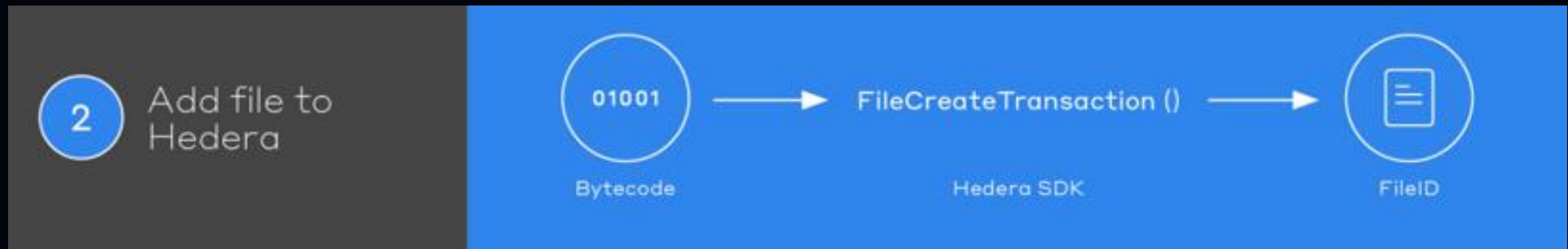
function getMobileNumber(string memory _name) public view returns(uint){
    return myDirectory[_name];
}

}
```

JavaScript SDK

```
// Import the compiled contract bytecode
const contractBytecode = fs.readFileSync("LookupContract_sol_LookupContract.bin");
```





JavaScript SDK

```
// Create a file on Hedera and store the bytecode
const fileCreateTx = new FileCreateTransaction()
  .setContents(contractBytecode)
  .setKeys([operatorKey])
  .freezeWith(client);

const fileCreateSign = await fileCreateTx.sign(operatorKey);
const fileCreateSubmit = await fileCreateSign.execute(client);
const fileCreateRx = await fileCreateSubmit.getReceipt(client);
const bytecodeFileId = fileCreateRx.fileId;
console.log(`- The bytecode file ID is: ${bytecodeFileId} \n`);
```



JavaScript SDK

```
// Instantiate the smart contract
const contractInstantiateTx = new ContractCreateTransaction()
  .setBytecodeFileId(bytecode fileId)
  .setGas(100000)
  .setConstructorParameters(new ContractFunctionParameters().addString("Alice").addUInt256(111111));
const contractInstantiateSubmit = await contractInstantiateTx.execute(client);
const contractInstantiateRx = await contractInstantiateSubmit.getReceipt(client);
const contractId = contractInstantiateRx.contractId;
const contractAddress = contractId.toSolidityAddress();
console.log(`- The smart contract ID is: ${contractId} \n`);
console.log(`- The smart contract ID in Solidity format is: ${contractAddress} \n`);
```



JavaScript SDK

```
// Query the contract to check changes in state variable
const contractQueryTx = new ContractCallQuery()
  .setContractId(contractId)
  .setGas(100000)
  .setFunction("getMobileNumber", new ContractFunctionParameters().addString("Alice"));
const contractQuerySubmit = await contractQueryTx.execute(client);
const contractQueryResult = contractQuerySubmit.getUint256(0);
console.log(`- Here's the phone number that you asked for: ${contractQueryResult}\n`);
```



```
// Call contract function to update the state variable
const contractExecuteTx = new ContractExecuteTransaction()
  .setContractId(contractId)
  .setGas(100000)
  .setFunction("setMobileNumber", new ContractFunctionParameters().addString("Bob").addUint256(222222));
const contractExecuteSubmit = await contractExecuteTx.execute(client);
const contractExecuteRx = await contractExecuteSubmit.getReceipt(client);
console.log(`- Contract function call status: ${contractExecuteRx.status}\n`);

// Query the contract to check changes in state variable
const contractQueryTx1 = new ContractCallQuery()
  .setContractId(contractId)
  .setGas(100000)
  .setFunction("getMobileNumber", new ContractFunctionParameters().addString("Bob"));
const contractQuerySubmit1 = await contractQueryTx1.execute(client);
const contractQueryResult1 = contractQuerySubmit1.getUint256(0);
console.log(`- Here's the phone number that you asked for: ${contractQueryResult1}\n`);
```

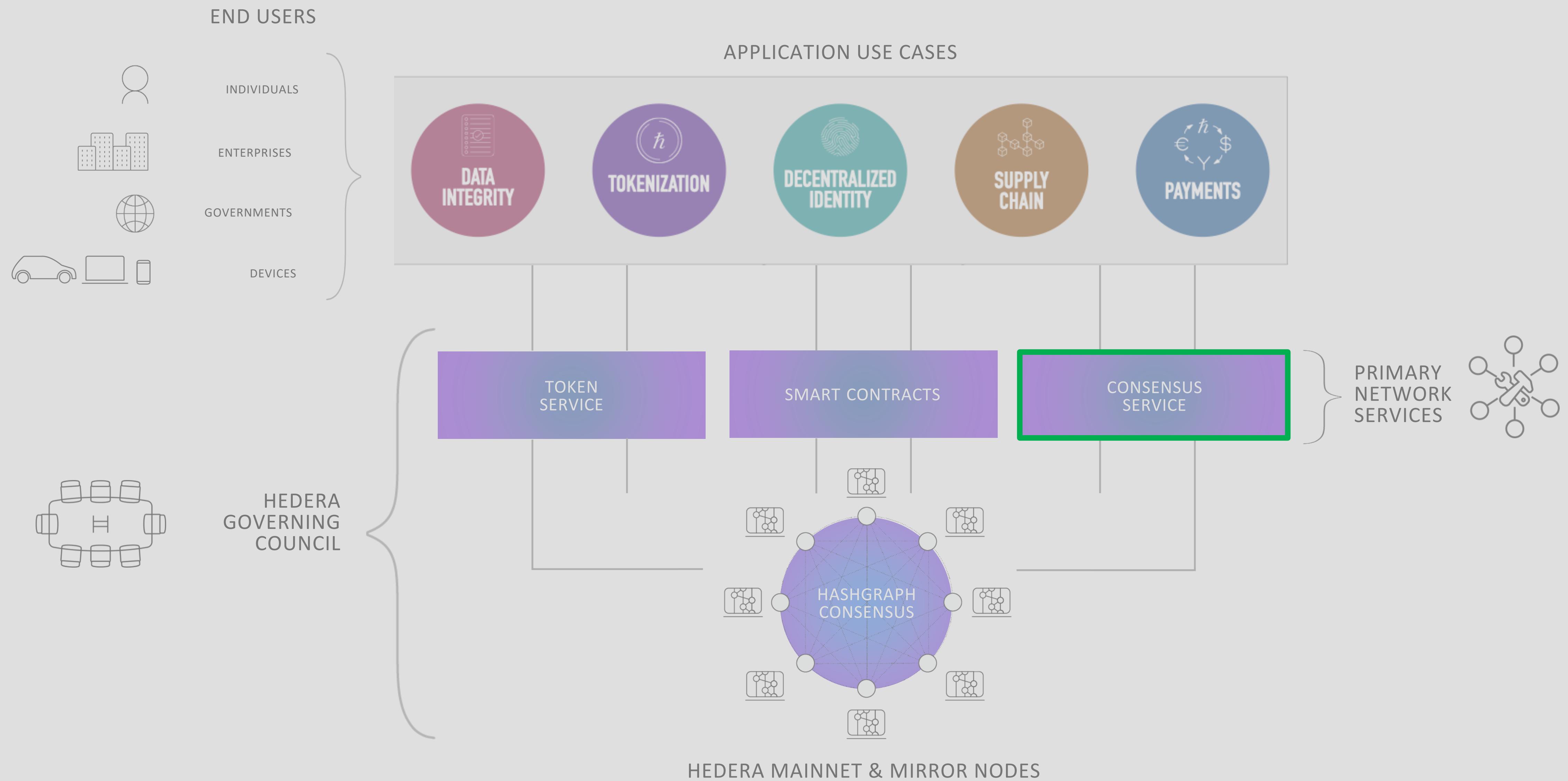
Console Output

- The bytecode file ID is: 0.0.30468526

 - The smart contract ID is: 0.0.30468614

 - The smart contract ID in Solidity format is: 000000000000000000000000000000001d0ea06

 - Here's the phone number that you asked for: 111111
 - Contract function call status: SUCCESS
 - Here's the phone number that you asked for: 222222





Consensus Service

“There are a number of challenges in using traditional ordering services for decentralized applications. By exposing this capability, Hedera is making a meaningful contribution to furthering the capabilities of distributed networks.”

Bryan Gross | Principal Product Manager | IBM Blockchain

64

Build with native performance

Create applications that demand high throughput 10,000+ txs per second, per shard

Amplify trust

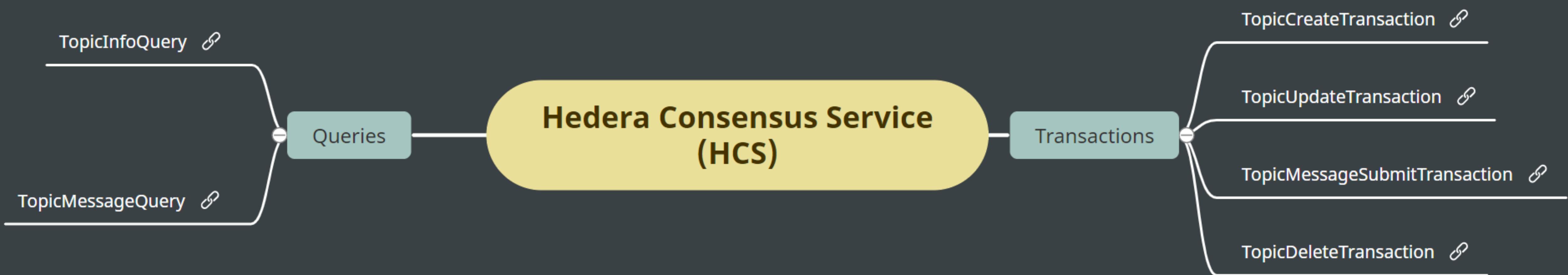
Validate the data in a multi-party database, notary service, or anything that needs transparency with shared infrastructure

Guarantee order

Results in a fair total consensus order, with accurate timestamps, a running hash of order in the topic, along with a state proof that guarantees it is correct

Integrate into existing ecosystems

Open-source integrations with Hyperledger Fabric create new architectures



TopicCreateTransaction()

```
const { Client, TopicCreateTransaction } = require("@hashgraph/sdk");
client = Client.forTestnet().setOperator(operatorId, operatorKey);

async function main() {

  tx = await new TopicCreateTransaction().execute(client);
  receipt = await tx.getReceipt(client);
  newTopicId = receipt.topicId;

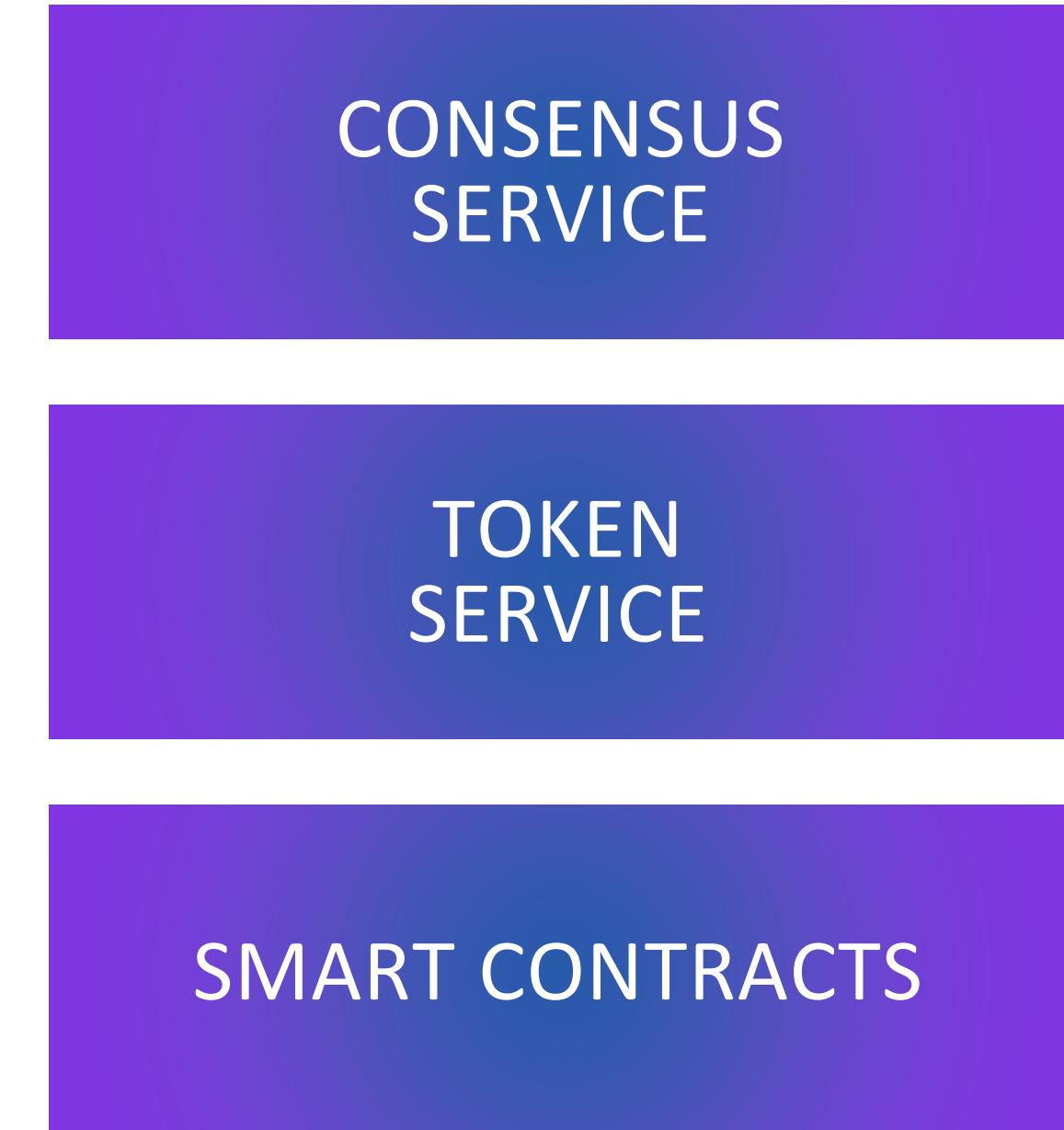
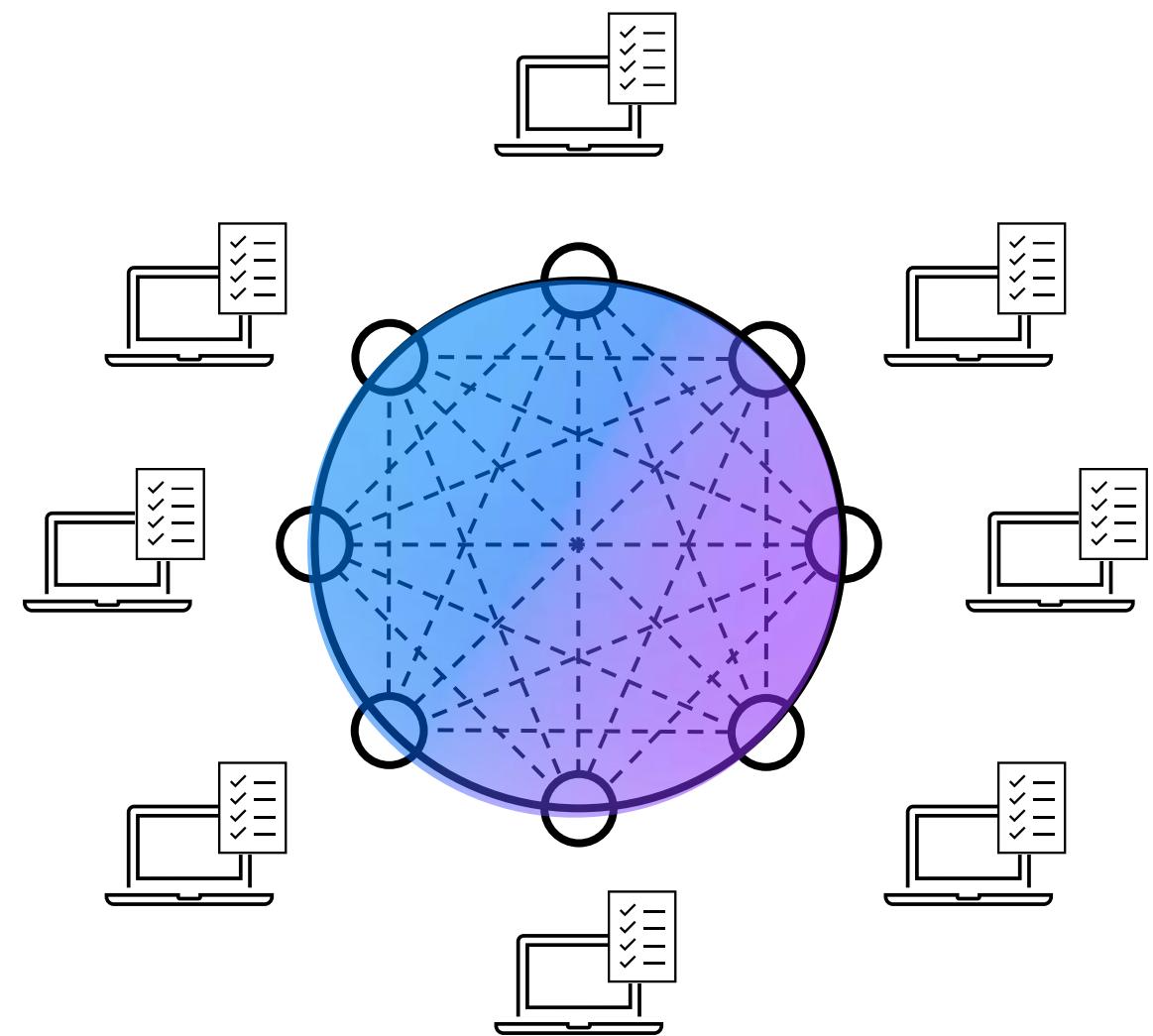
  console.log(`New HCS topic ID: ${newTopicId}`);
}

void main();
```

TopicMessageQuery()

```
new TopicMessageQuery()
  .setTopicId(topicId)
  .subscribe(
    client,
    (error) => console.log(`Error: ${error}`),
    (message) => console.log(message.toString())
  );
```

In this session, you will learn how the **Hedera network functions**, how you can **start development**, and **where to learn more**



Introduction

Understand
the Hedera Network

Start Developing on
Hedera

Get Resources and
Learn More

Tutorials

Learn more about building on Hedera Hashgraph through a hands-on-tutorial.

[GET STARTED](#)



Considering use cases

What makes a good decentralized use case?

- Transparent collaboration
- Auditability and attestability
- Censorship resistance
- Composability & compounding
- Inherent lack of trust between parties

SECURITY TOKENS ISSUERS & EXCHANGES



TOKO



BANKS



PAYMENTS INFRASTRUCTURE



STABLE COINS



MICROPAYMENTS



CBDC



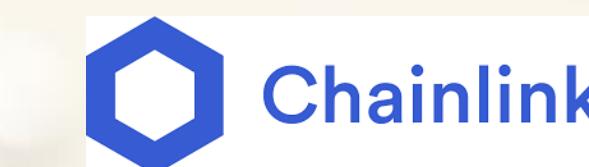
EMTECH

ZIMBALI

WALLETS



DEFI & NEOBANKS



SYSTEMS INTEGRATORS



CUSTODY



KABUTO



Develops and maintains the Universal Positive Offer File for manufacturer coupons using the Hedera Consensus Service.

"Only Hedera Hashgraph was able to provide the real-time, tamper-proof logging capabilities that we needed to bring transparency, trust, agnosticism and industry oversight to a platform that connects all coupon industry stakeholders"

BRANDI JOHNSON | CEO

QUICK FACTS

CHALLENGE

Coupon and associated promotional data is fragmented, non-standardized, and rife with fraud. Without the necessary controls, transparency or trust, the industry usage of this promotional vehicle has decreased in recent years.

SOLUTION

The Coupon Bureau can bring its standardization and added efficiencies to the marketplace with transparency and trust. By allowing 3rd party validation and audit capabilities among authorized stakeholders, the industry will be able to have confidence in this centralized, agnostic solution.

MORE INFORMATION

[HEDERA.COM/USERS/COUPON-BUREAU](https://hedera.com/users/COUPON-BUREAU)

eftpos

Micropayments that aims to create a seamless Australian payment experience for web users by providing an alternative to traditional online paywalls and subscriptions.

“By working with Hedera, we are leveraging next generation payments infrastructure technology that can support Australian dollar-based micropayments and open up entirely new ways of conducting business online.”

ROBERT ALLEN | ENTREPRENEUR IN RESIDENCE

CHALLENGE

Eftpos needed to facilitate payments and micropayments using a digital dollar that enables fast, secure, and affordable micropayment transactions.

SOLUTION

Eftpos will test the capability of a digital Australian dollar stable coin, enabled by the hedera consensus service, allowing consumers to load a wallet with a few dollars and then pay for web-based content seamlessly.



Real-time auditing at scale to eliminate advertising fraud and remove costly intermediaries.

"Hedera has proven it has the scale, speed and reliability to handle all of the live advert tracking data and programmatic event data that we have been handling in recent campaigns."

IAN MULLINS | FOUNDER & CEO

CHALLENGE

The digital advertising industry is rife with fraud and trust issues, while over 75% of fees are placed into the pockets of expensive intermediaries.

SOLUTION

AdsDax is leveraging Hedera to build a platform that will track and verify advertising events and engagement, while providing security, resilience, scalability, and transparency, without the need for costly intermediaries.

EVERWARE

Digitally track and monitor critical or high-value assets.

“For managing highly critical information, like the COVID-19 vaccine’s temperature, Hedera Hashgraph provides us immutable trust at the speed required for our IoT devices.”

TOM SCREEN | TECHNICAL DIRECTOR

QUICK FACTS

CHALLENGE

Monitoring the temperature of key assets, like the COVID-19 vaccine is of interest to multiple parties, with liability and risk to be had. Everyware wanted a more secure method to manage this device data than a traditional database.

SOLUTION

Everyware uses Hedera Consensus Service to keep an immutable, tamper-proof, and verifiable record of sensor data.

MORE INFORMATION

[HEDERA.COM/USERS/EVERYWARE](https://hedera.com/users/everyware)



power transition

Peer-to-peer energy microgrids built on a stable network with high-throughput, to exchange value.

“The Hedera Token Service has transformed the way our customers can trade electricity and other forms of value across the energy sector.”

JIRO OLCOTT | DIRECTOR

QUICK FACTS

CHALLENGE

Peer-to-peer energy trading and microgrid management is becoming a reality with the increased adoption of solar panels and batteries. Power Transition needed a way to enable economically feasible micropayments between disparate parties who require trust.

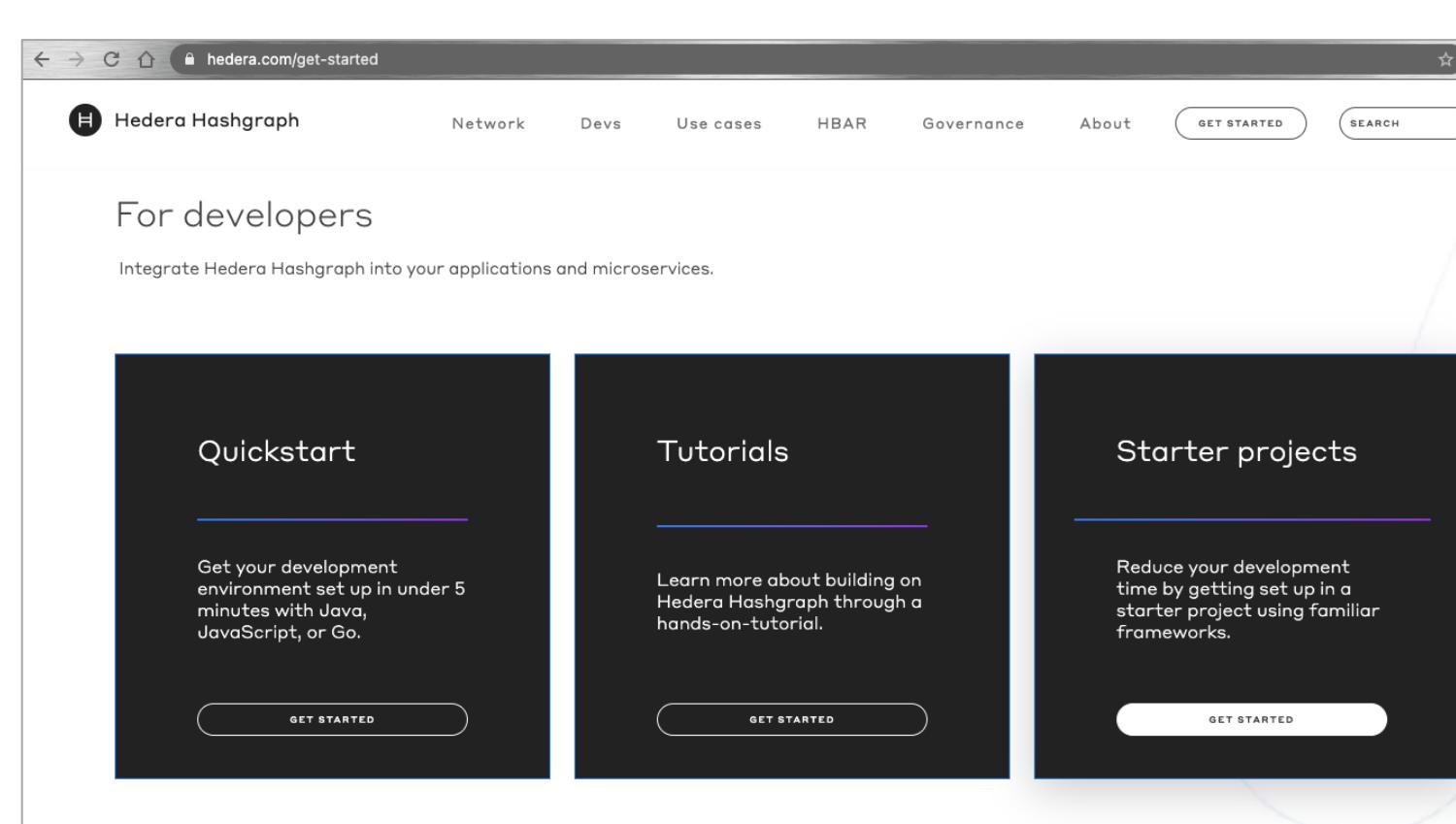
SOLUTION

Hedera was chosen for its network stability, high-throughput, low latency, and predictable fees to power HBAR cryptocurrency micropayments between homes on the Power Transition microgrid.

MORE INFORMATION

[HEDERA.COM/USERS/POWER-TRANSITION](https://hedera.com/users/power-transition)

Learn more and start building web 3!



For developers

Integrate Hedera Hashgraph into your applications and microservices.

Quickstart

Get your development environment set up in under 5 minutes with Java, JavaScript, or Go.

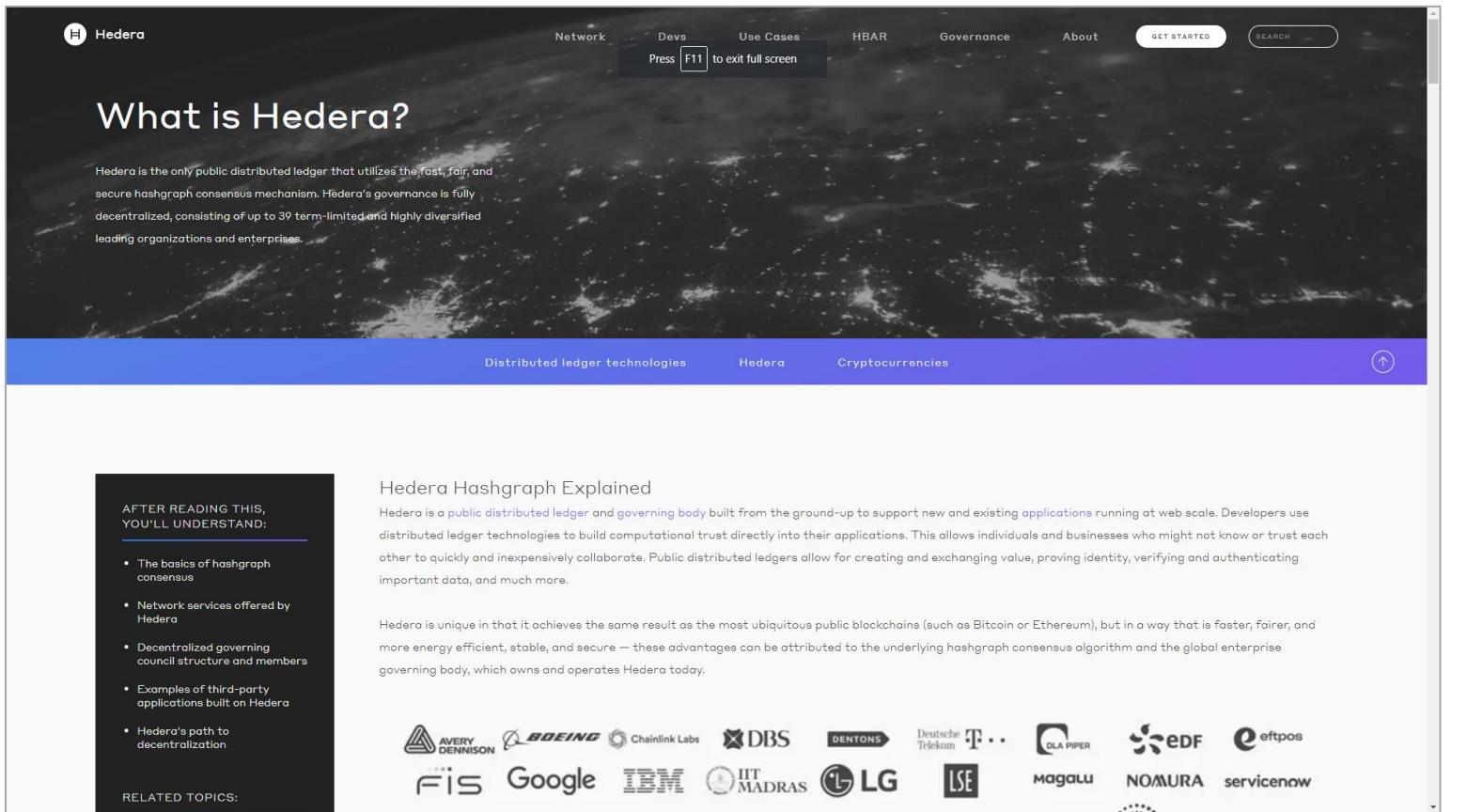
Tutorials

Learn more about building on Hedera Hashgraph through a hands-on-tutorial.

Starter projects

Reduce your development time by getting set up in a starter project using familiar frameworks.

GET STARTED



What is Hedera?

Hedera is the only public distributed ledger that utilizes the fast, fair, and secure hashgraph consensus mechanism. Hedera's governance is fully decentralized, consisting of up to 39 term-limited and highly diversified leading organizations and enterprises.

Distributed ledger technologies **Hedera** **Cryptocurrencies**

Hedera Hashgraph Explained

Hedera is a public distributed ledger and governing body built from the ground-up to support new and existing applications running at web scale. Developers use distributed ledger technologies to build computational trust directly into their applications. This allows individuals and businesses who might not know or trust each other to quickly and inexpensively collaborate. Public distributed ledgers allow for creating and exchanging value, proving identity, verifying and authenticating important data, and much more.

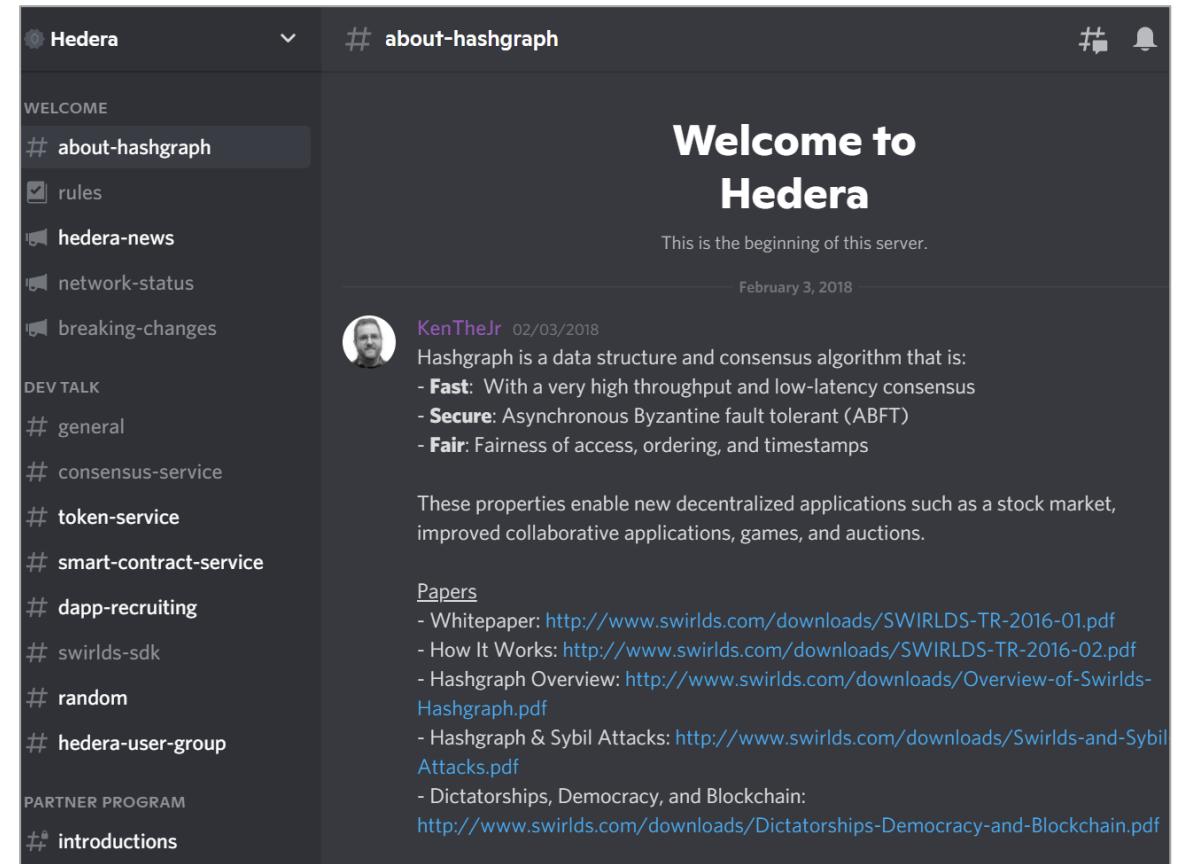
Hedera is unique in that it achieves the same result as the most ubiquitous public blockchains (such as Bitcoin or Ethereum), but in a way that is faster, fairer, and more energy efficient, stable, and secure – these advantages can be attributed to the underlying hashgraph consensus algorithm and the global enterprise governing body, which owns and operates Hedera today.

AFTER READING THIS, YOU'LL UNDERSTAND:

- The basics of hashgraph consensus
- Network services offered by Hedera
- Decentralized governing council structure and members
- Examples of third-party applications built on Hedera
- Hedera's path to decentralization

RELATED TOPICS:

AVERY DENISON, BOEING, Chainlink Labs, DBS, DENTON, Deutsche Telekom, FIS, Google, IBM, IIT MADRAS, LG, LSE, MAGALU, EDF, eftpos, NOMURA, servicenow



WELCOME

about-hashgraph

rules

hederanews

network-status

breaking-changes

DEV TALK

general

consensus-service

token-service

smart-contract-service

dapp-recruiting

swirls-sdk

random

hederau-user-group

PAPERS

- Whitepaper: <http://www.swirls.com/downloads/SWIRLDS-TR-2016-01.pdf>
- How It Works: <http://www.swirls.com/downloads/SWIRLDS-TR-2016-02.pdf>
- Hashgraph Overview: <http://www.swirls.com/downloads/Overview-of-Swirls-Hashgraph.pdf>
- Hashgraph & Sybil Attacks: <http://www.swirls.com/downloads/Swirls-and-Sybil-Attacks.pdf>
- Dictatorships, Democracy, and Blockchain: <http://www.swirls.com/downloads/Dictatorships-Democracy-and-Blockchain.pdf>

Welcome to Hedera

This is the beginning of this server.

February 3, 2018

KenTheJr 02/03/2018

Hashgraph is a data structure and consensus algorithm that is:

- **Fast:** With a very high throughput and low-latency consensus
- **Secure:** Asynchronous Byzantine fault tolerant (ABFT)
- **Fair:** Fairness of access, ordering, and timestamps

These properties enable new decentralized applications such as a stock market, improved collaborative applications, games, and auctions.

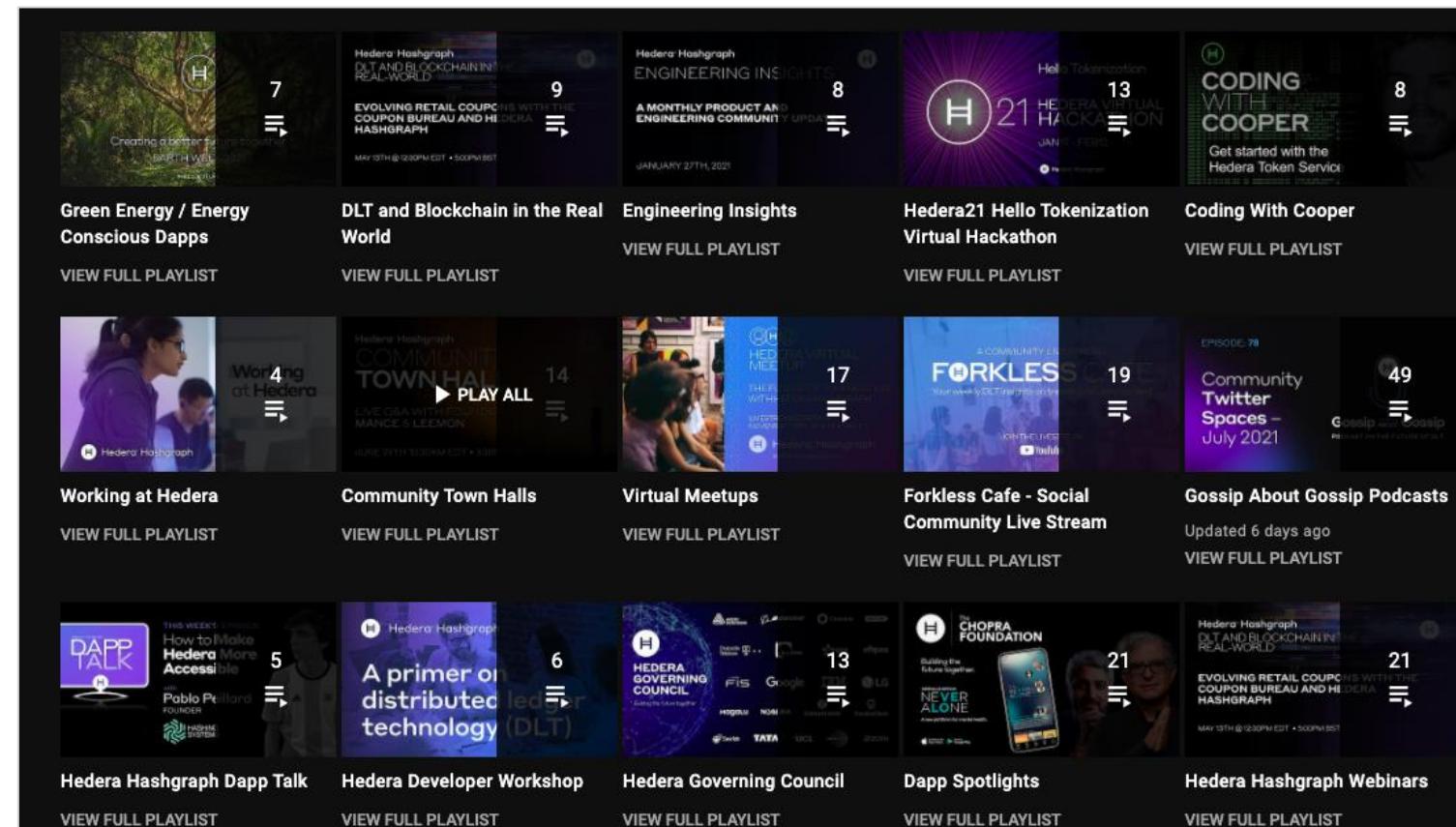
Papers

- Whitepaper: <http://www.swirls.com/downloads/SWIRLDS-TR-2016-01.pdf>
- How It Works: <http://www.swirls.com/downloads/SWIRLDS-TR-2016-02.pdf>
- Hashgraph Overview: <http://www.swirls.com/downloads/Overview-of-Swirls-Hashgraph.pdf>
- Hashgraph & Sybil Attacks: <http://www.swirls.com/downloads/Swirls-and-Sybil-Attacks.pdf>
- Dictatorships, Democracy, and Blockchain: <http://www.swirls.com/downloads/Dictatorships-Democracy-and-Blockchain.pdf>

hedera.com/get-started

[Hedera Learning Center](#)

[Join Developer Discord](#)



Green Energy / Energy Conscious Dapps

DLT and Blockchain in the Real World

Engineering Insights

Hedera21 Hello Tokenization Virtual Hackathon

Coding With Cooper

Working at Hedera

Community Town Halls

Virtual Meetups

Forkless Cafe - Social Community Live Stream

Gossip About Gossip Podcasts

Hedera Hashgraph Dapp Talk

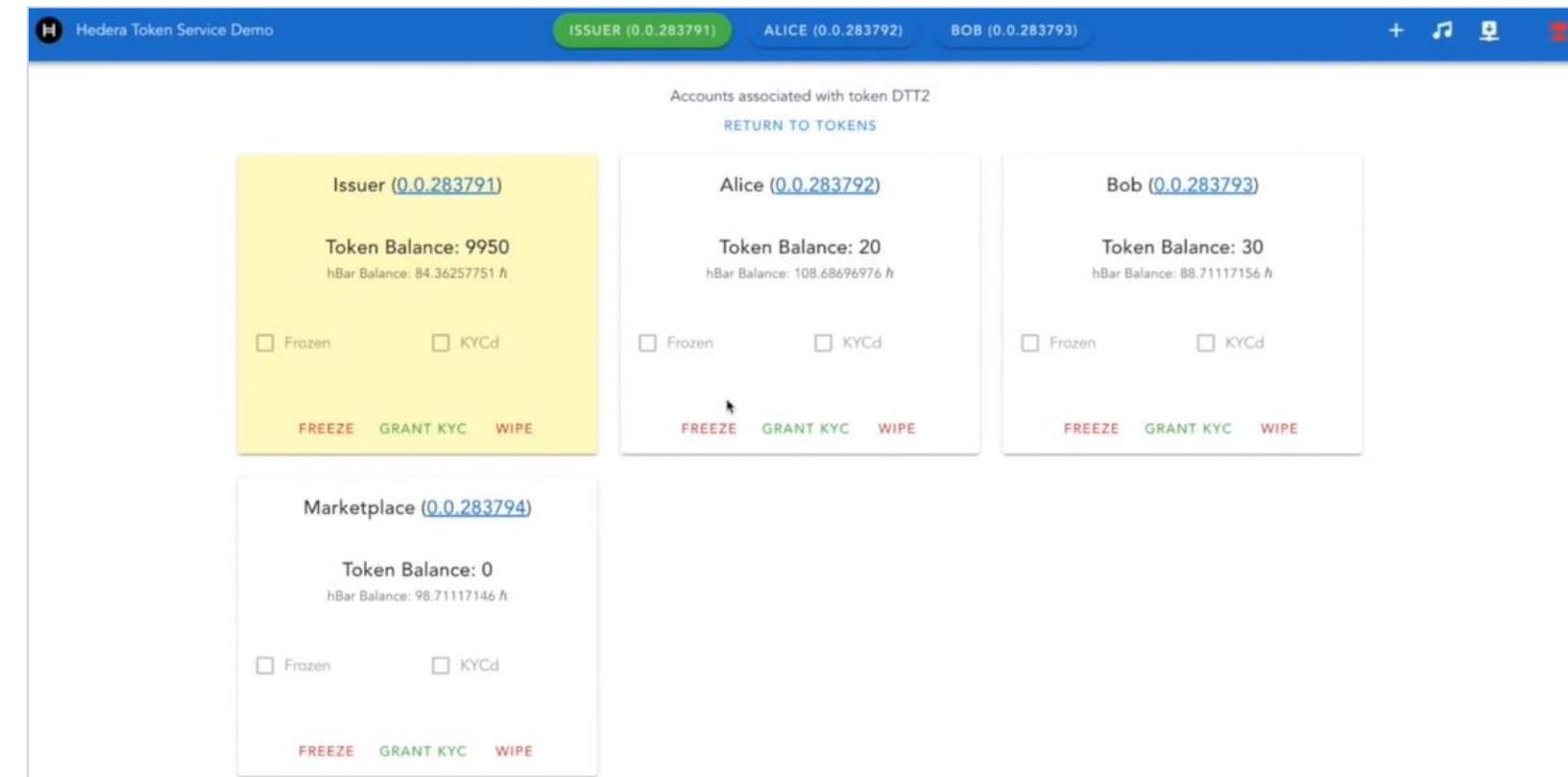
Hedera Developer Workshop

Hedera Governing Council

Dapp Spotlights

Hedera Hashgraph Webinars

[Hedera YouTube Channel](#)



Issuer (0.0.283791) ALICE (0.0.283792) BOB (0.0.283793)

Accounts associated with token DTT2

Issuer (0.0.283791)

Token Balance: 9950
hBar Balance: 84.36257751 h

Frozen KYCd

Alice (0.0.283792)

Token Balance: 20
hBar Balance: 108.68696976 h

Frozen KYCd

Bob (0.0.283793)

Token Balance: 30
hBar Balance: 88.71117156 h

Frozen KYCd

Marketplace (0.0.283794)

Token Balance: 0
hBar Balance: 98.71117146 h

Frozen KYCd

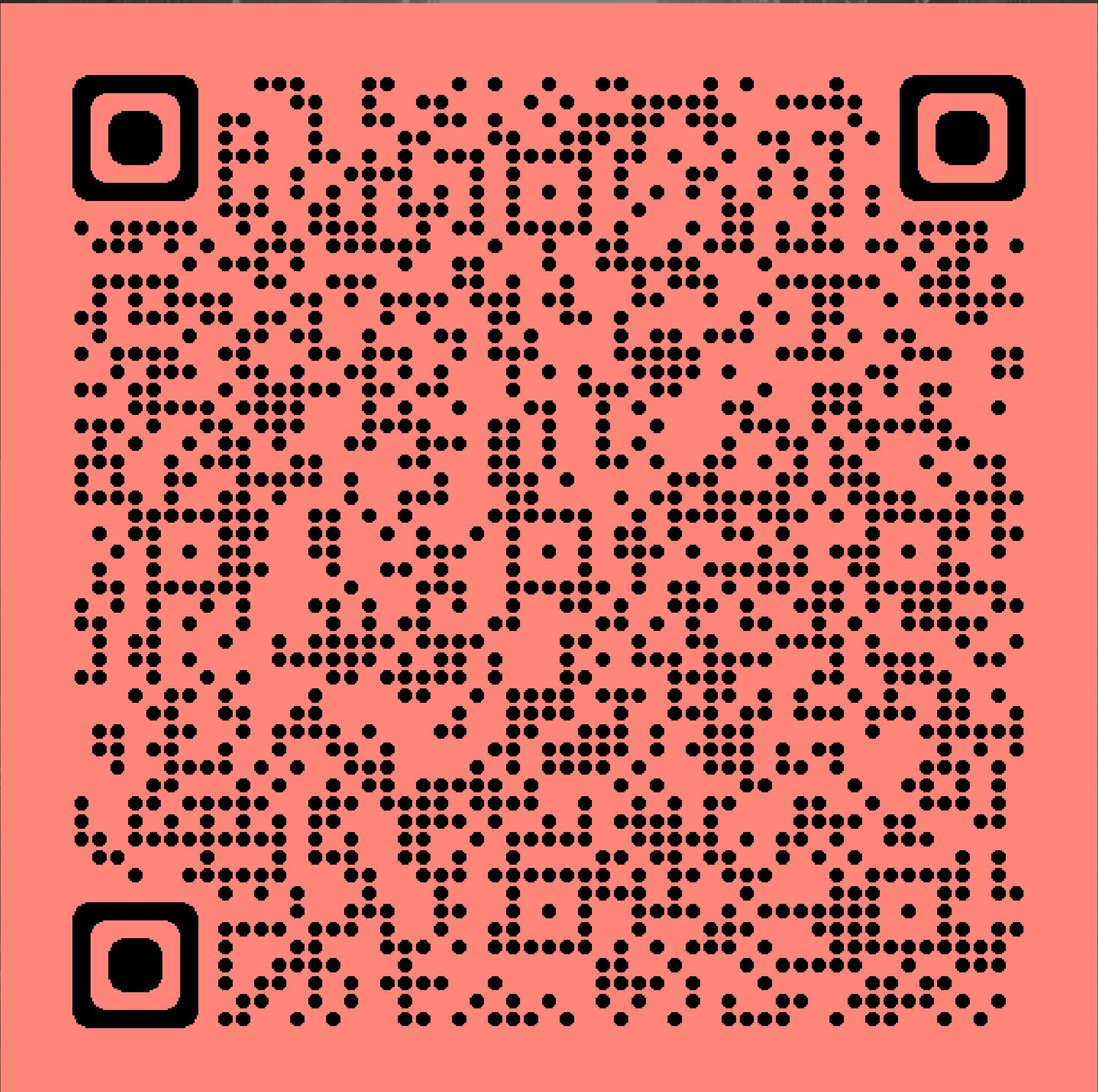
FREEZE **GRANT KYC** **WIPE**

[Application Demos](#)

Web 3 is ready for you, and with Hedera, you are ready for Web 3!



GET THE SLIDES NOW ➔



Ed Marquez – Hedera

Viv Diwakar – The HBAR Foundation



/ed-marquez | /in/vivdiwakar



@ed_marquez | @HBAR_foundation