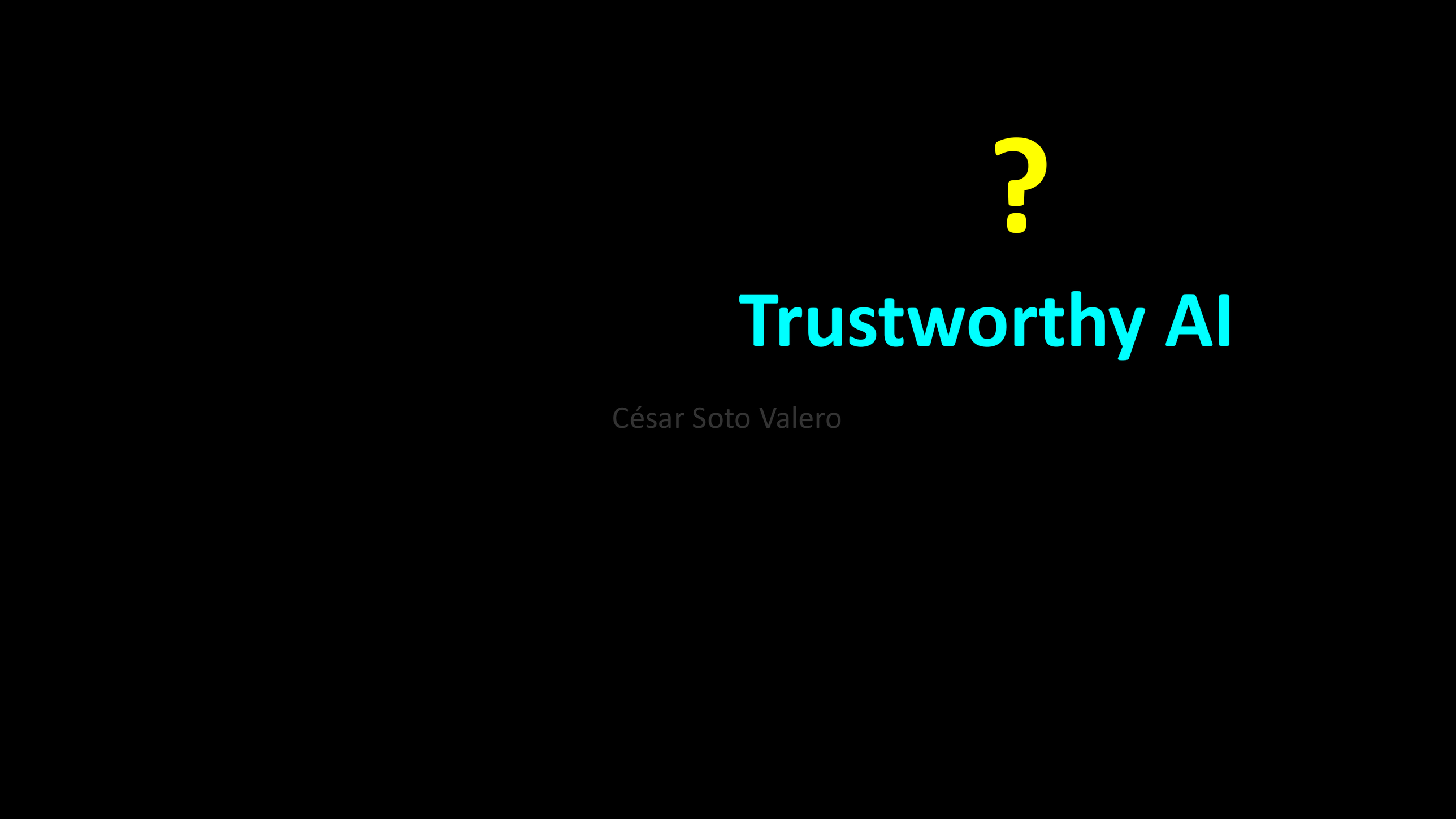
César Soto Valero

AI

César Soto Valero

Trustworthy AI

César Soto Valero


?
Trustworthy AI
César Soto Valero

Promise Trustworthy AI

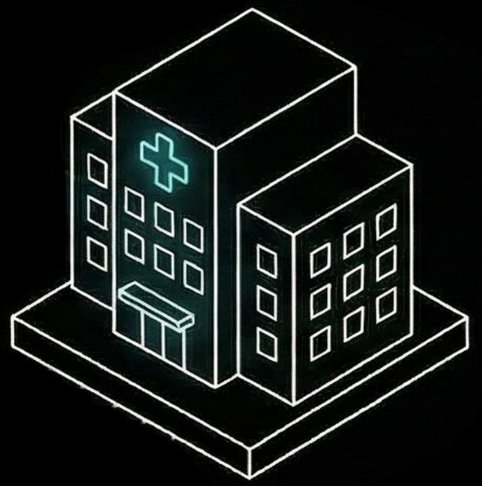
César Soto Valero

The Promise of Trustworthy AI

César Soto Valero

Trust

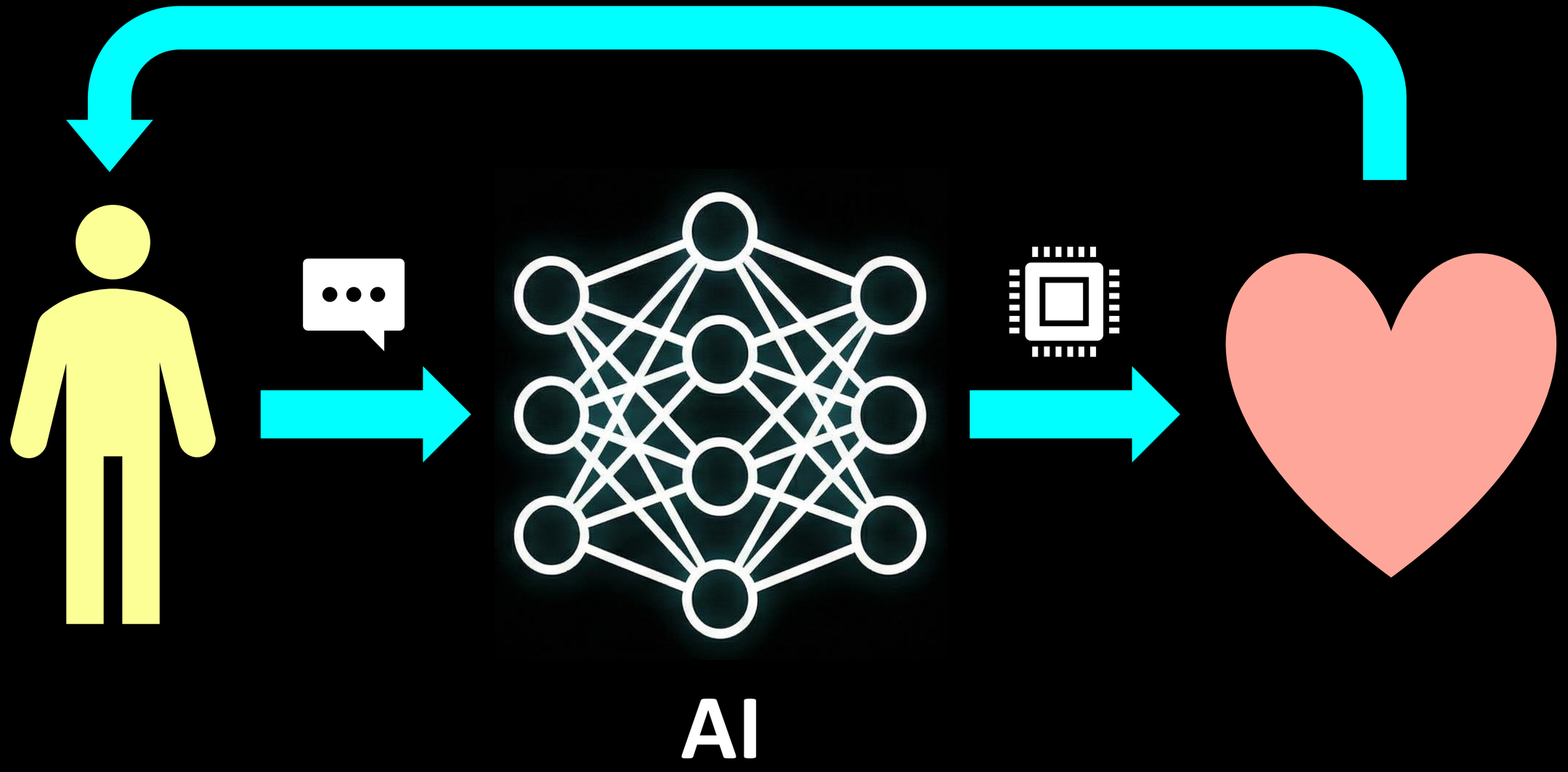
Trust

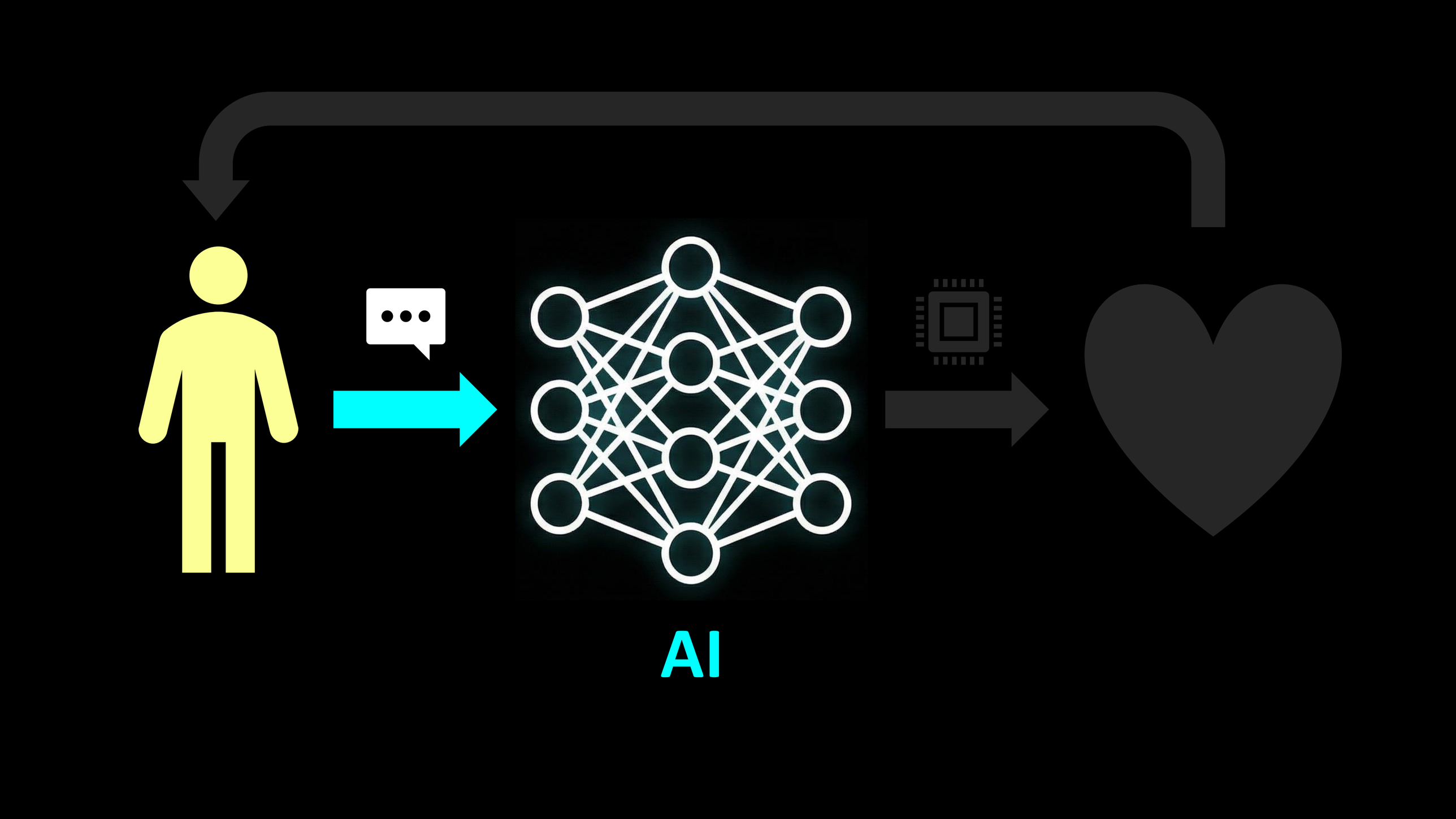


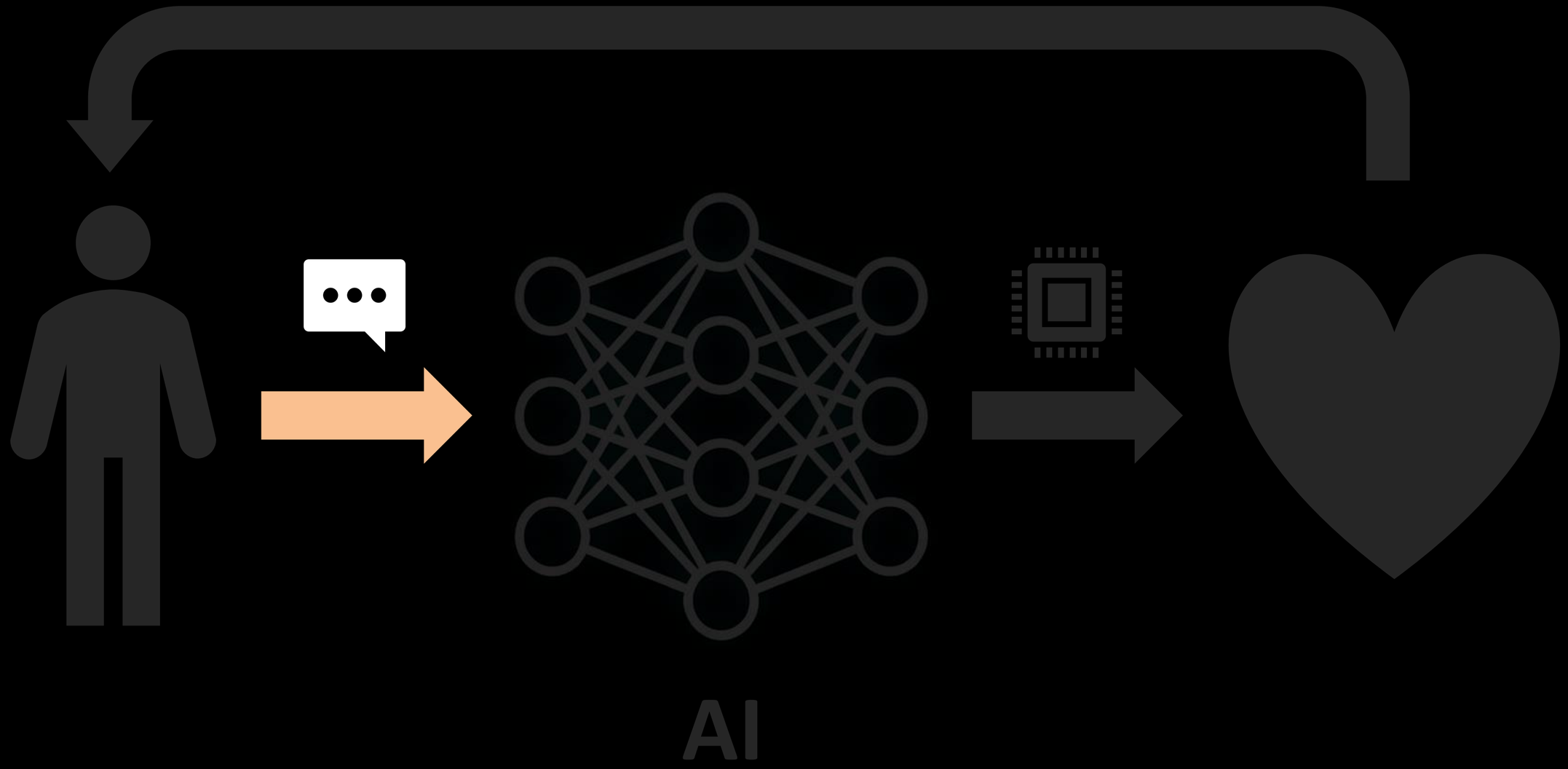
Healthcare

Education

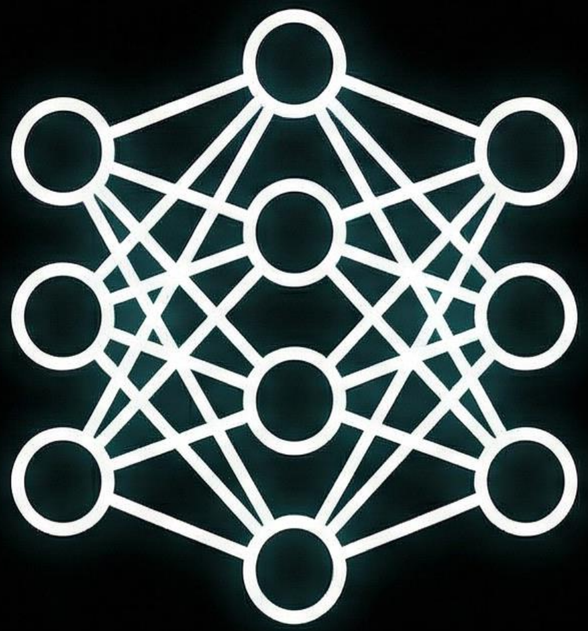
Finance


AI


AI

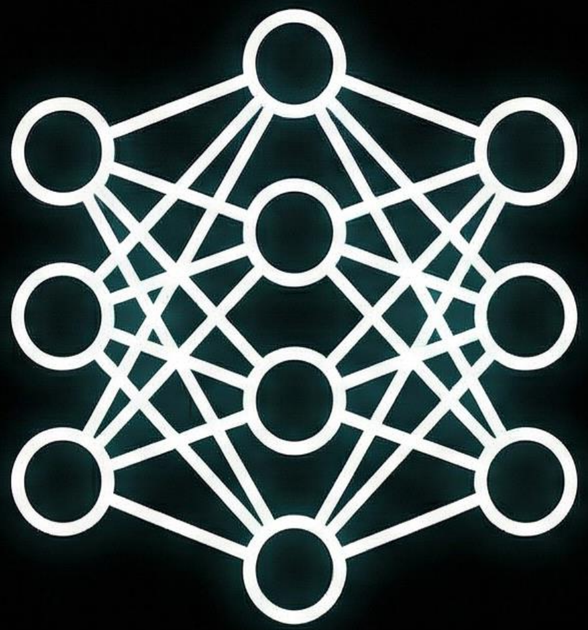

AI

More powerful AI



Less privacy

More powerful AI

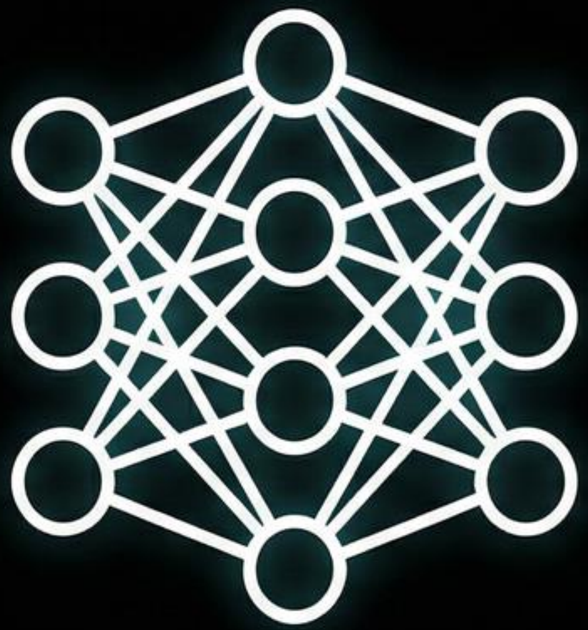


Less privacy

More privacy

Less powerful AI

More powerful AI



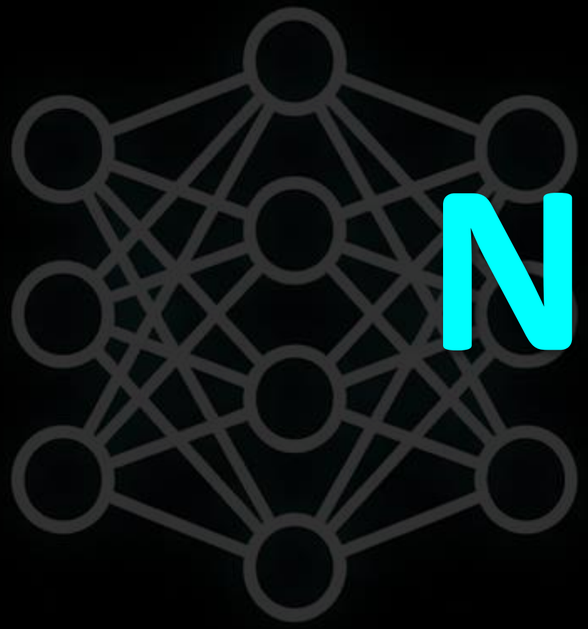
Less privacy

More privacy

Less powerful AI

More powerful AI

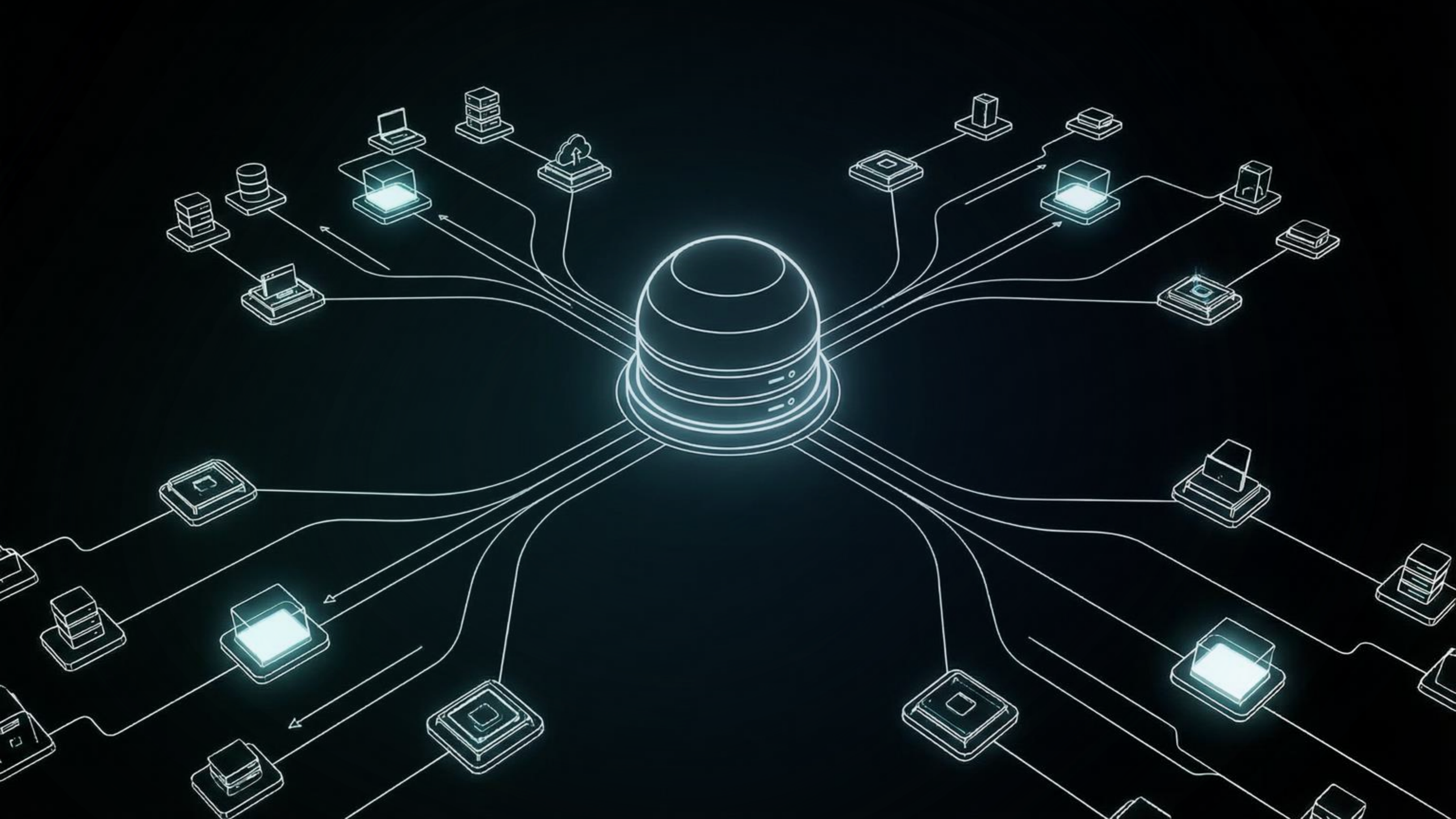
More privacy



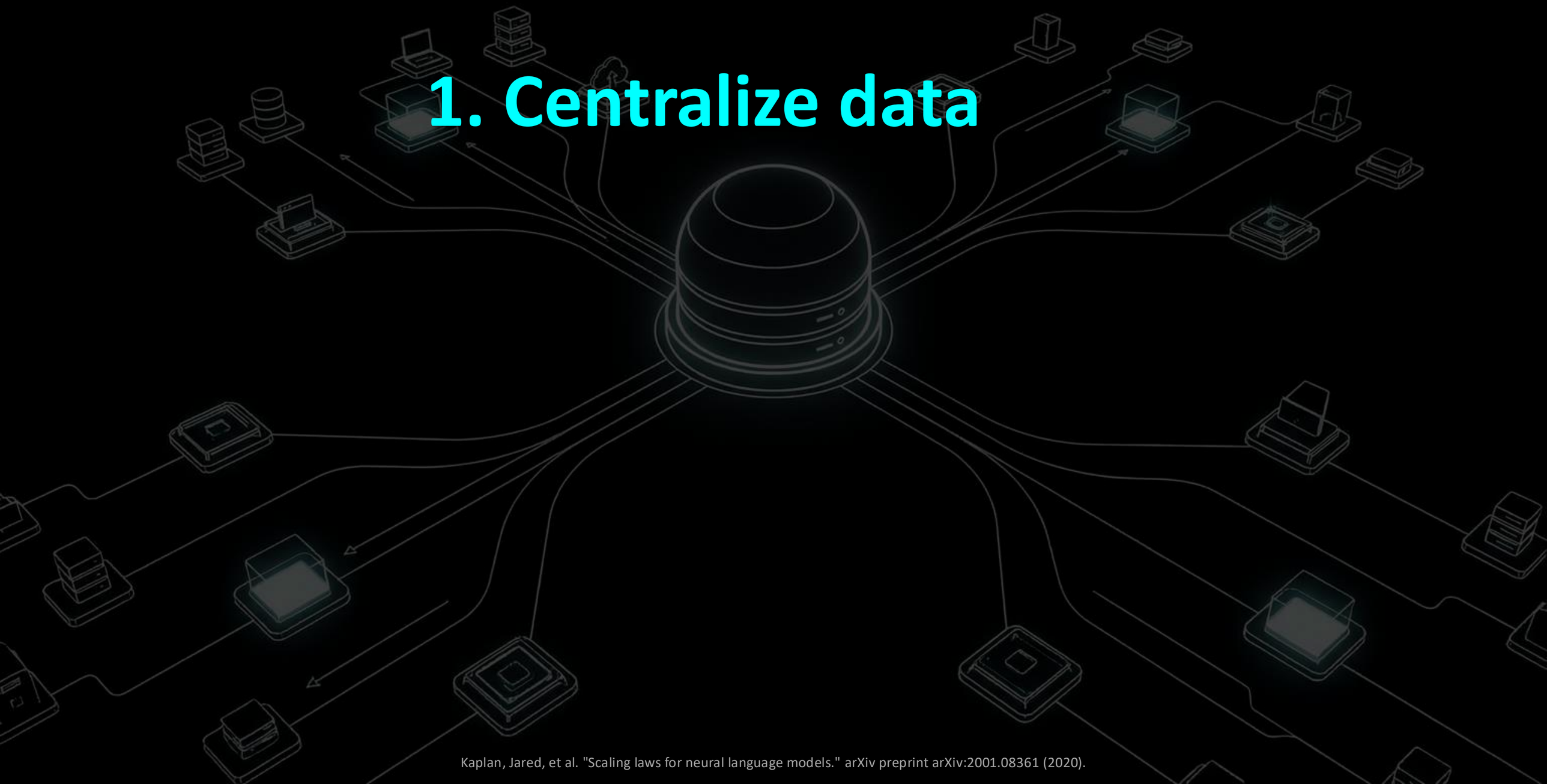
Not really!

Less privacy

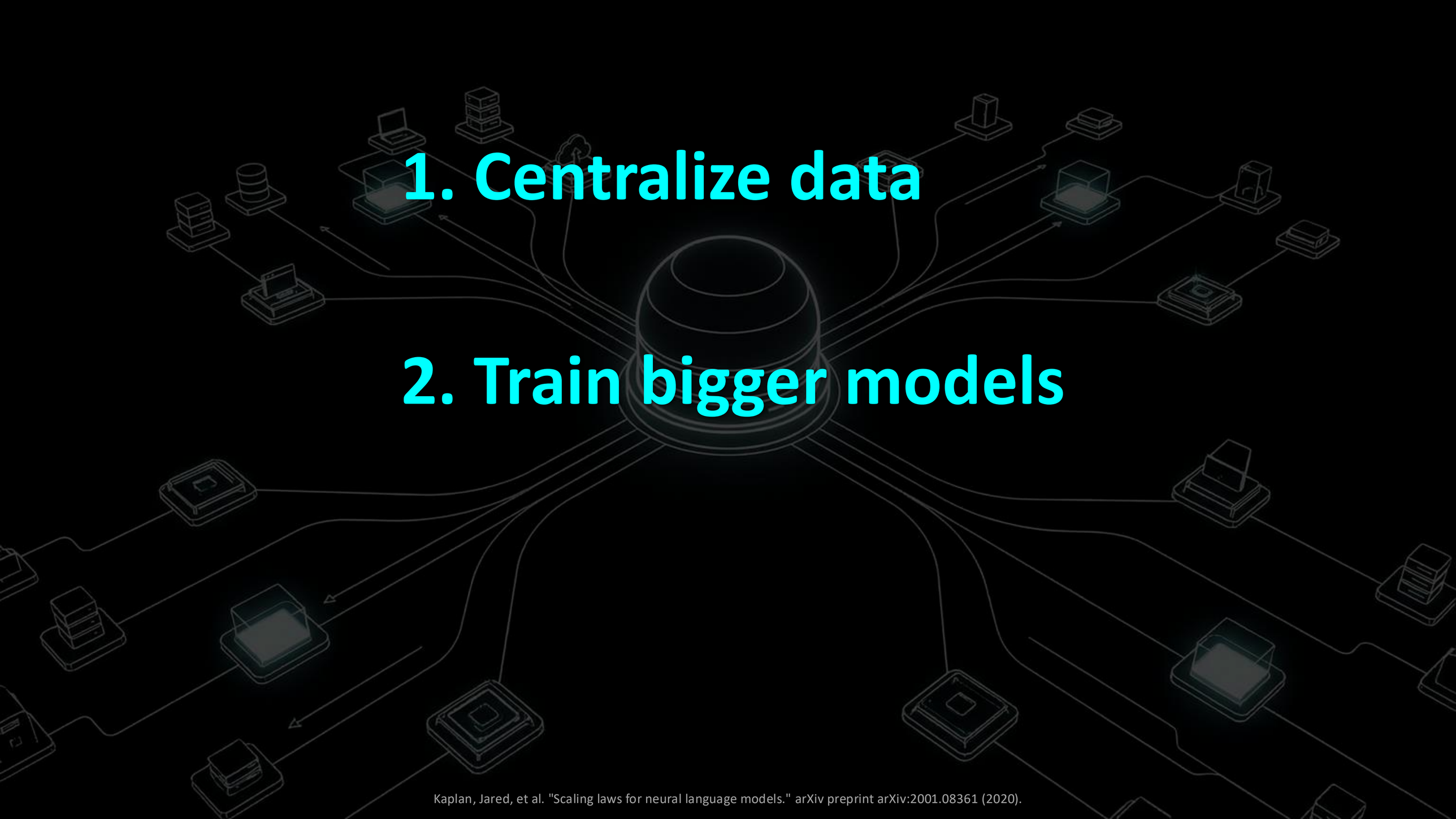
Less powerful AI



1. Centralize data



Kaplan, Jared, et al. "Scaling laws for neural language models." arXiv preprint arXiv:2001.08361 (2020).



Kaplan, Jared, et al. "Scaling laws for neural language models." arXiv preprint arXiv:2001.08361 (2020).

Kaplan, Jared, et al. "Scaling laws for neural language models." arXiv preprint arXiv:2001.08361 (2020).

Problematic

Problematic

Security

Problematic

Security

Compliance

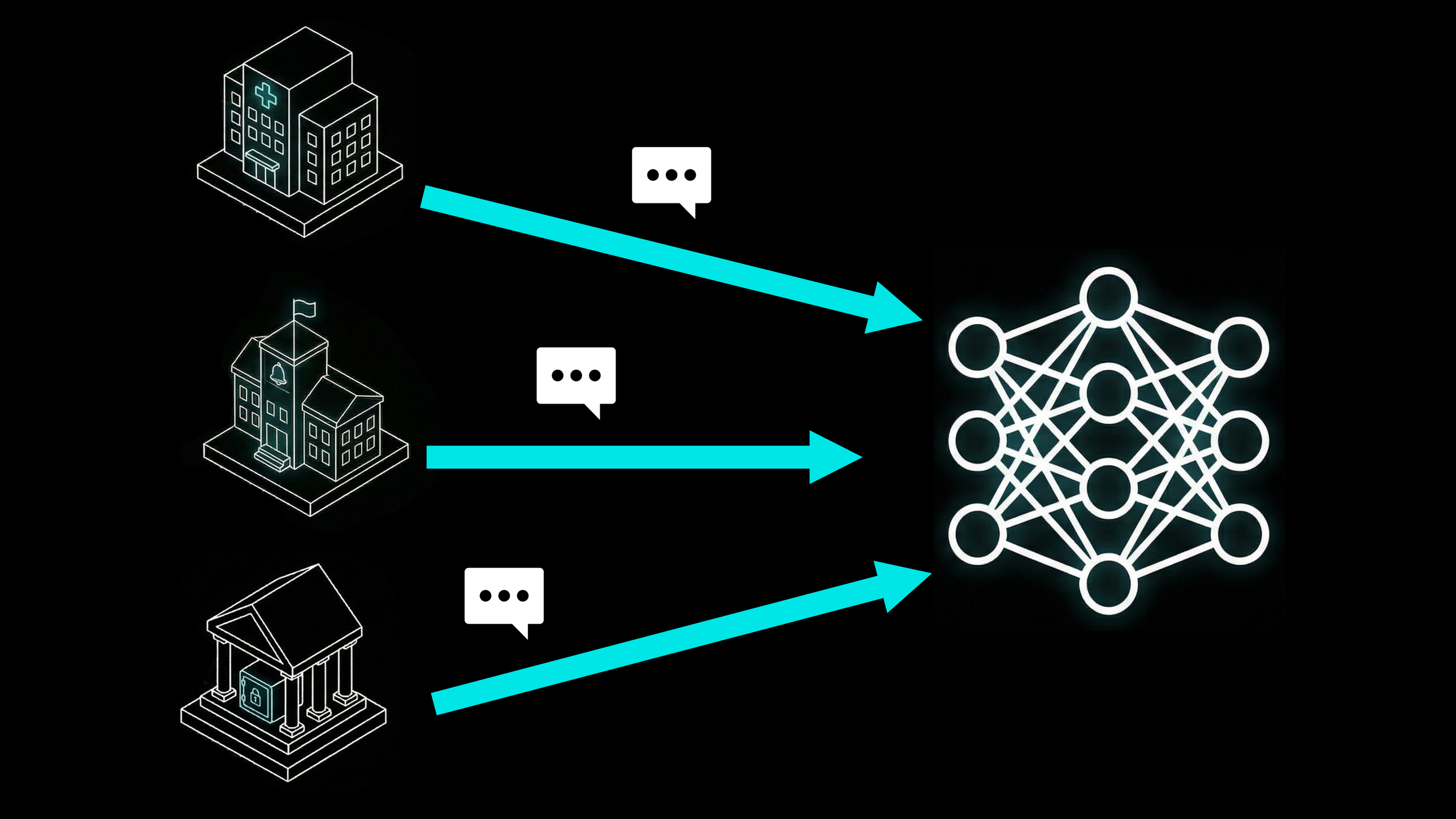
Problematic

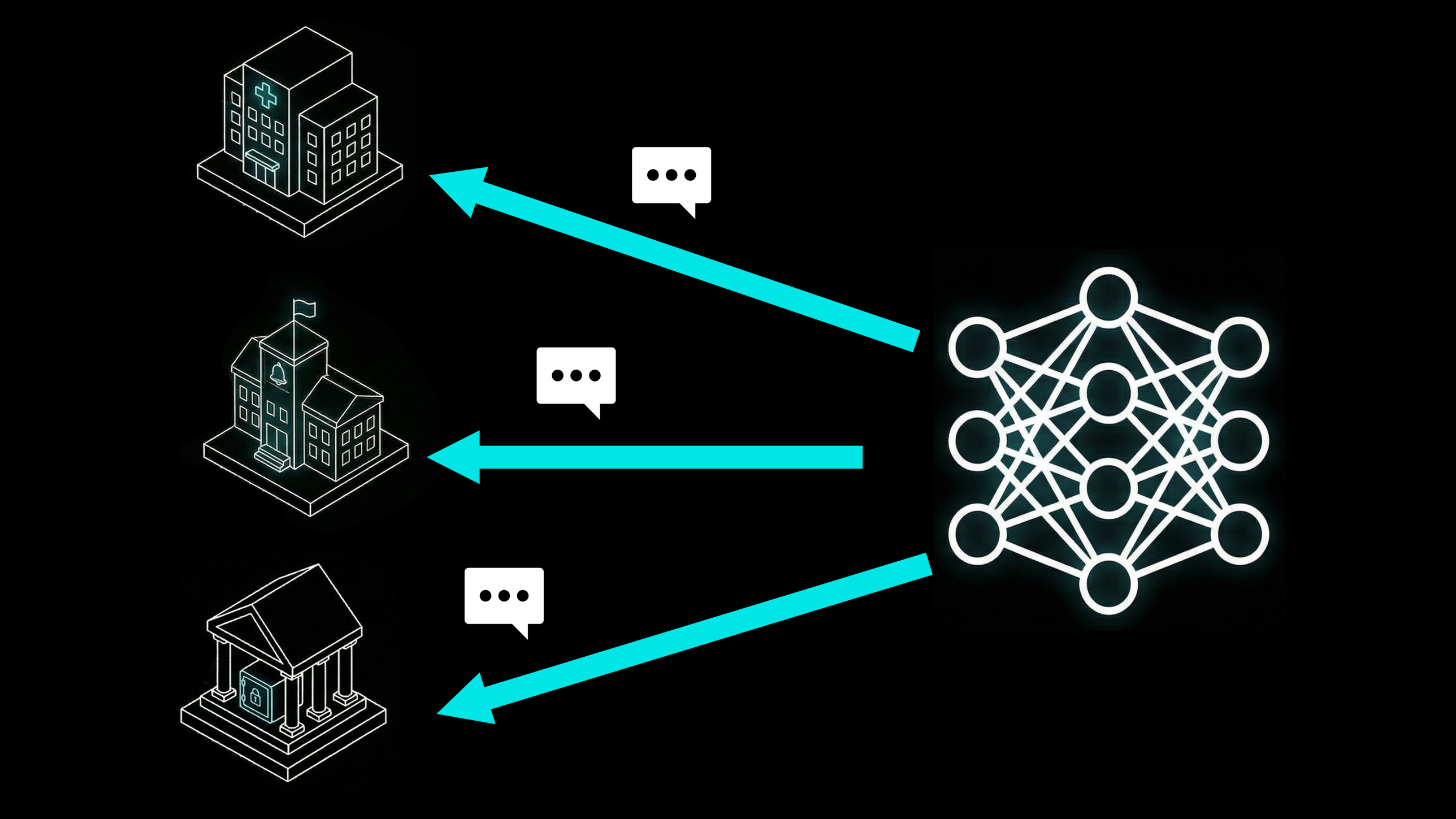
Security

Compliance

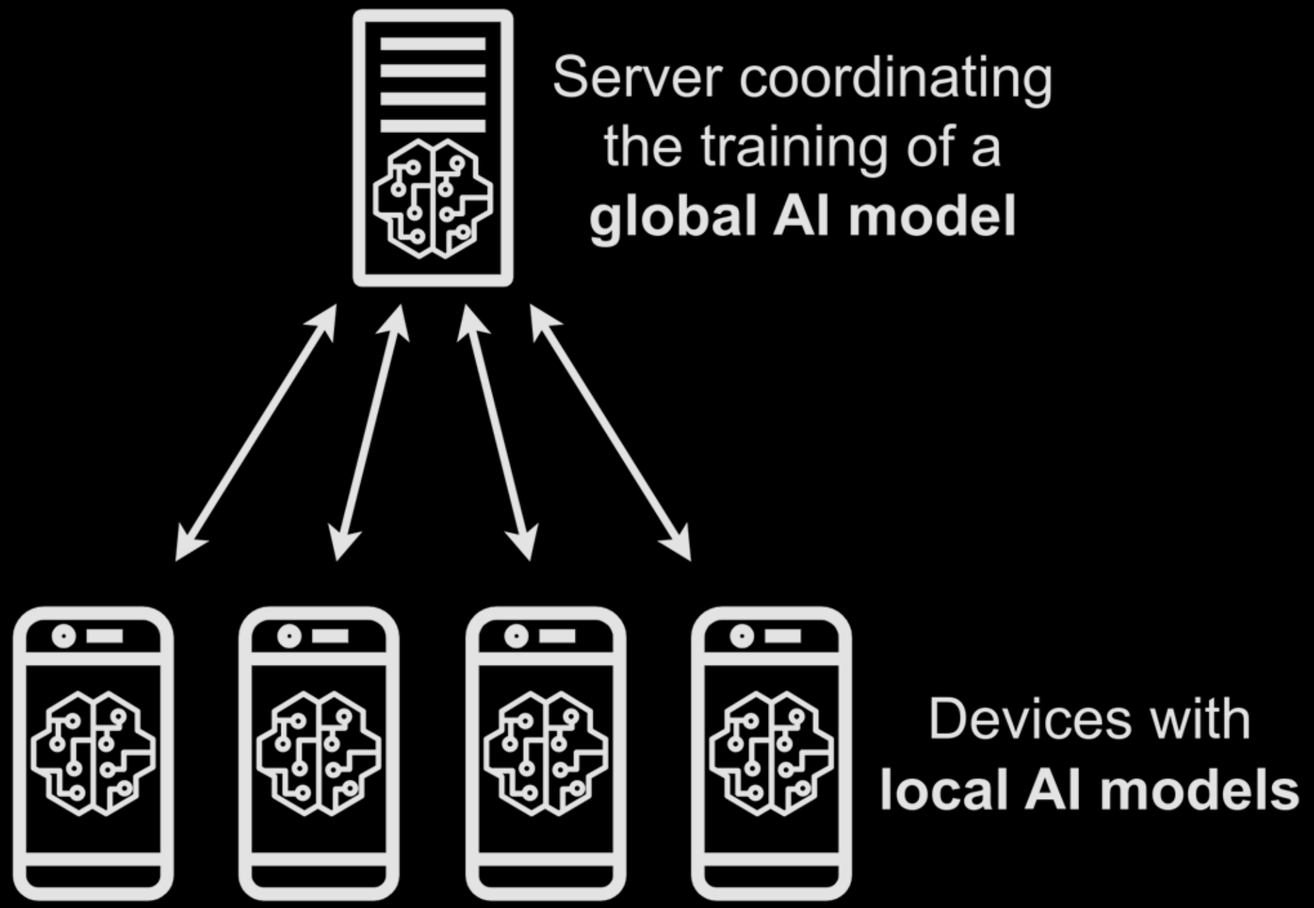
Data gravity

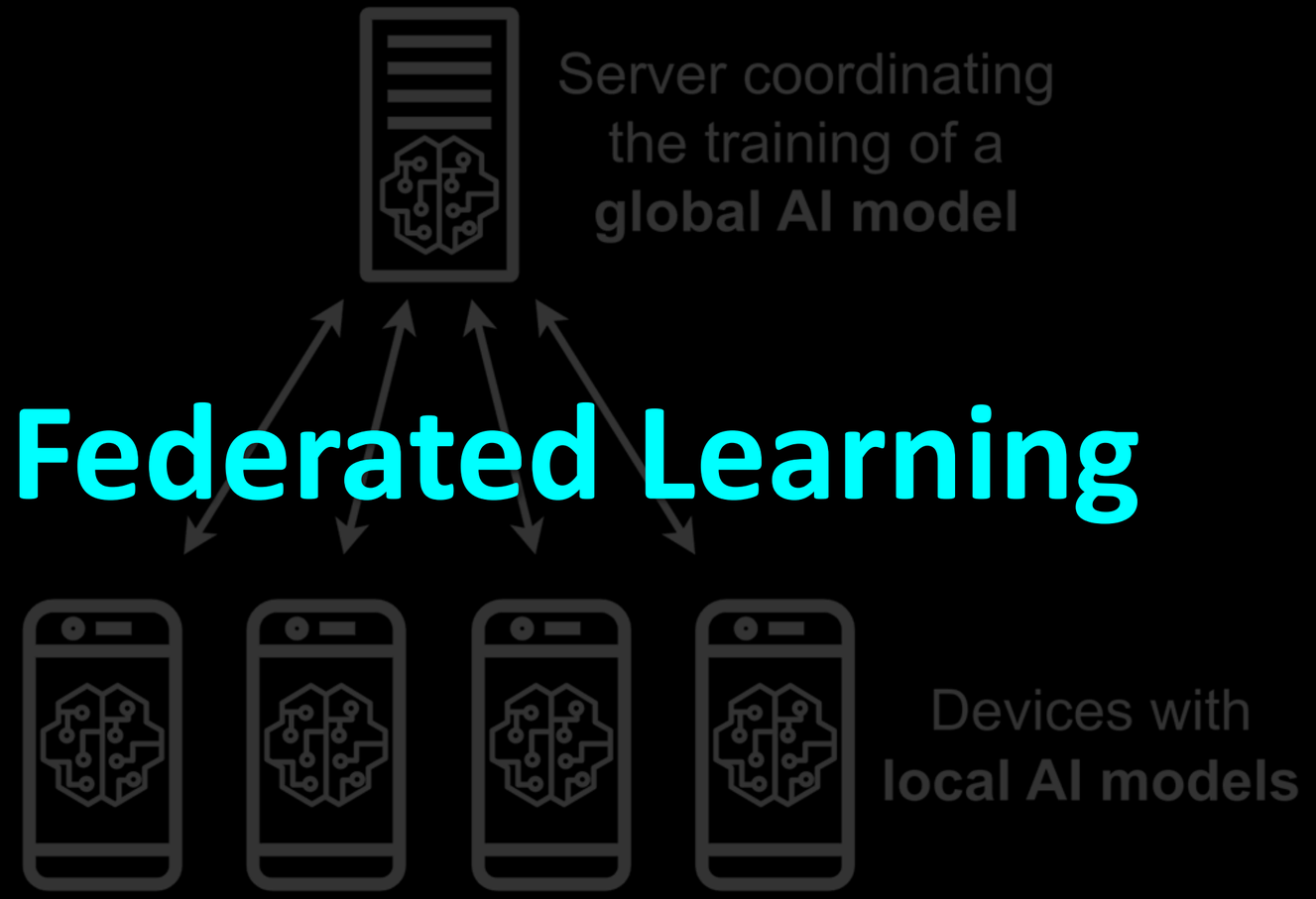
We have been bringing the data to the AI





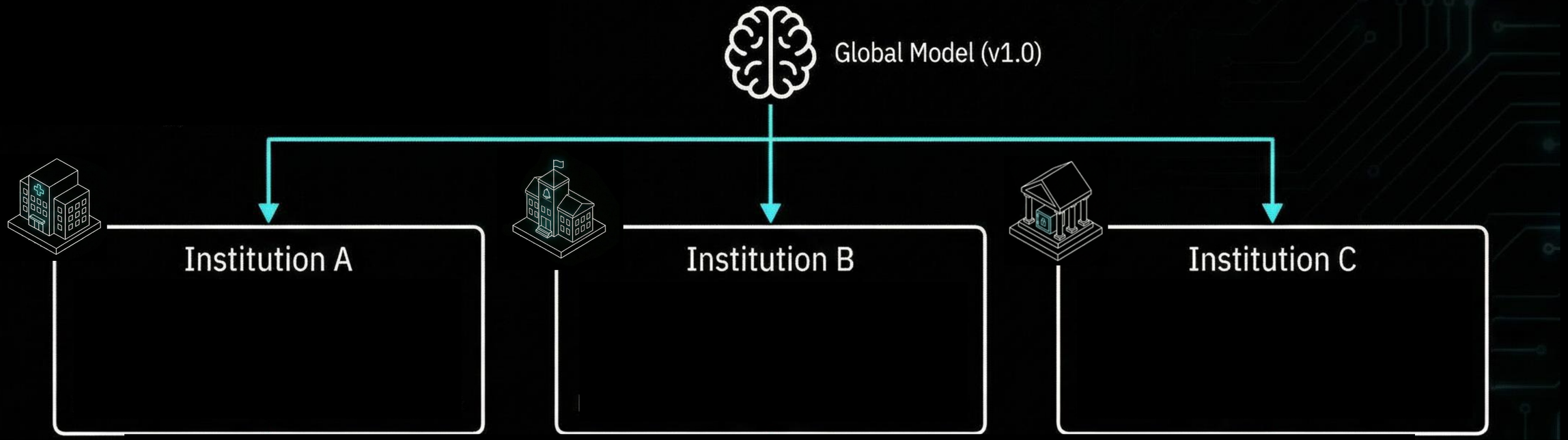
Let’s bring the AI to the data


Server coordinating
the training of a
global AI model
Devices with
local AI models

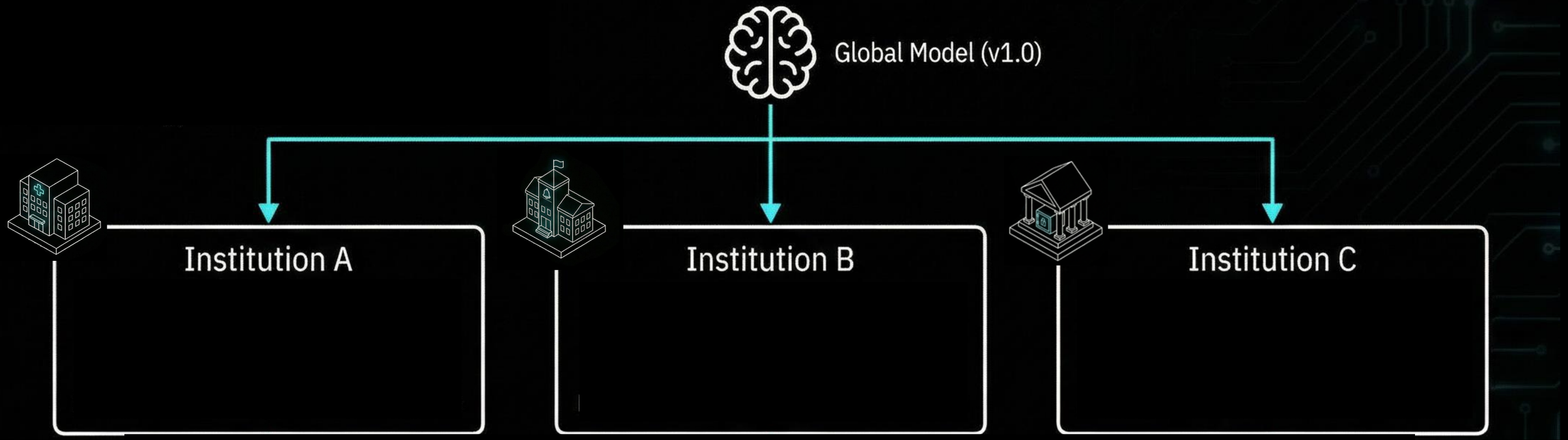


Federated Learning

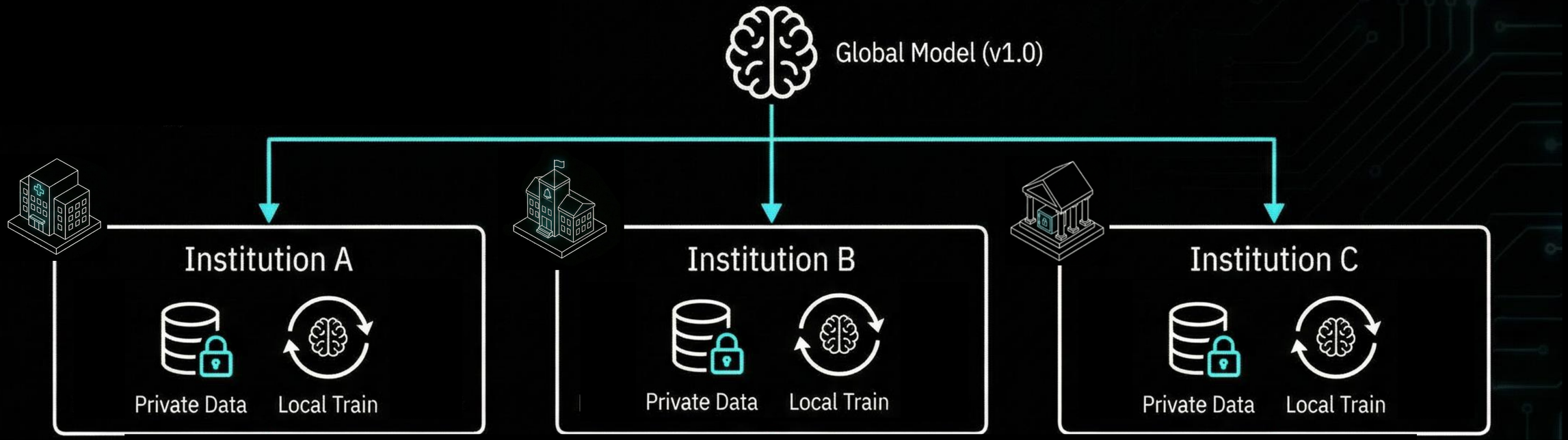
1. Initialize



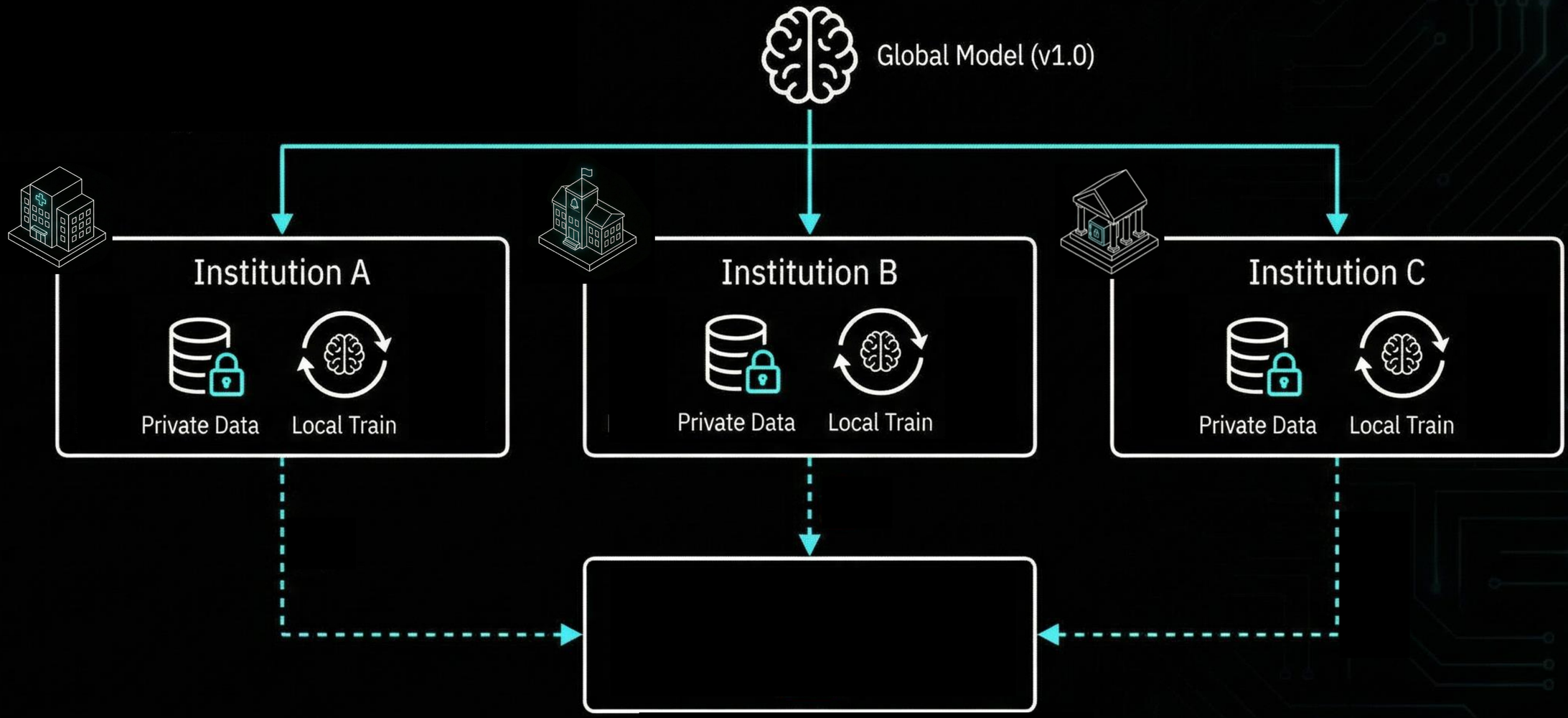
1. Initialize



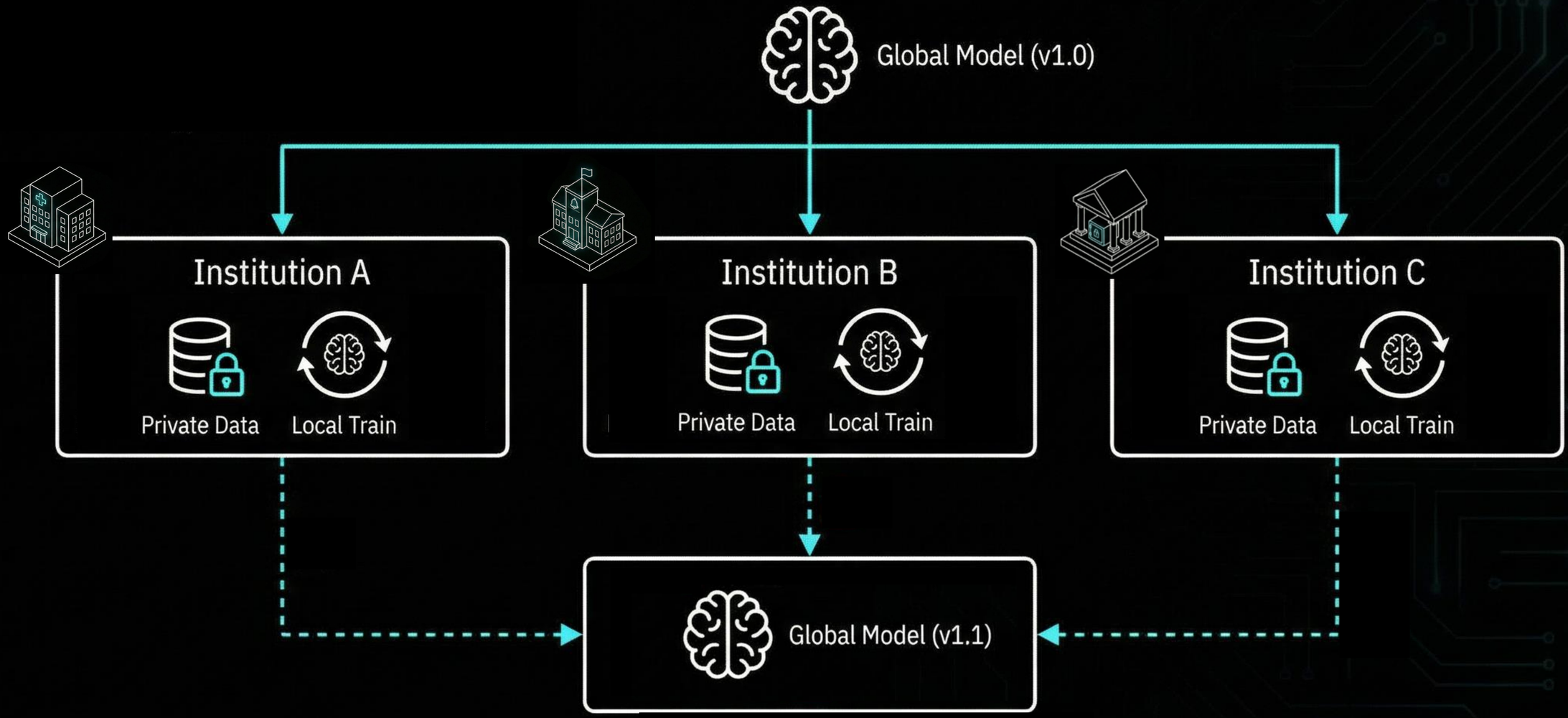
2. Train locally



3. Share updates



4. Aggregate


Global Model (v1.0)
Institution A
Private Data
Local Train
Institution B
Private Data
Local Train
Institution C
Private Data
Local Train
Global Model (v1.1)

It works!

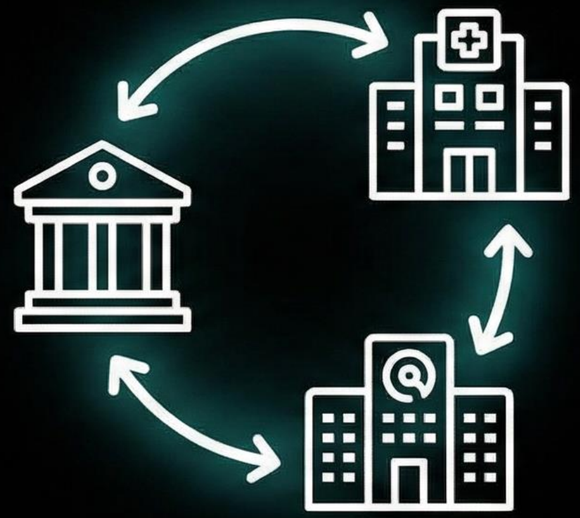
Advantages

Advantages

Locality

Advantages

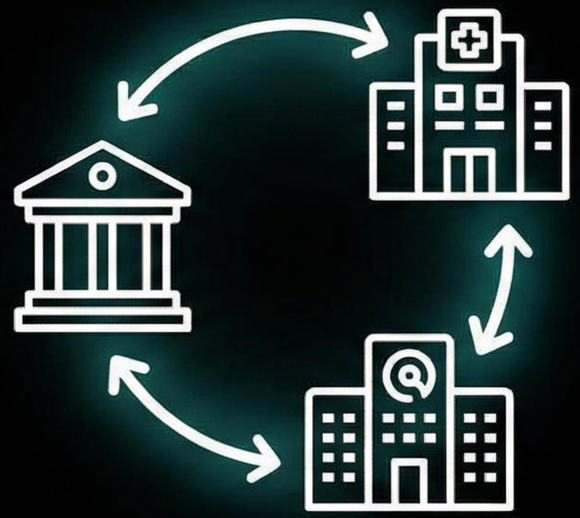
Locality



Collaboration

Advantages

Locality



Collaboration

Control

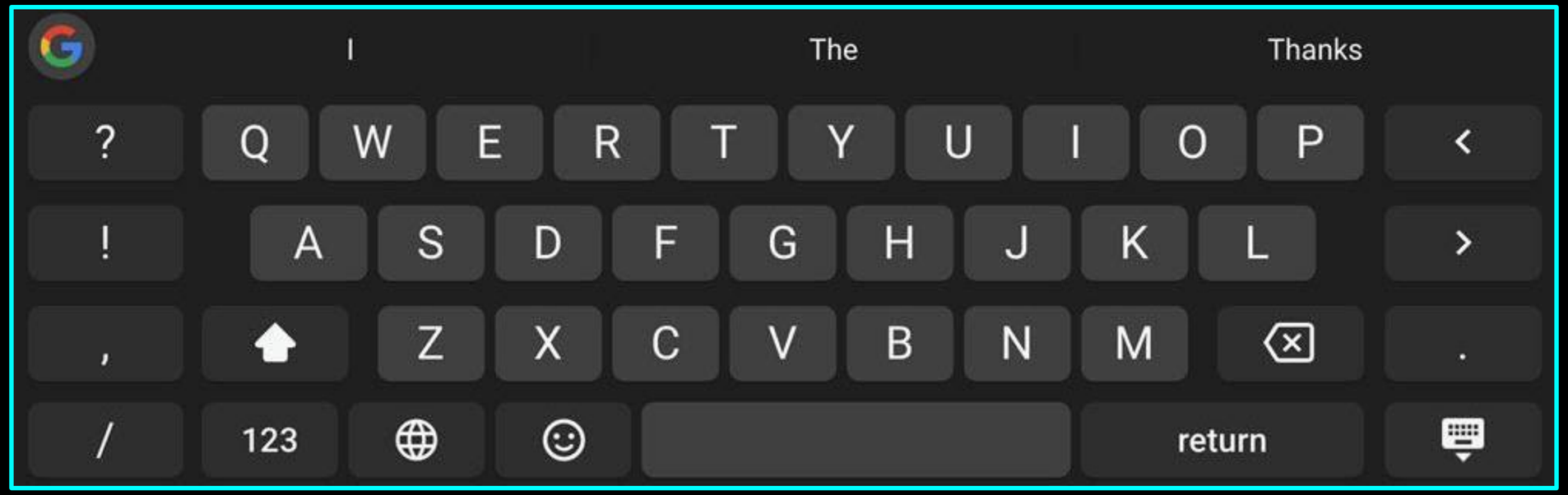
Advances in private training for production on-device language models

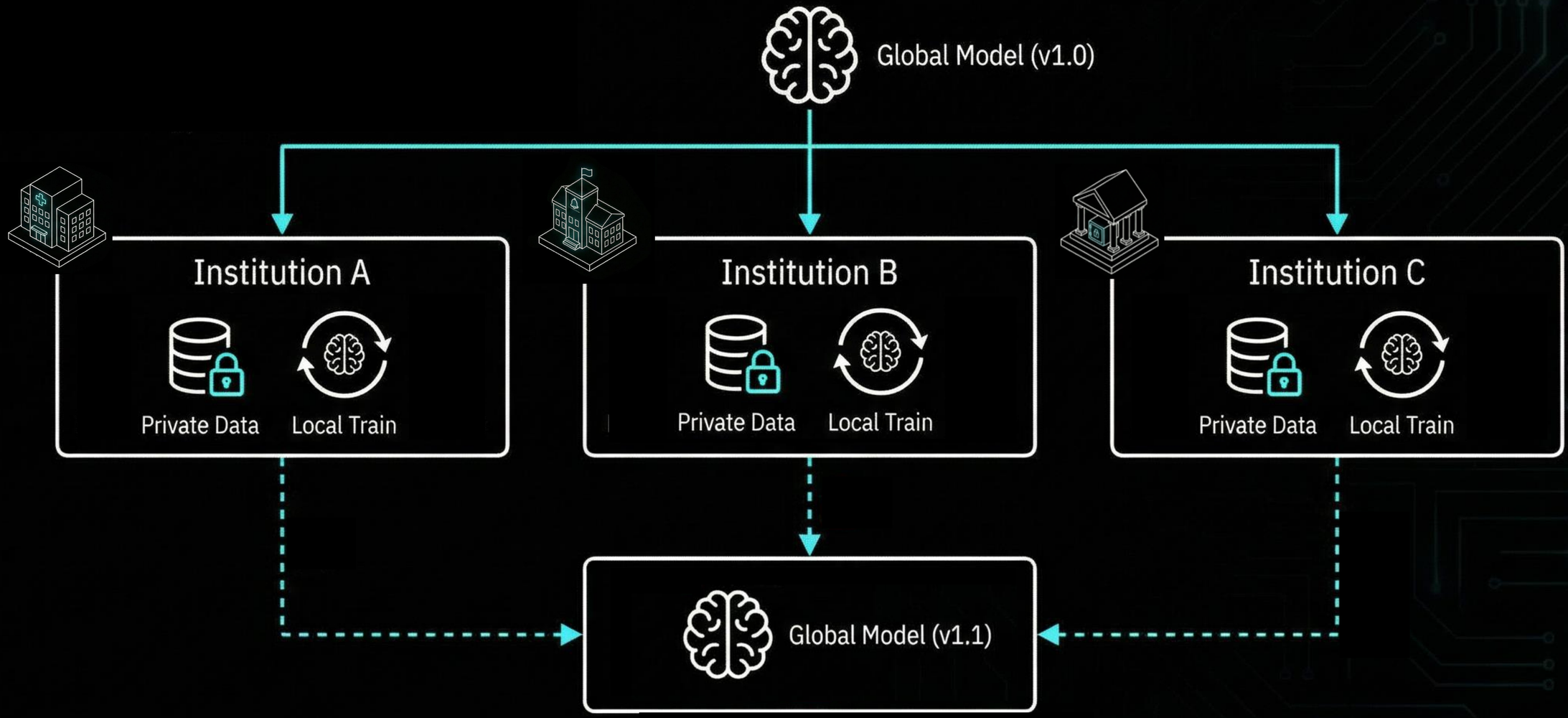
February 21, 2024 ·

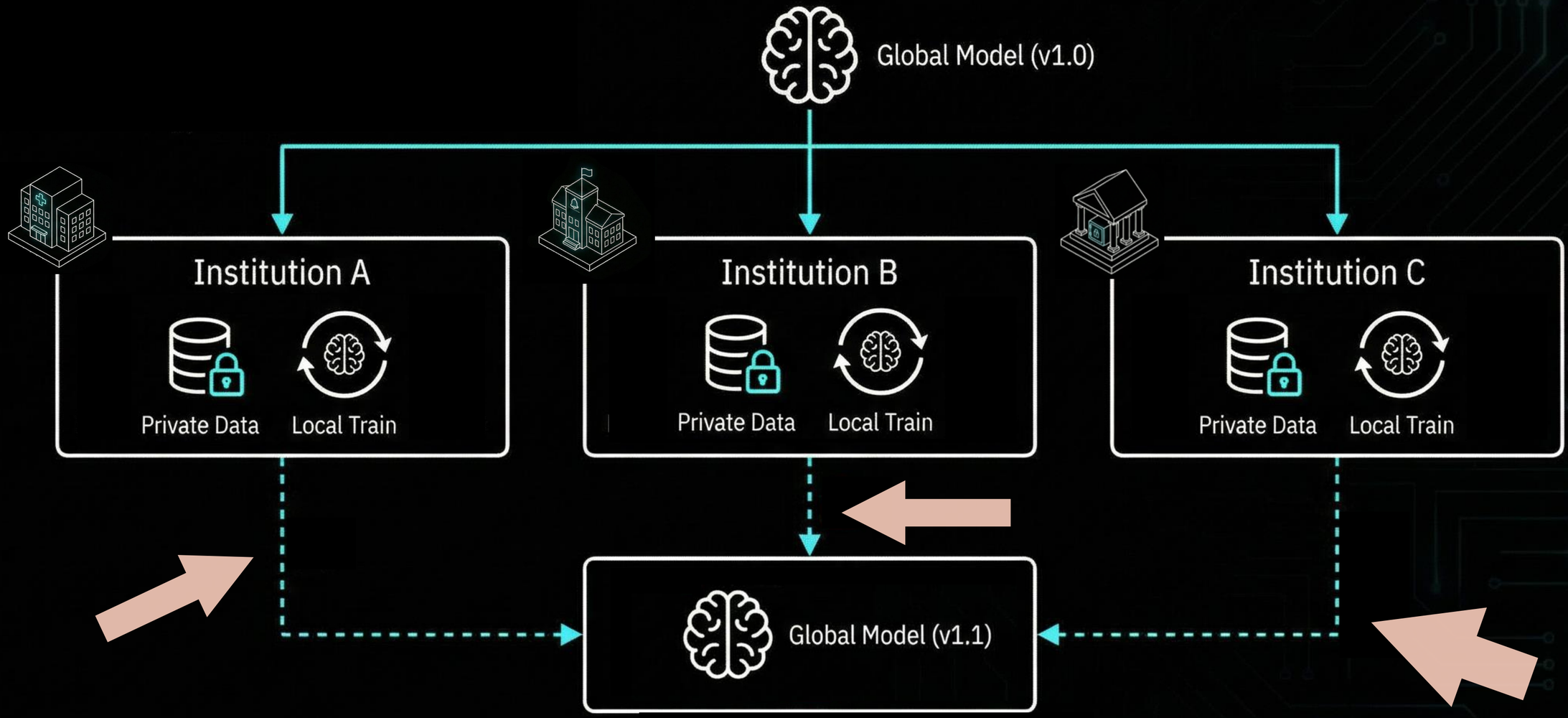
Posted by Zheng Xu, Research Scientist, and Yanxiang Zhang, Software Engineer, Google

https://research.google/blog/advances-in-private-training-for-production-on-device-language-models/

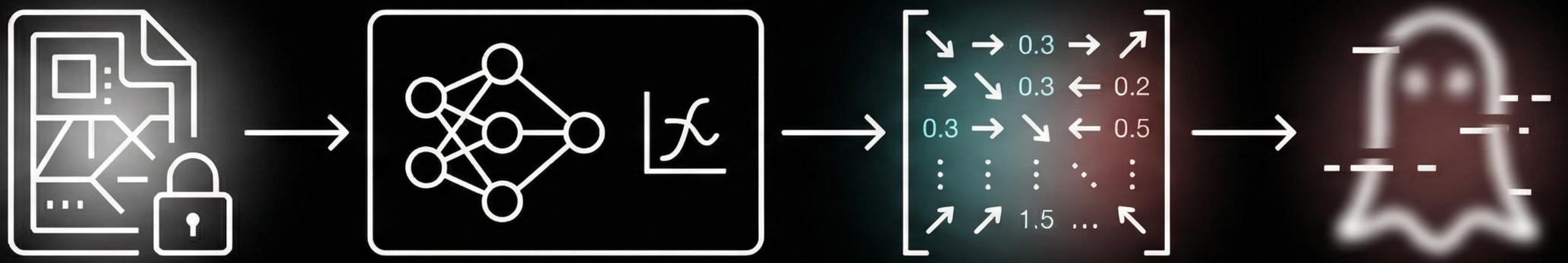
Gboard




Global Model (v1.0)
Institution A
Private Data
Local Train
Institution B
Private Data
Local Train
Institution C
Private Data
Local Train
Global Model (v1.1)

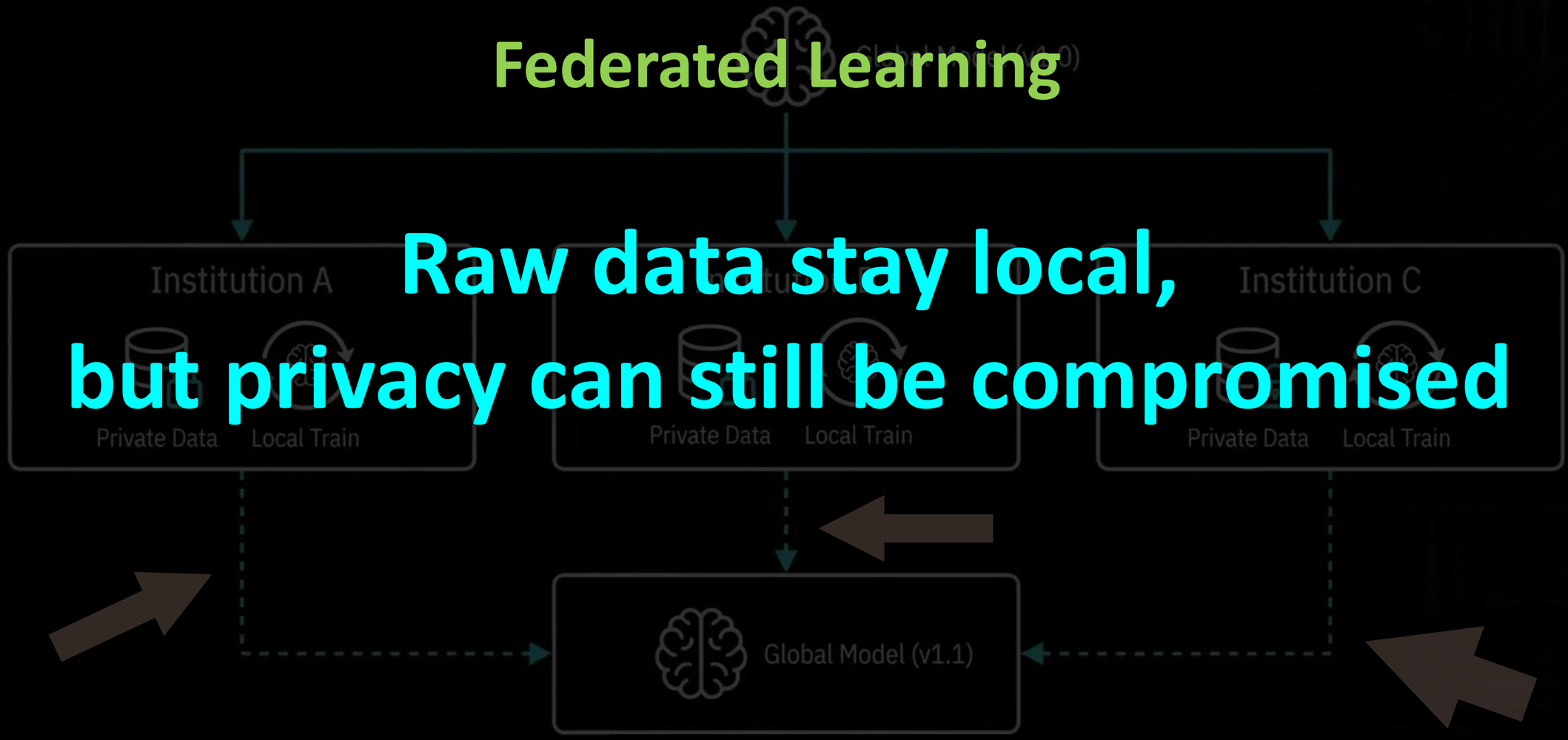

Global Model (v1.0)
Institution A
Private Data
Local Train
Institution B
Private Data
Local Train
Institution C
Private Data
Local Train
Global Model (v1.1)

Gradients can reveal data



Zhu et al., “Deep Leakage from Gradients,” NeurIPS 2019

Federated Learning



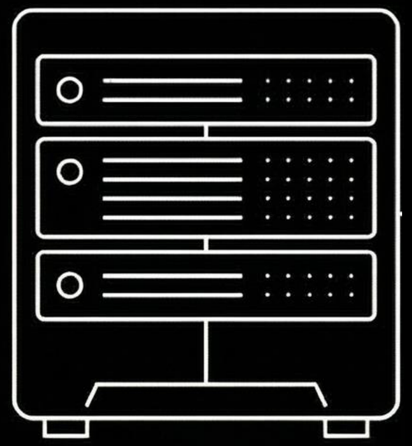
Raw data stay local,
but privacy can still be compromised

Solution?

Homomorphic Encryption

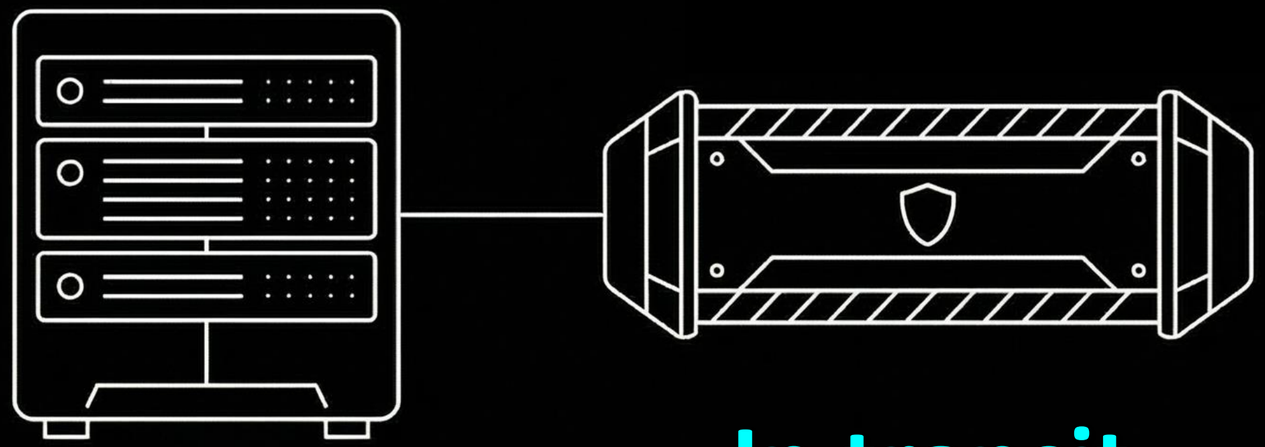
Data States

Data States



At rest

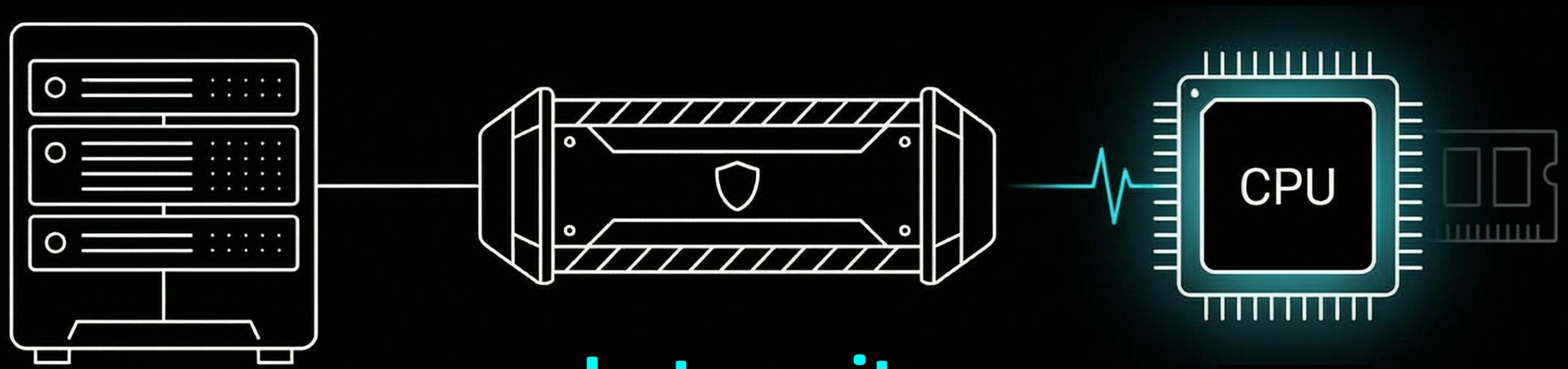
Data States



In transit

At rest

Data States

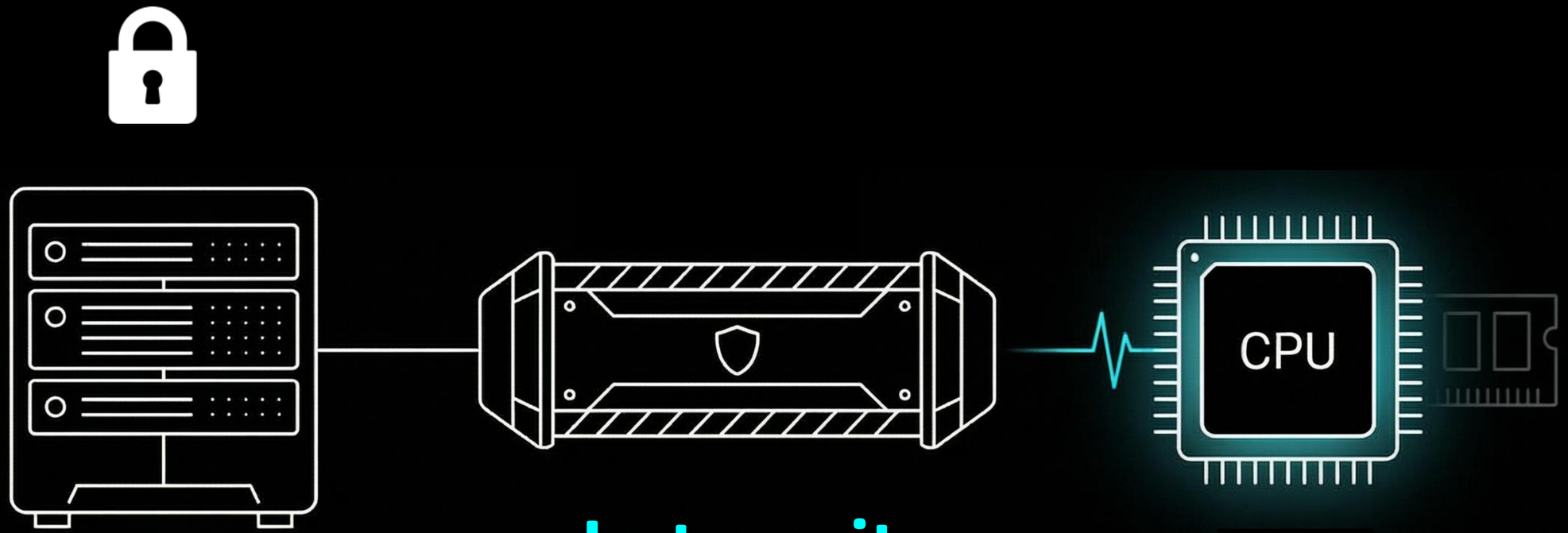


At rest

In transit

In use

Data States

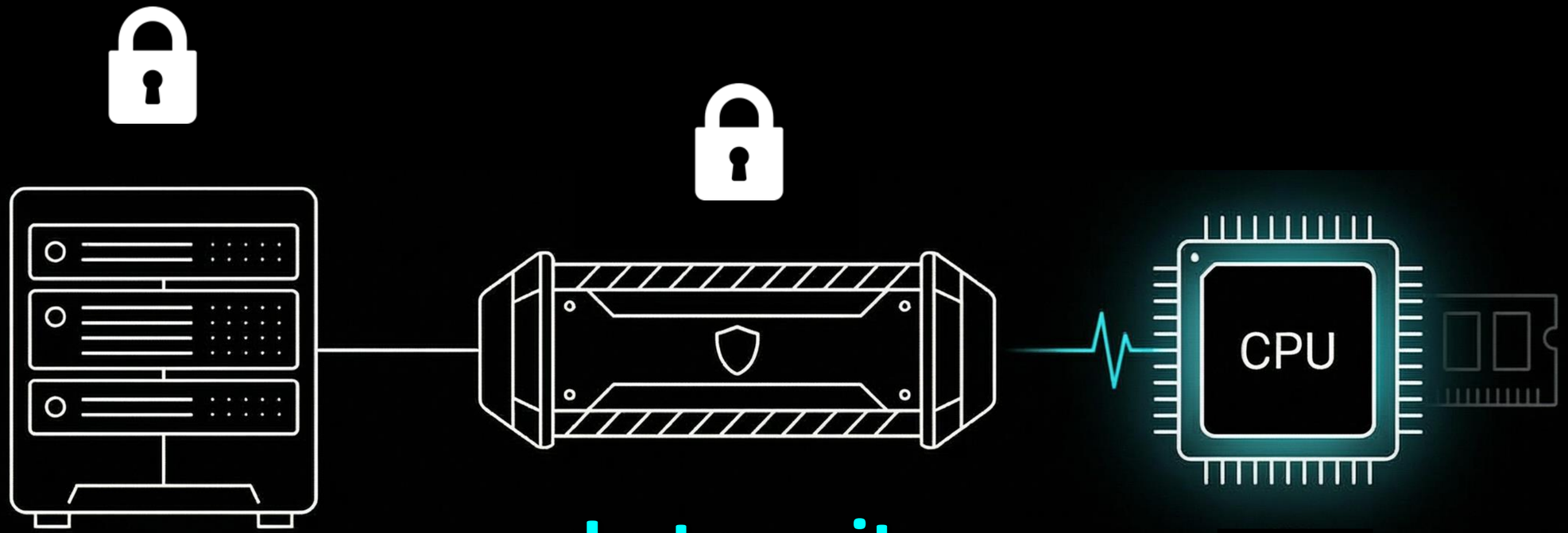


At rest

In transit

In use

Data States

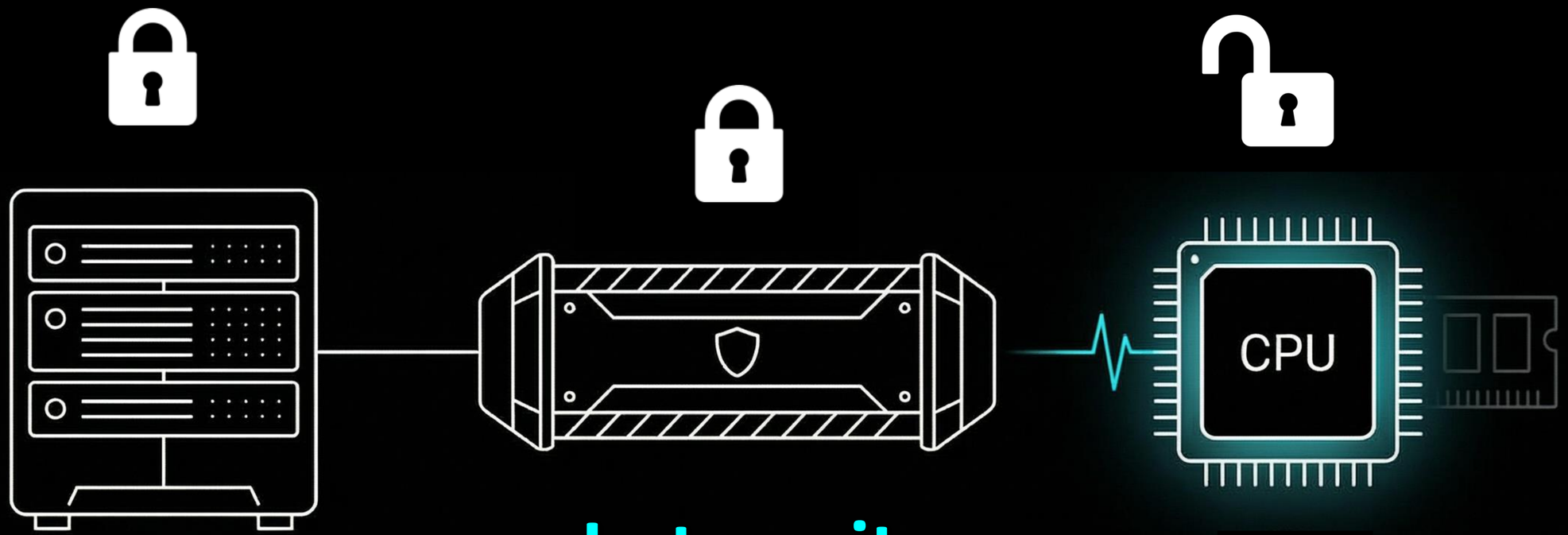


At rest

In transit

In use

Data States

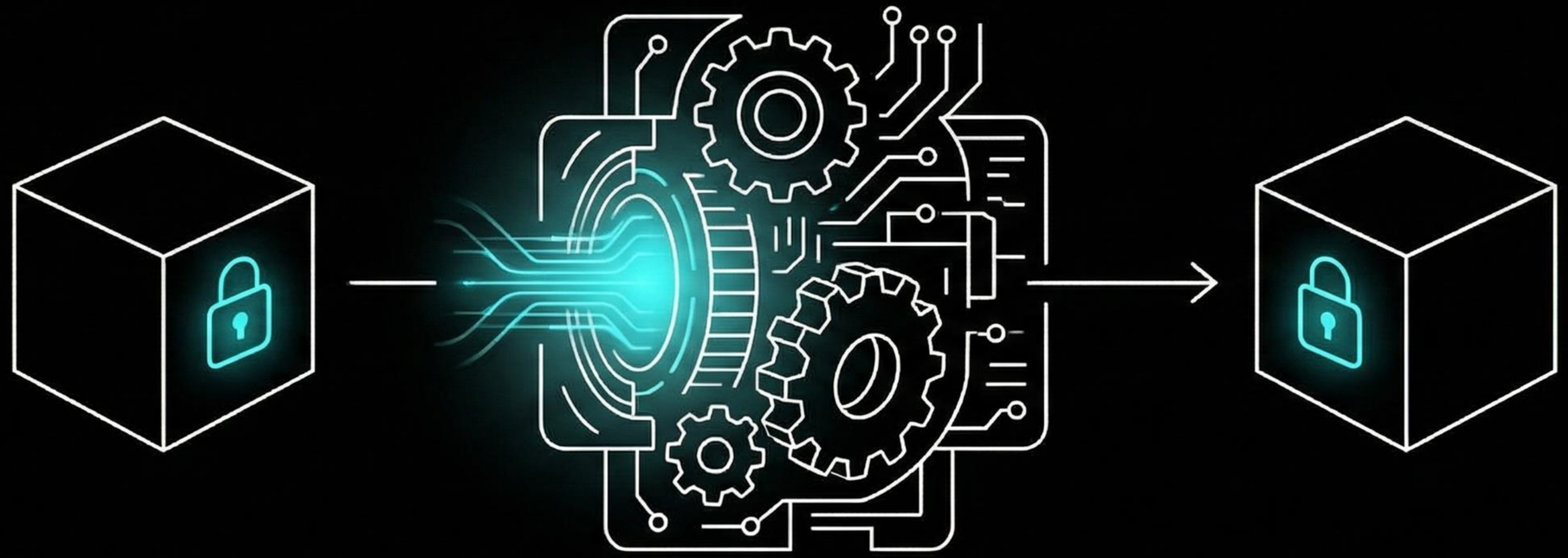


At rest

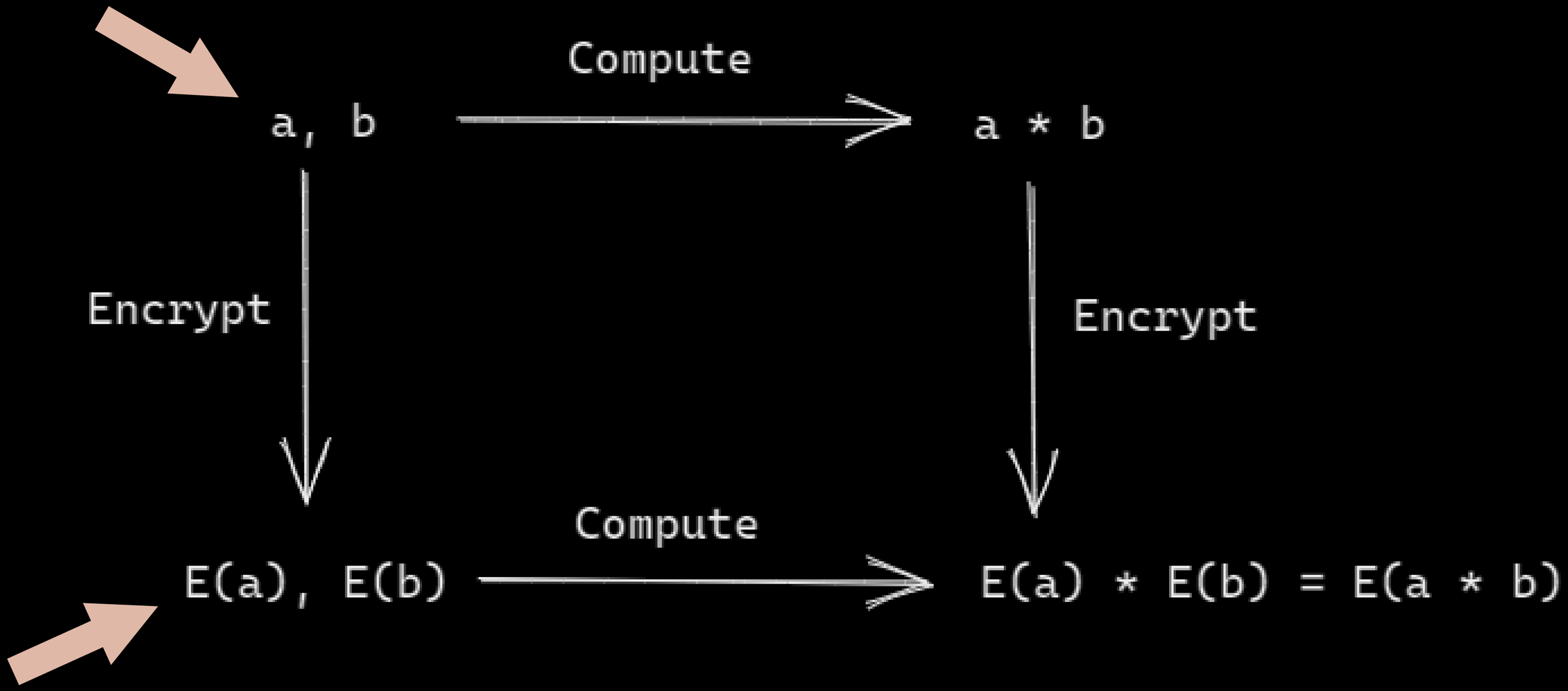
In transit

In use

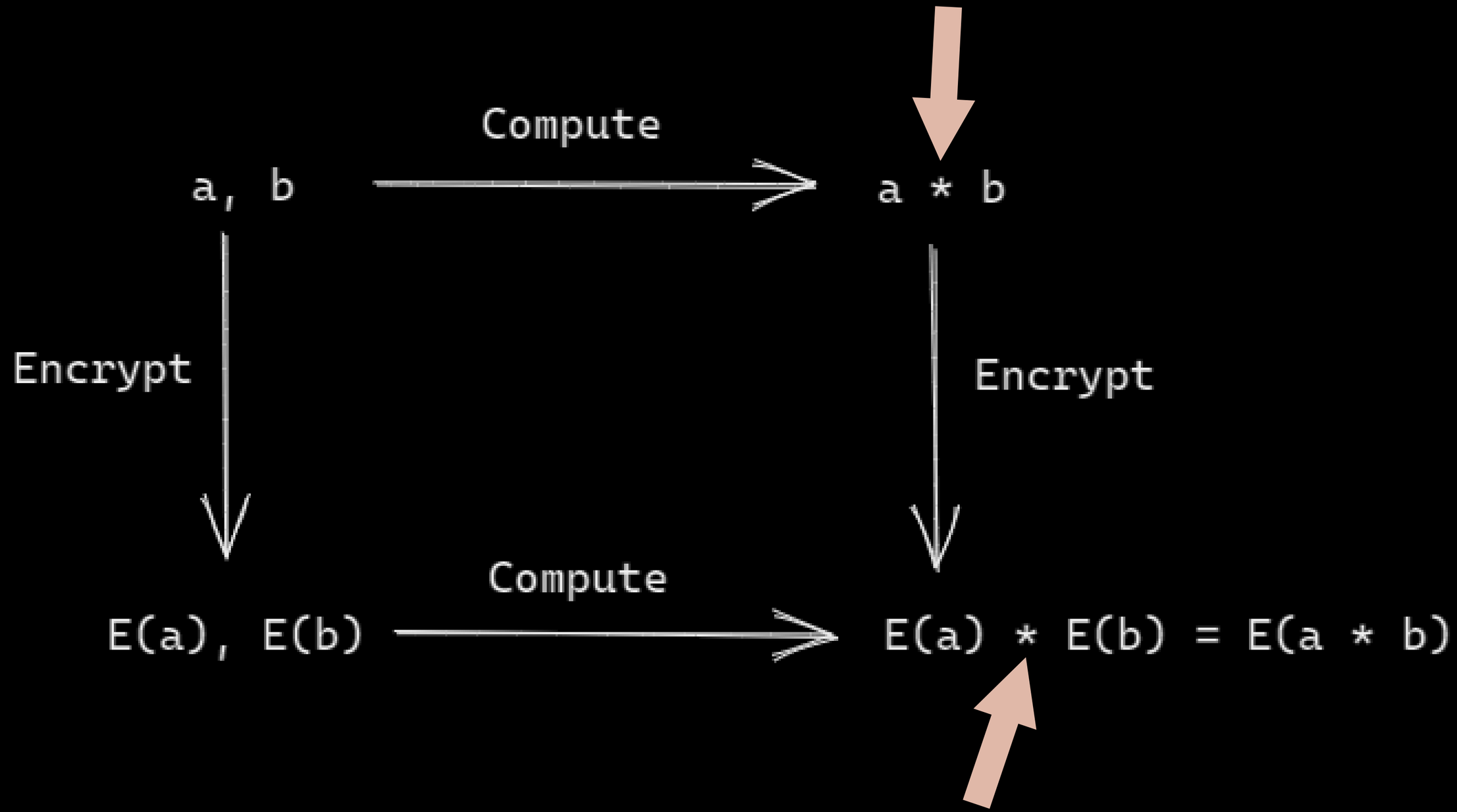
Homomorphic Encryption



Computing on encrypted data



https://openmined.org/blog/private-ai-machine-learning-on-encrypted-data



https://openmined.org/blog/private-ai-machine-learning-on-encrypted-data

Demo

What I did:

Enc(20) → 30

Enc(22) → 32

Server computes:

30 + 32 = 62

I decrypt:

62 − 20 = 42

What I did:

Enc(20) → 30

Enc(22) → 32

Server computes:

30 + 32 = 62

I decrypt:

62 – 20 = 42

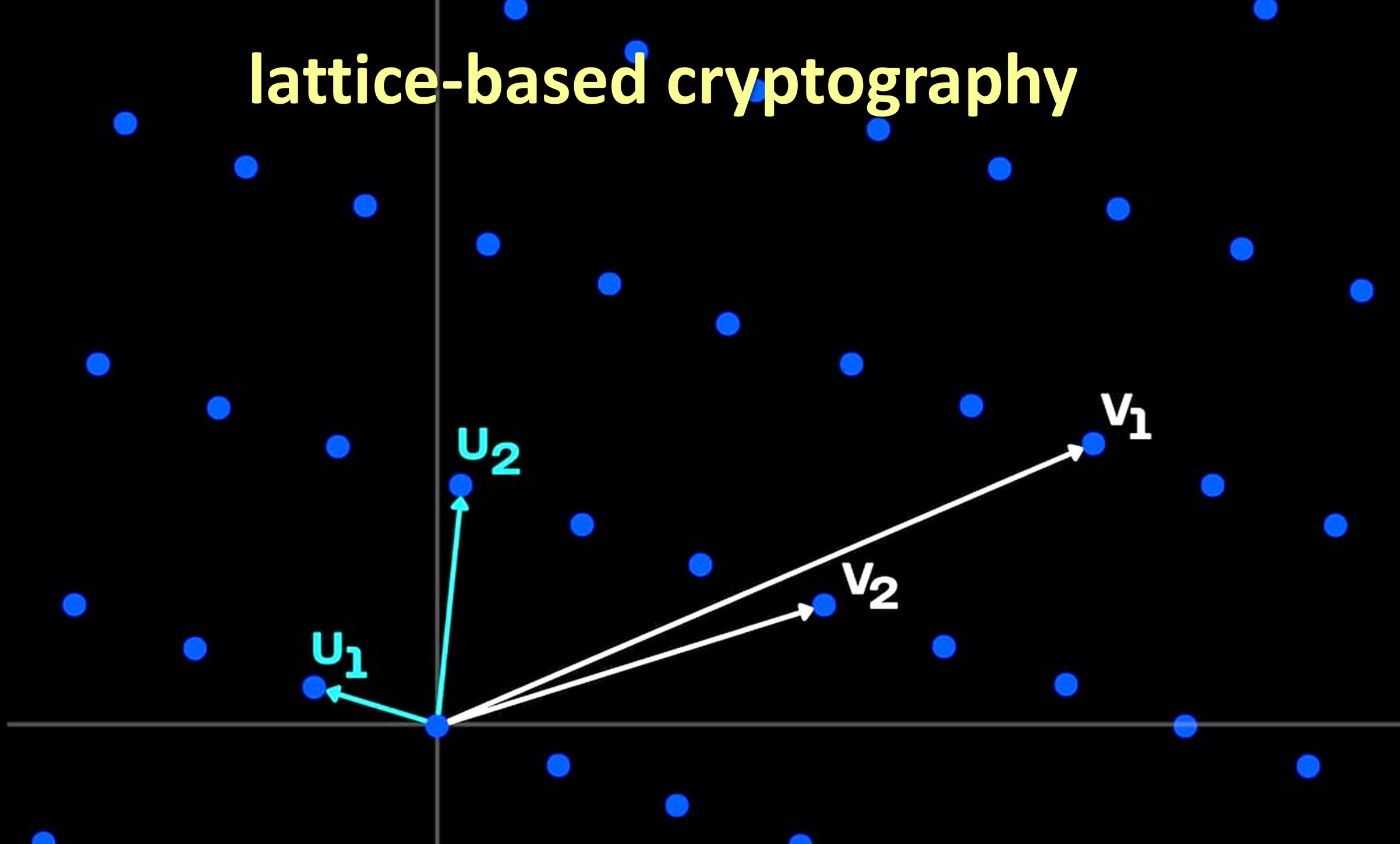
What this means:

The server computed on encoded values without seeing the original numbers.

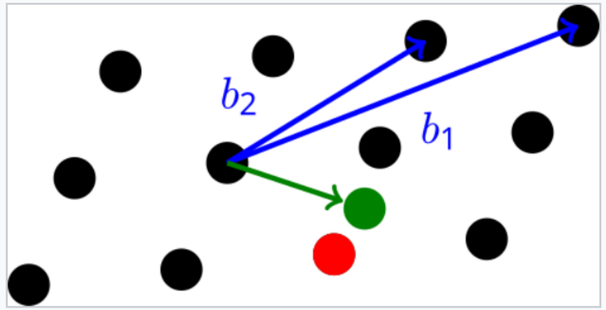
Simple working example:

Dec(Enc(a) + Enc(b)) = a + b

lattice-based cryptography



Closest vector problem (CVP) [edit]



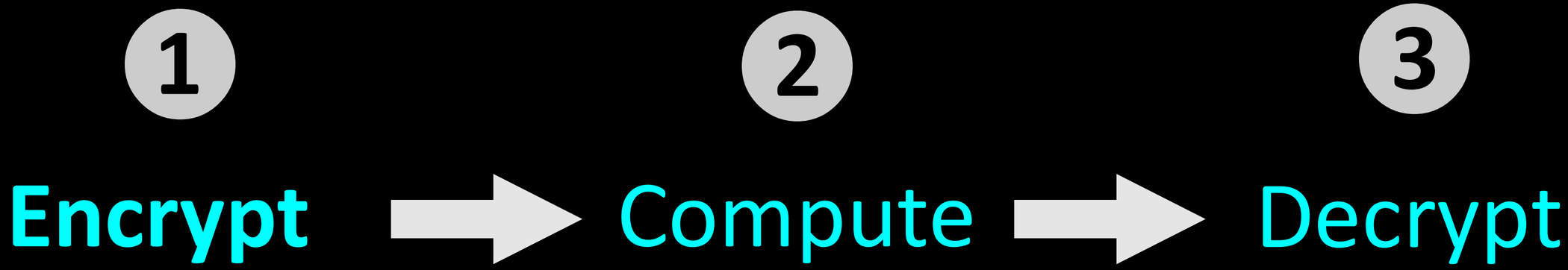
This is an illustration of the closest vector problem (basis vectors in blue, external vector in green, closest vector in red).

In CVP, a basis of a vector space *V* and a metric *M* (often L^2) are given for a lattice *L*, as well as a vector *v* in *V* but not necessarily in *L*. It is desired to find the vector in *L* closest to *v* (as measured by *M*). In the γ-approximation version CVP_γ, one must find a lattice vector at distance at most γ.

Relationship with SVP [edit]

The closest vector problem is a generalization of the shortest vector problem. It is easy to show that given an oracle for CVP_γ (defined below), one can solve SVP_γ by making some queries to the oracle.[21] The naive method to find the shortest vector by calling the CVP_γ oracle to find the closest vector to $\mathbf{0}$ does not work because $\mathbf{0}$ is itself a lattice vector and the algorithm could potentially output $\mathbf{0}$.

The reduction from SVP_γ to CVP_γ is as follows: Suppose that the input to the SVP_γ is the basis for lattice $B = [b_1, b_2, \ldots, b_n]$. Consider the basis $B^i = [b_1, \ldots, 2b_i, \ldots, b_n]$ and let x_i be the vector returned by $\text{CVP}_\gamma(B^i, b_i)$. The claim is that the shortest vector in the set $\{x_i - b_i\}$ is the shortest vector in the given lattice.


1
2
3
Encrypt
Compute
Decrypt

vitalik.eth
@VitalikButerin

Prediction:

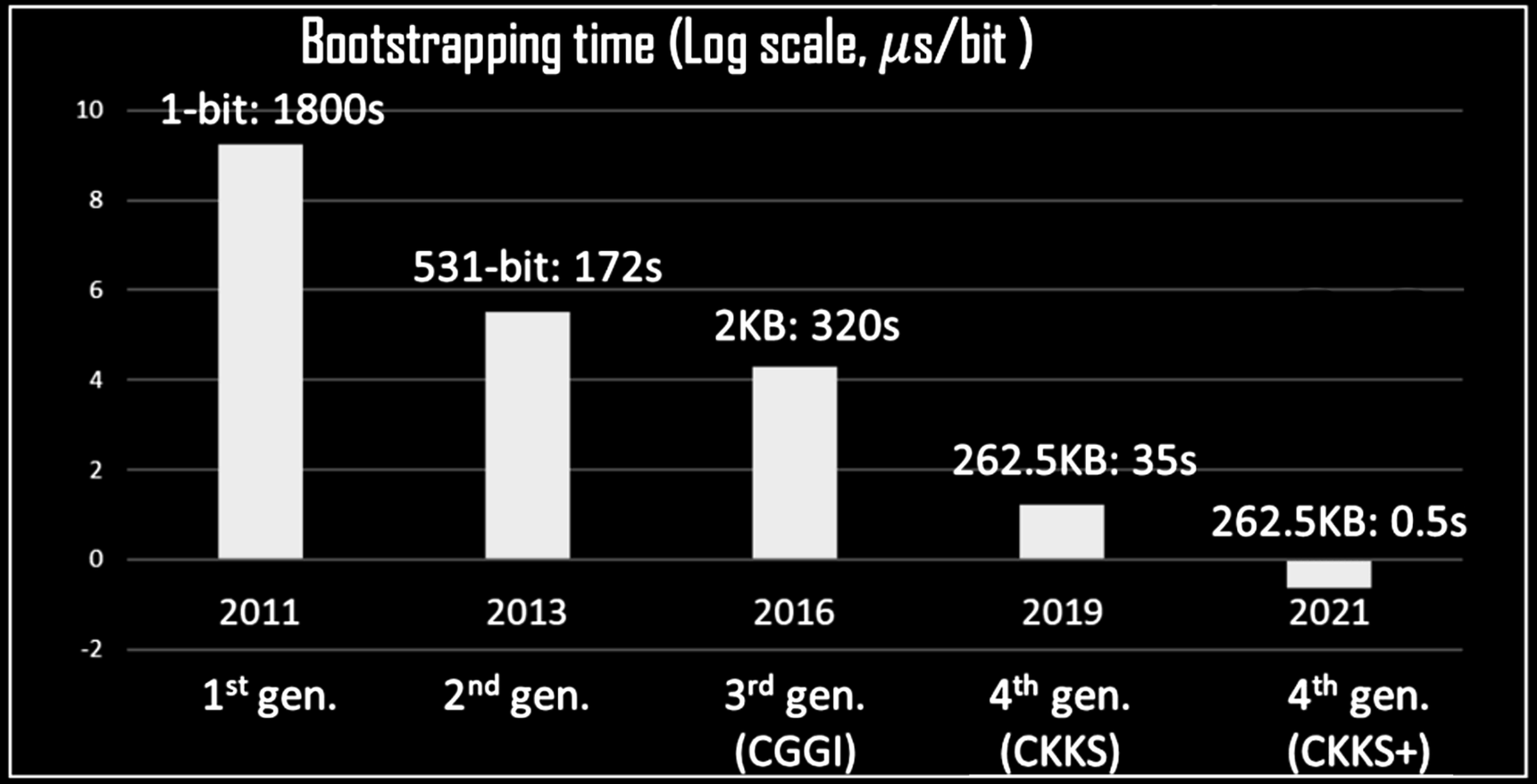
The megatrend in cryptography of the 2010s was elliptic curves, pairings and general purpose ZKPs/SNARKs.

The megatrend of the 2020s will be (in addition to broad adoption of the above) lattices, LWE, multilinear maps, homomorphic encryption, MPC and obfuscation.

4:08 AM · Apr 11, 2020

HE is getting 8x faster every year

e.g. Bootstrapping time: the most time-consuming operation in HE



Homomorphic Encryption for Large Integers from Nested Residue Number Systems

Dan Boneh and Jaehyung Kim

Stanford University

dabo@cs.stanford.edu, jaehk@stanford.edu

June 10, 2025

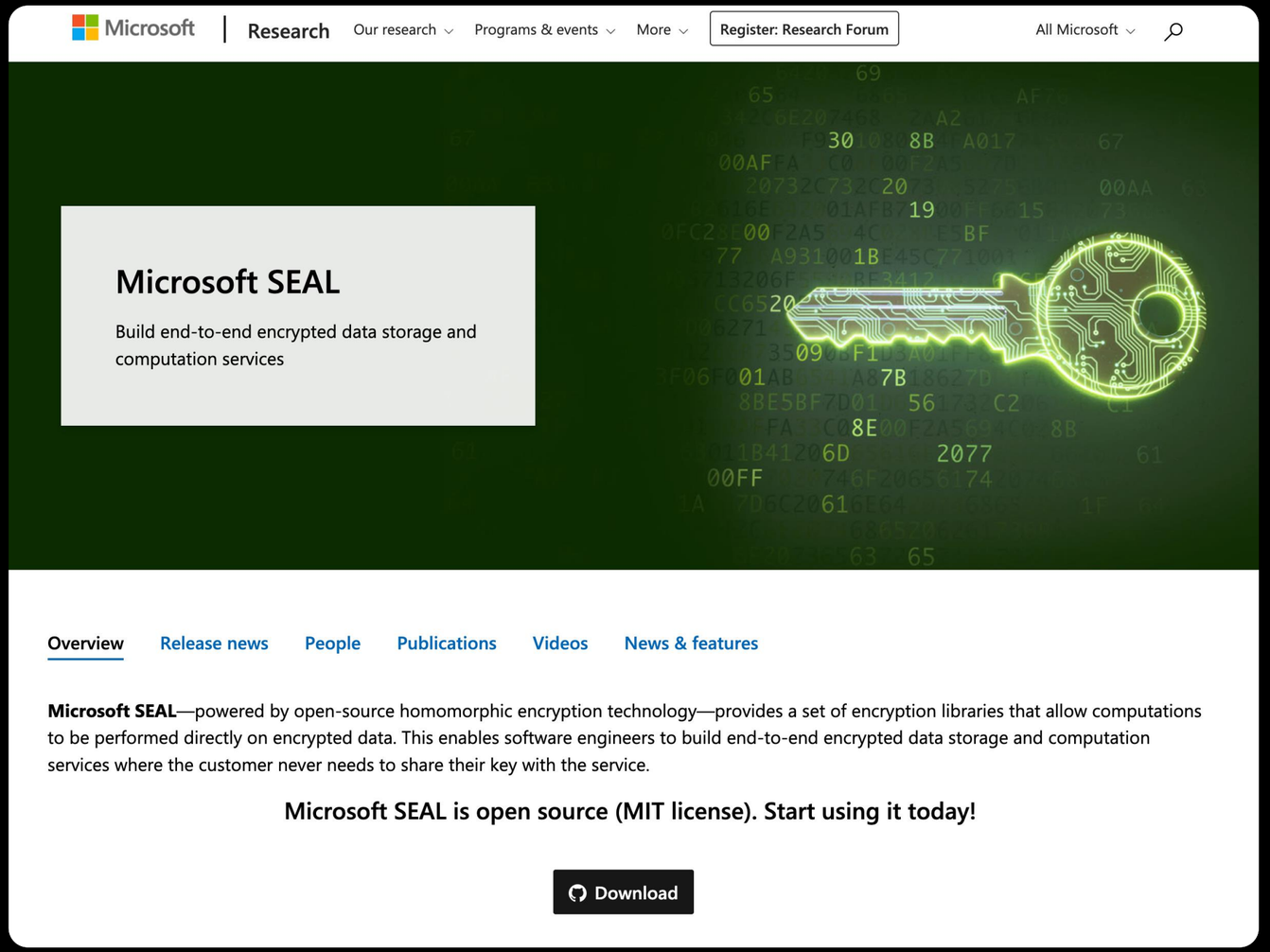
Abstract

Existing fully homomorphic encryption (FHE) schemes primarily support a plaintext space defined over a relatively small prime. However, in some important applications of FHE one needs arithmetic over a large prescribed prime. In this paper we construct a new FHE system that is specifically designed for this purpose. Our system composes three layers of residue systems to enable much better performance than was previously possible. Our experiments show that for arithmetic modulo a 256-bit integer, when compared to the TFHE-rs implementation of 256-bit arithmetic, our new system achieves a factor of two thousand better multiplication throughput and a factor of twenty better latency. Moreover, for a 2048-bit prime modulus we achieve far better performance than was previously possible.

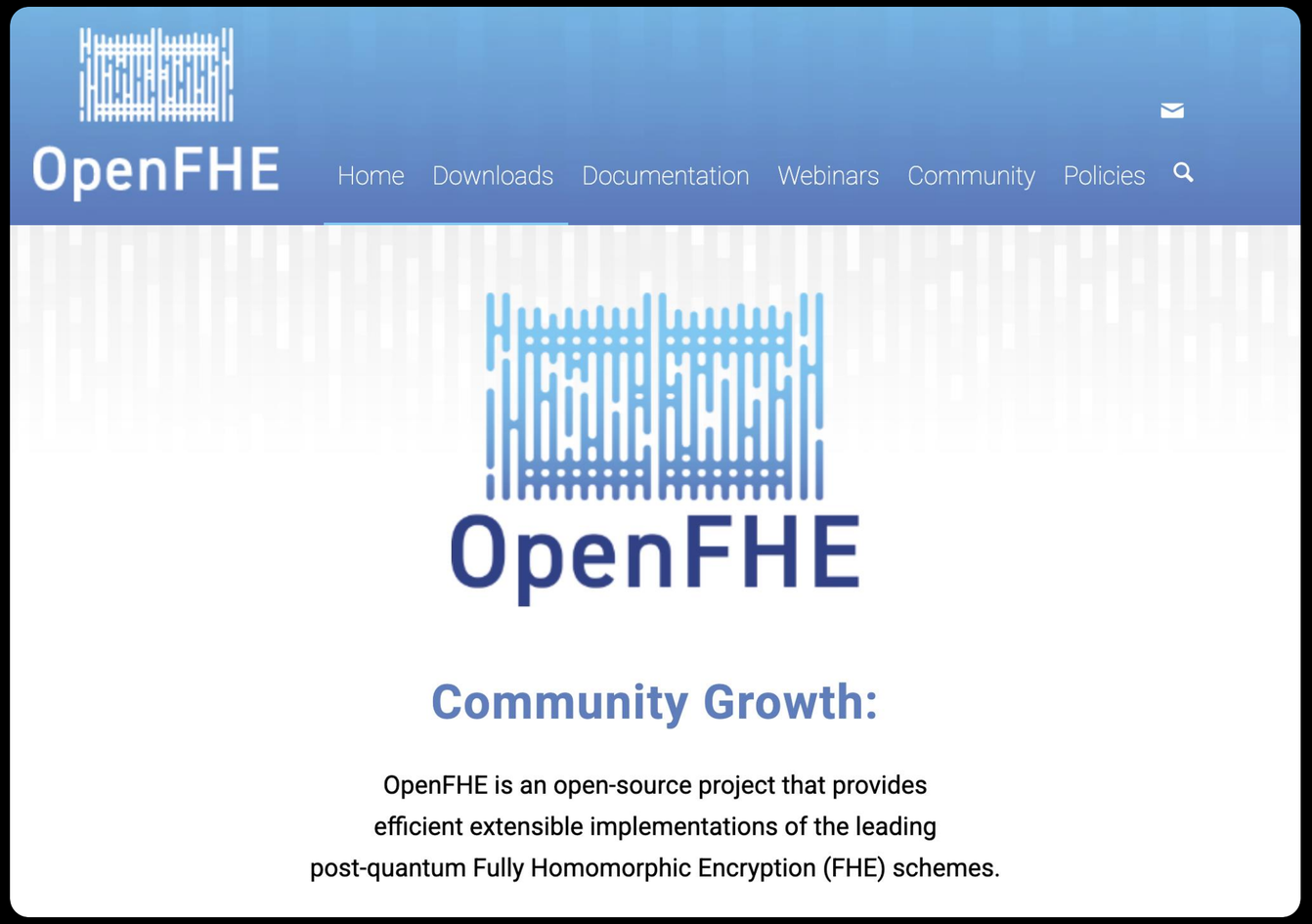
Boneh, Dan, and Jaehyung Kim. "Homomorphic encryption for large integers from nested residue number systems." Annual International Cryptology Conference. Cham: Springer Nature Switzerland, 2025.

Ecosystem

SEAL



https://www.microsoft.com/en-us/research/project/microsoft-seal



OpenFHE

https://openfhe.org

HEIR

HEIR: Homomorphic Encryption Intermediate Representation

What is HEIR?

HEIR is a compiler toolchain for fully homomorphic encryption (FHE). We aim to be the industry-standard compiler for FHE. Application developers, compiler engineers, hardware designers, and cryptography researchers can build upon HEIR to accelerate the research and development of production-strength privacy-first software systems.

Why HEIR?

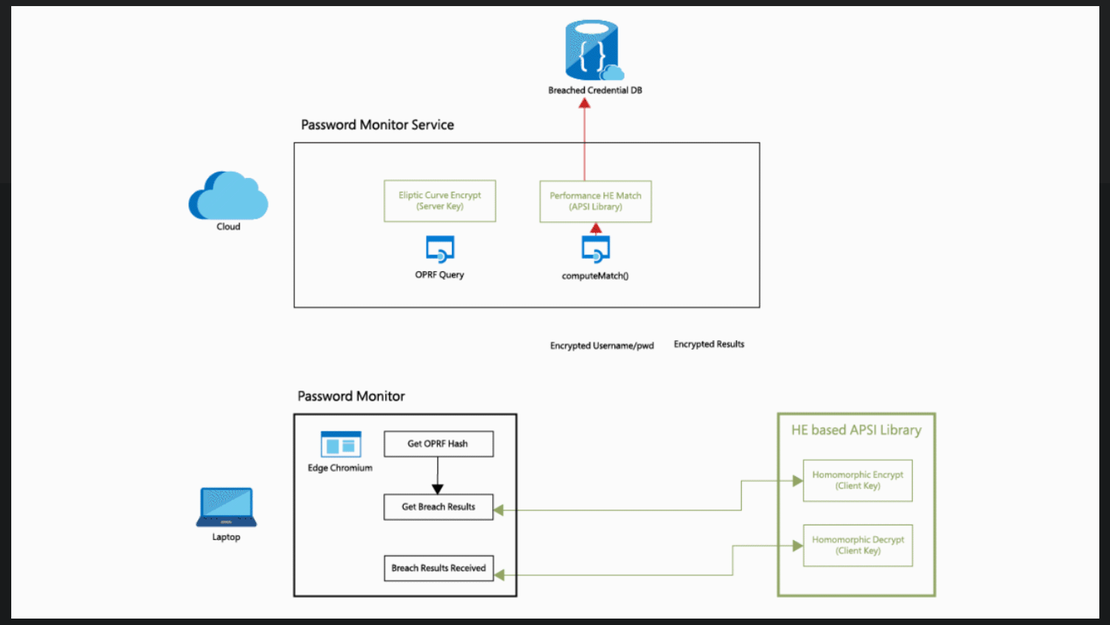
For application developers, HEIR aims to provide a simple entrypoint to start working with FHE. Write a program in Python, annotate the types to mark which are secret, and HEIR will compile the rest.

Applications

Password Monitor: Safeguarding passwords in Microsoft Edge

Published January 21, 2021

By Kristin Lauter, Principal Researcher and Partner Research Manager; Sreekanth Kannepalli, Principal Group Manager; Kim Laine, Principal Researcher; Radames Cruz Moreno, Senior Research SDE

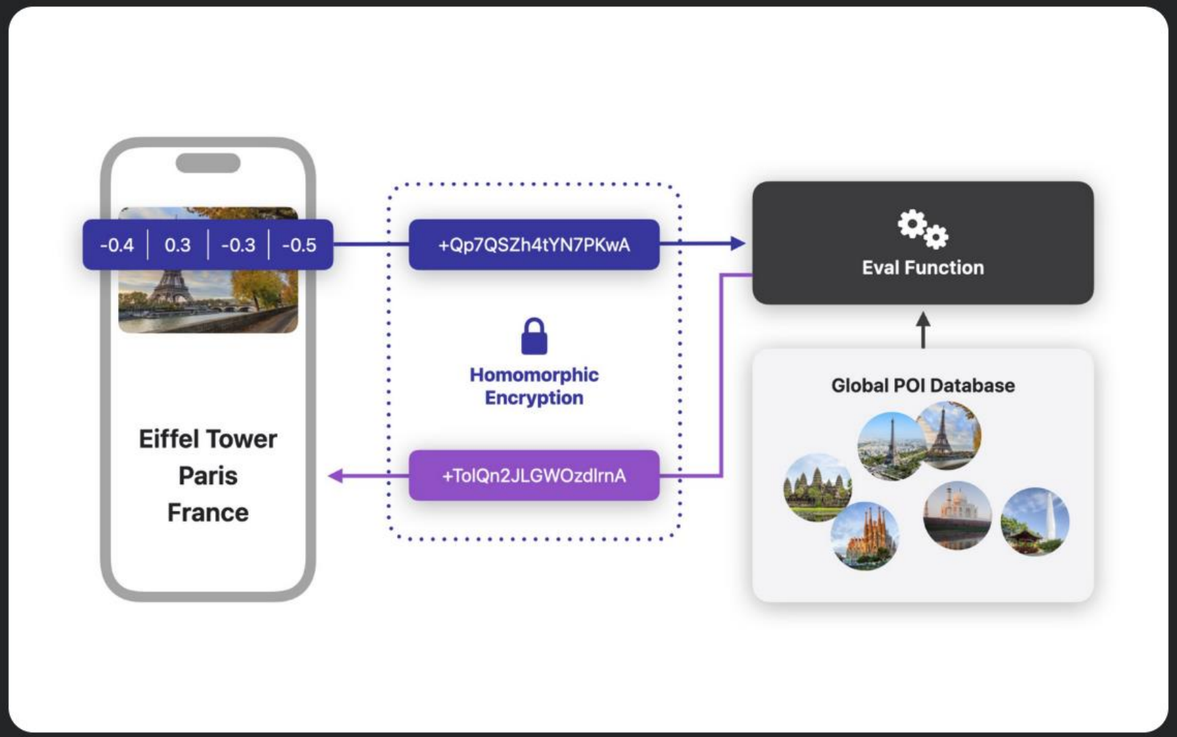


https://www.microsoft.com/en-us/research/blog/password-monitor-safeguarding-passwords-in-microsoft-edge

Highlight | October 24, 2024

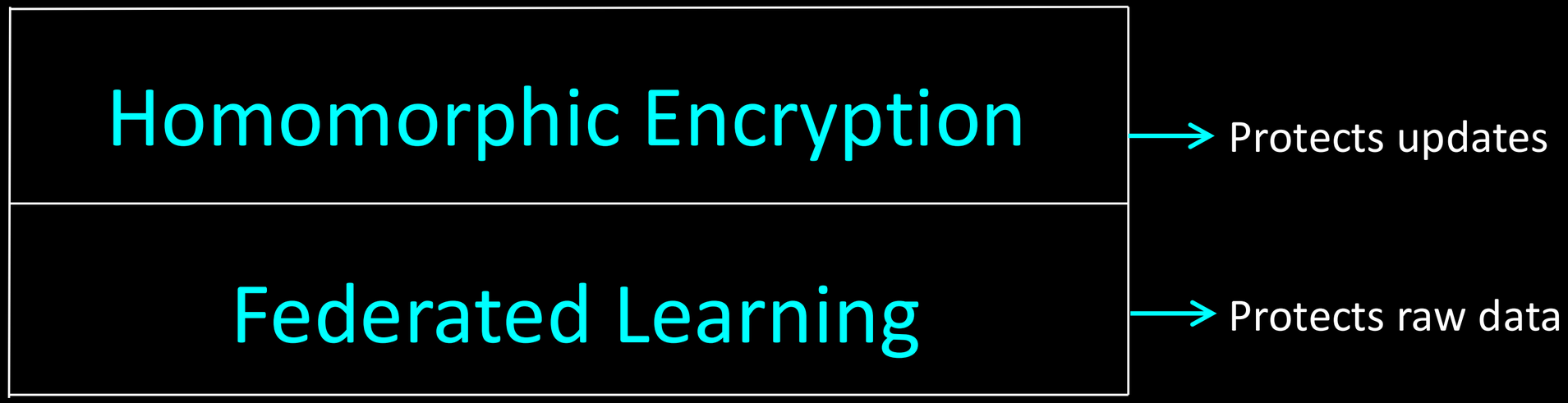
Privacy

Combining Machine Learning and Homomorphic Encryption in the Apple Ecosystem

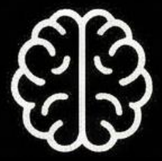


Using Private Nearest Neighbor Search for Enhanced Visual Search for photos.

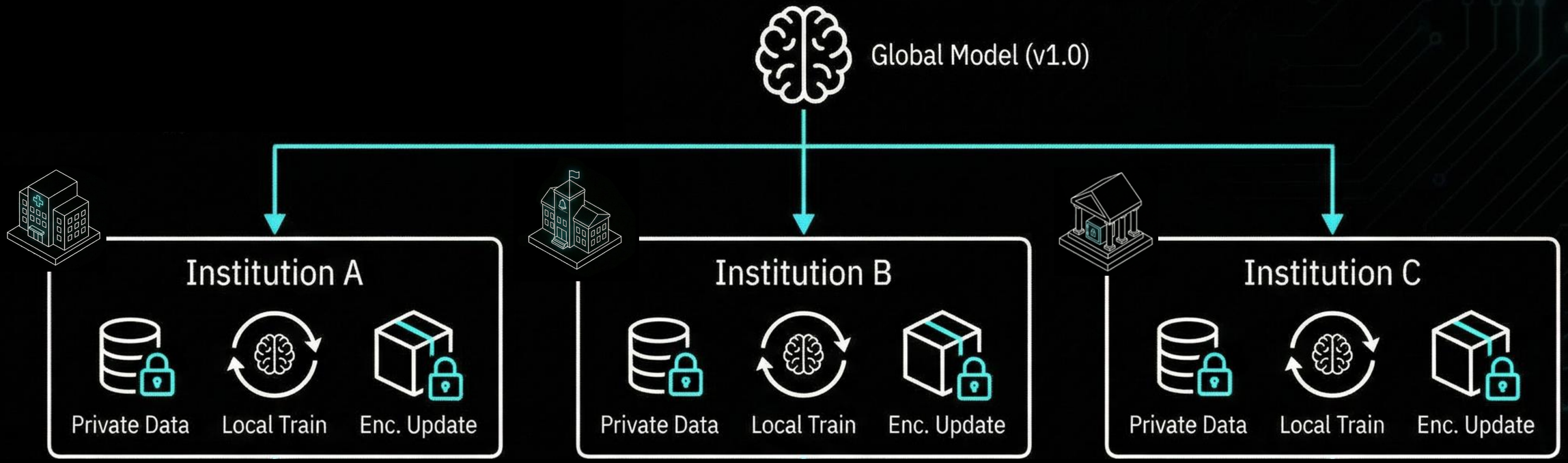
Trustworthy AI

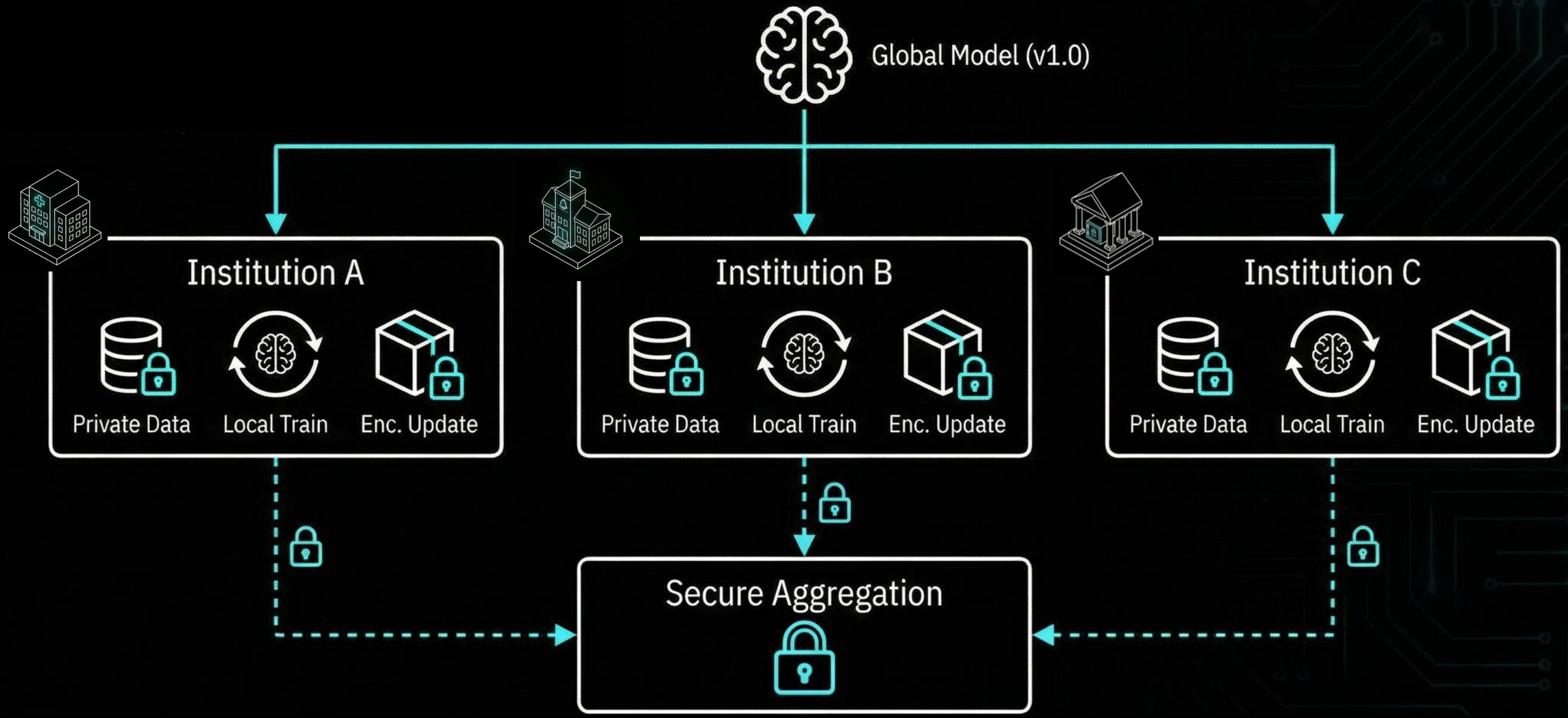


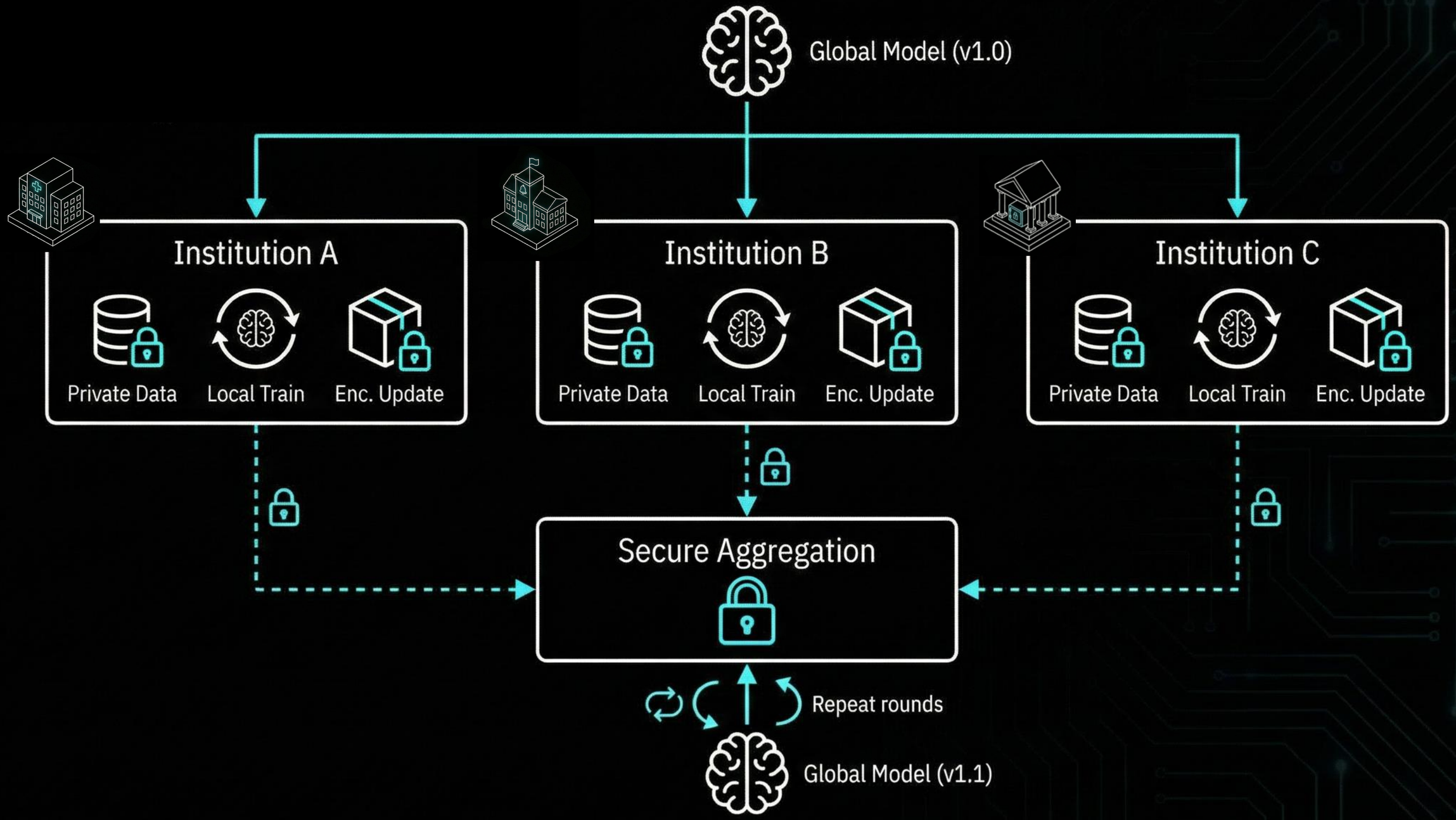
Let’s put it all together

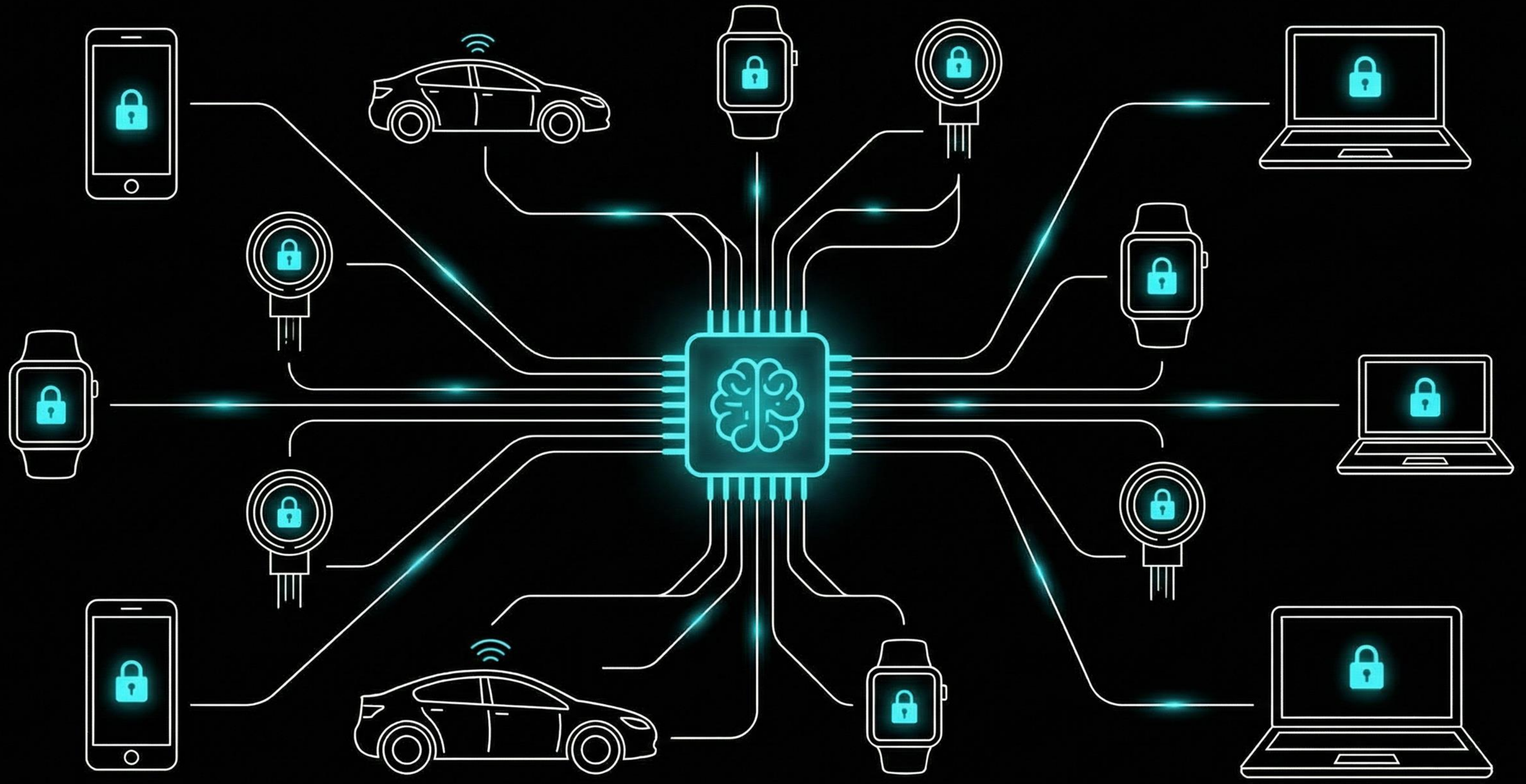


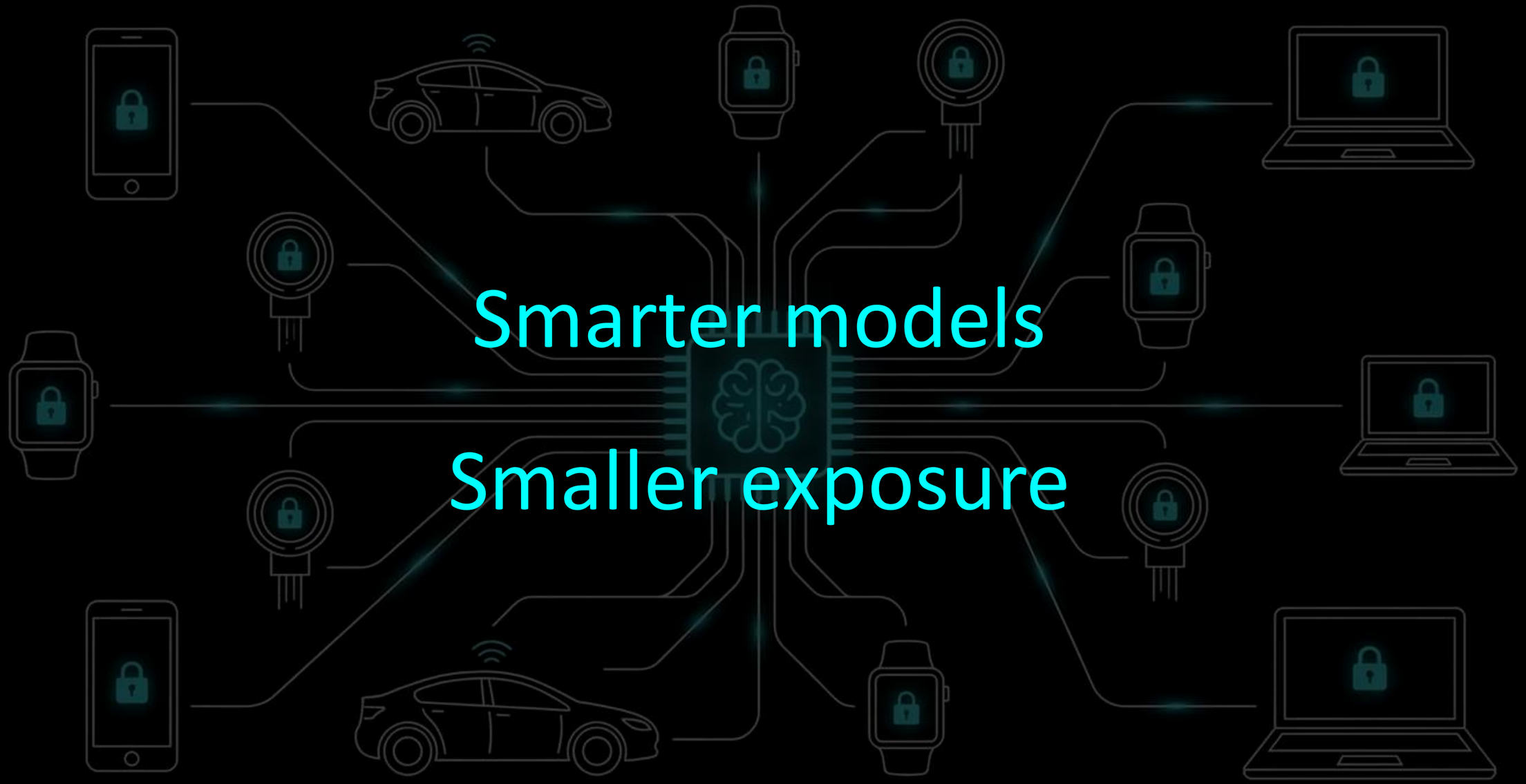
Global Model (v1.0)


Global Model (v1.0)
Institution A
Private Data
Local Train
Enc. Update
Institution B
Private Data
Local Train
Enc. Update
Institution C
Private Data
Local Train
Enc. Update


Global Model (v1.0)
Institution A
Private Data
Local Train
Enc. Update
Institution B
Private Data
Local Train
Enc. Update
Institution C
Private Data
Local Train
Enc. Update
Secure Aggregation


Global Model (v1.0)
Institution A
Private Data
Local Train
Enc. Update
Institution B
Private Data
Local Train
Enc. Update
Institution C
Private Data
Local Train
Enc. Update
Secure Aggregation
Repeat rounds
Global Model (v1.1)




Smarter models
Smaller exposure

What to remember:

What to remember:

1. Keep data local

What to remember:

1. Keep data local

2. Encrypt the updates

What to remember:

1. Keep data local
2. Encrypt the updates
3. Share the learning

Trustworthy AI by design,
not by promise

The smartest AI systems will be the ones that need to see the least.

Selected References

1. McMahan et al., “*Communication-Efficient Learning of Deep Networks from Decentralized Data*,” AISTATS 2017
2. Bonawitz et al., “*Practical Secure Aggregation for Privacy-Preserving Machine Learning,*” CCS 2017
3. Zhu et al., “*Deep Leakage from Gradients*,” NeurIPS 2019
4. Google Research, “*Federated Learning without Centralized Training Data,*” 2017
5. HomomorphicEncryption.org, “*Homomorphic Encryption Standard,” 2026*
6. Apple, “*Announcing Swift Homomorphic Encryption*,” 2024
7. Microsoft SEAL / OpenFHE / HEIR documentation