The Infrastructure for Blockchain 3.0

Ted Yin // Ava Labs & Cornell University
Twitter: @Tederminant @avalabsofficial

A Bit About Myself.

Maofan "Ted" Yin
Co-founder & Chief Protocol Architect @ Ava
PhD Student @ Cornell

- I enjoy building practical systems
- Backed by theories

HotStuff protocol and its prototype

- Consensus protocol used by Libra
- libhotstuff surpasses the state-of-art PBFT implementation performance

Snow protocol family and Avalanche

- This talk!

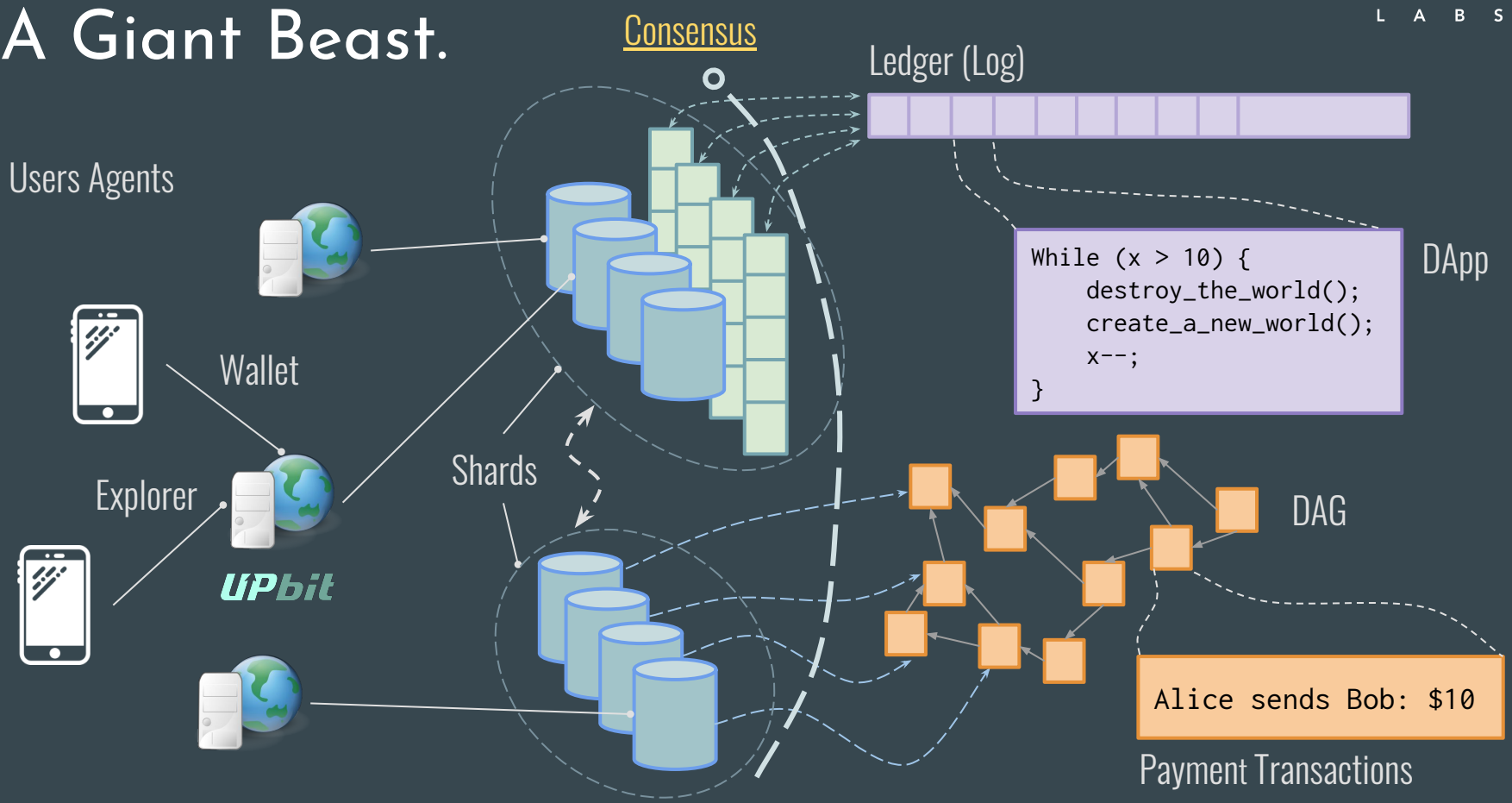
The Infrastructure for Blockchain 3.0

Infrastructure.

What is a “Blockchain Infrastructure” ?

How is it like in the year of 2019?

A Giant Beast.

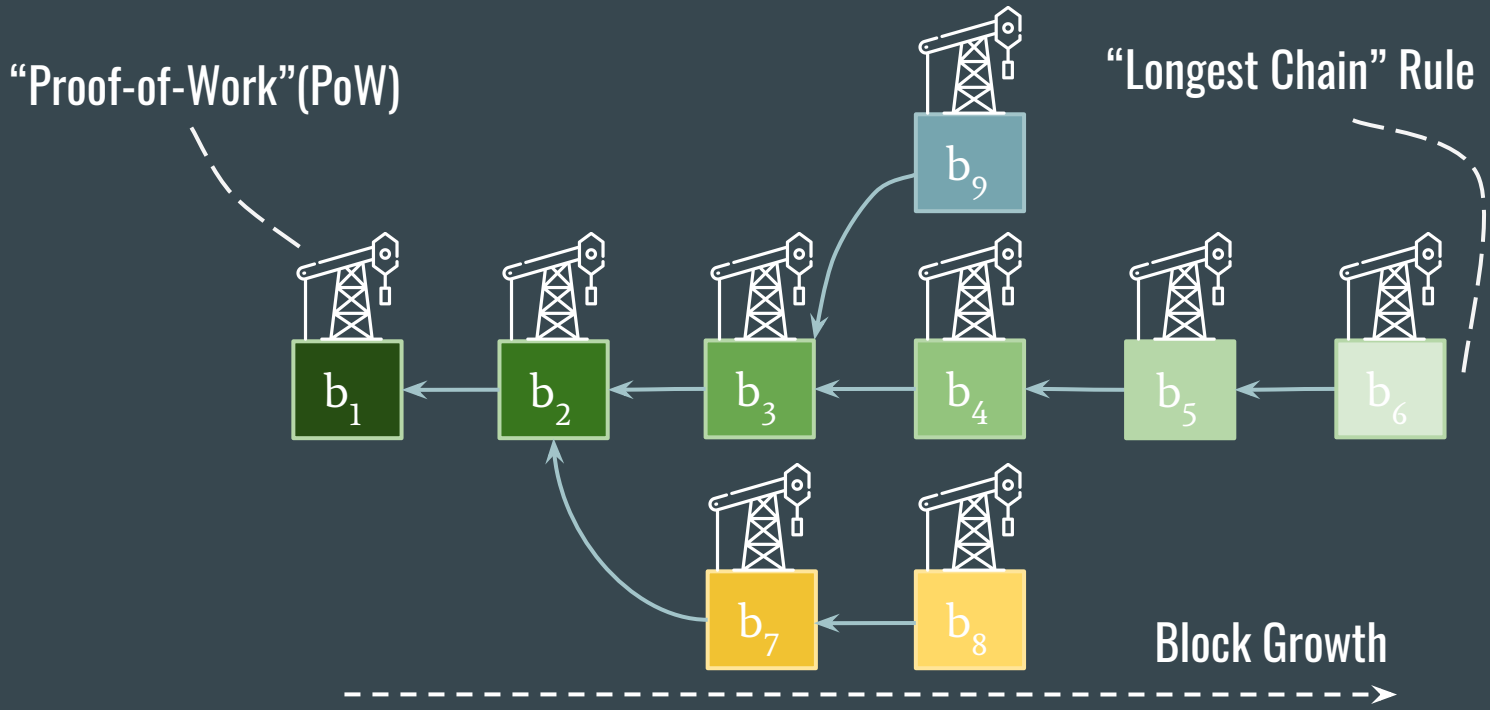


Blockchain 3.0?

Blockchain 1.0
Blockchain 2.0
Blockchain 3.0

LABS

Blockchain 1.0: Nakamoto Era.



Bitcoin: A Peer-to-Peer Electronic Cash System
Satoshi Nakamoto

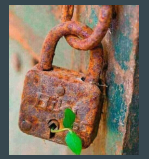
Blockchain 1.0: Nakamoto Era.

↑ Proof-of-work also serves as Sybil prevention (“Permissionless”)
↑ Graceful safety degradation
↑ Loose membership knowledge

↓ Proof-of-work wastes a lot of energy!
  - As of 2018: one Austria, two Denmarks, or three Irelands

↓ Extremely low capacity (throughput as low as ~3 TPS)

↓ Extremely long confirmation time (> 1 hour)
↓ Poor efficiency in safety (compared to Blockchain 2.0)



LABS

Blockchain 2.0: PBFT — Resurrect the Pharaoh.

↑ Very fast when the network is small
↑ Deterministic safety (100% safe)
↑ A long line of research

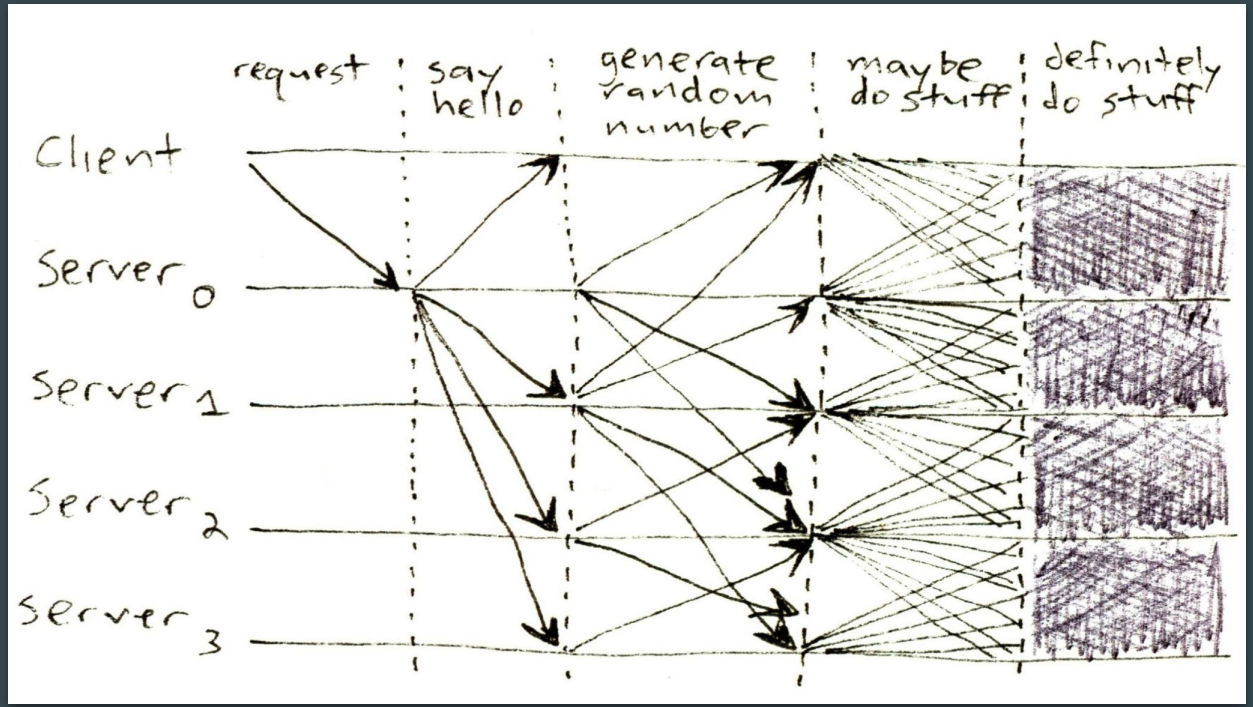
↓ Cannot scale easily: the “leader/coordinator” dilemma
  - Randomized BFT approaches are even worse

↓ Quorum-based: 100% accurate knowledge of all participants
↓ Hard limit on the Byzantine adversaries

↓ Many are very complicated

LABS

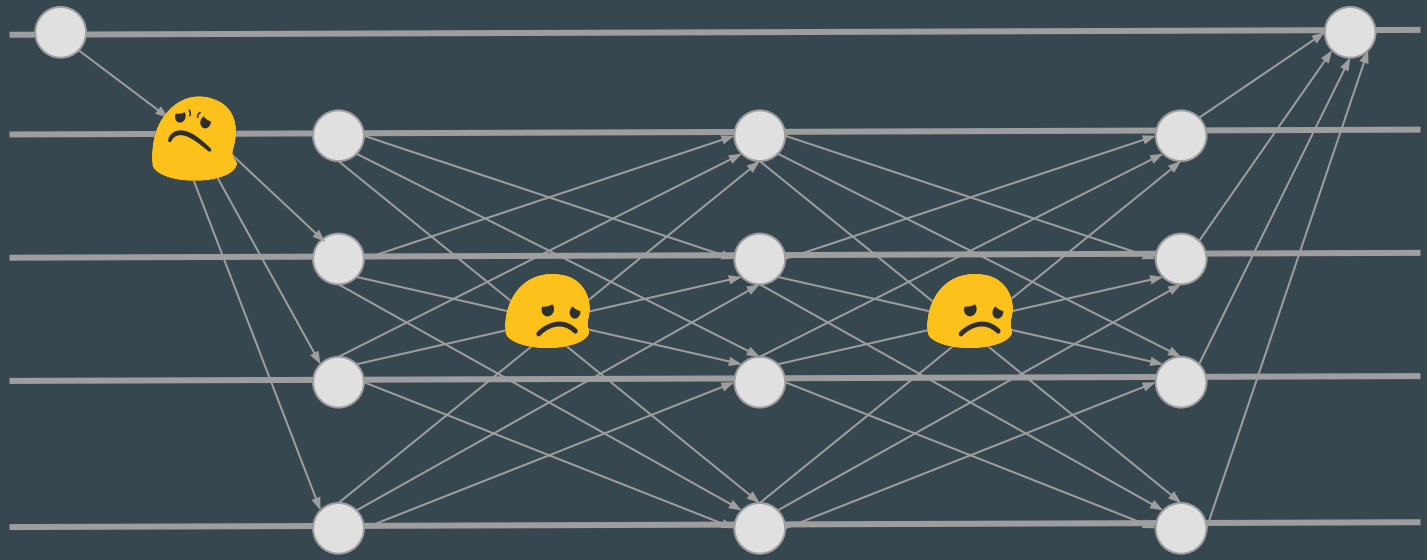
Blockchain 2.0: PBFT — Resurrect the Pharaoh.



The Saddest Moment, Mickens 2013

LABS

Performance Bottlenecks: PBFT as an Example.



LABS

Blockchain 3.0?

Blockchain 1.0
Blockchain 2.0
Blockchain 3.0

LABS

Blockchain 3.0...

Consensus 1.0	Blockchain 1.0
Consensus 2.0	Blockchain 2.0
Consensus 3.0	Blockchain 3.0

LABS

Wait!

> **“But we have sharding!”**
>
> Theorem 1: Sharding does NOT address the scalability issue of Consensus.
>
> Corollary 1: Sharding comes at the cost of losing fault tolerance.

The “Sharding” Logic.

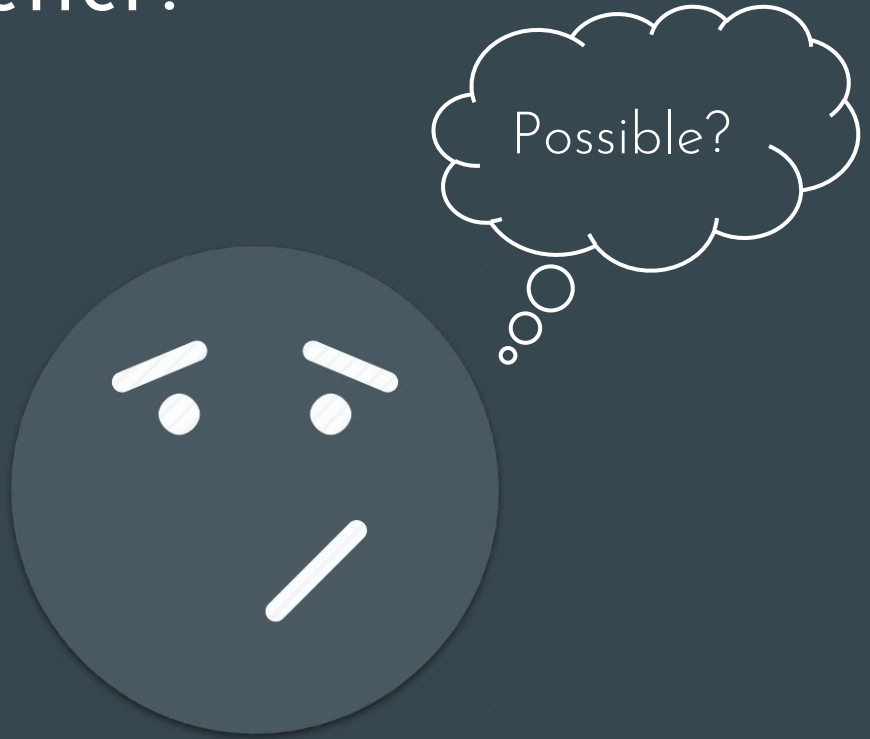
> “This soup tastes sooo bad! :(”

> “But you can add some Gochujang so you don’t feel too bad.”

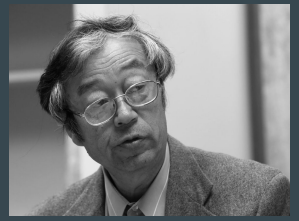
Sharding! Sharding? Sharding...

> "This soup tastes sooo bad! :("
> "But you can add some Gochujang so you don't feel too bad."
> ...
>
> Why couldn't I just have Budae Jjigae? And...you can also add Gochujang. :)

Then, Could We Do Better?



LABS



…, 1989, 1999, …
Quorum-Based;
Paxos/Raft, PBFT

“2f + 1”
“3f + 1”

2008
Longest-Chain;
Nakamoto

50% power

2018
Random Sampling;
Snow/Avalanche

???

Blockchain 3.0: Snow/Avalanche.

- A paper presenting “Avalanche” was dropped on IPFS in May 2018
- Inspired by epidemic protocols and gossip networks
- New methodology that combines the best of 1.0 and 2.0

↑ Graceful safety degradation
↑ Loose membership knowledge
↑ Very fast regardless of the network size
↑ Energy efficient
↑ The operational logic just makes sense

Wait!

> **“But many other protocols also claim very high TPS!”**
>
> Theorem 2: Throughput only reflects how much load a system can buffer, while latency only reflects how fast a single user is served.
>
> Corollary 2: We will need them both when evaluating a system!

The TPS vs. Latency Logic.

A bank branch could serve a single person very quickly.

But what if there are many people waiting in the line?

Throughput reflects the capacity.

The TPS vs. Latency Logic.

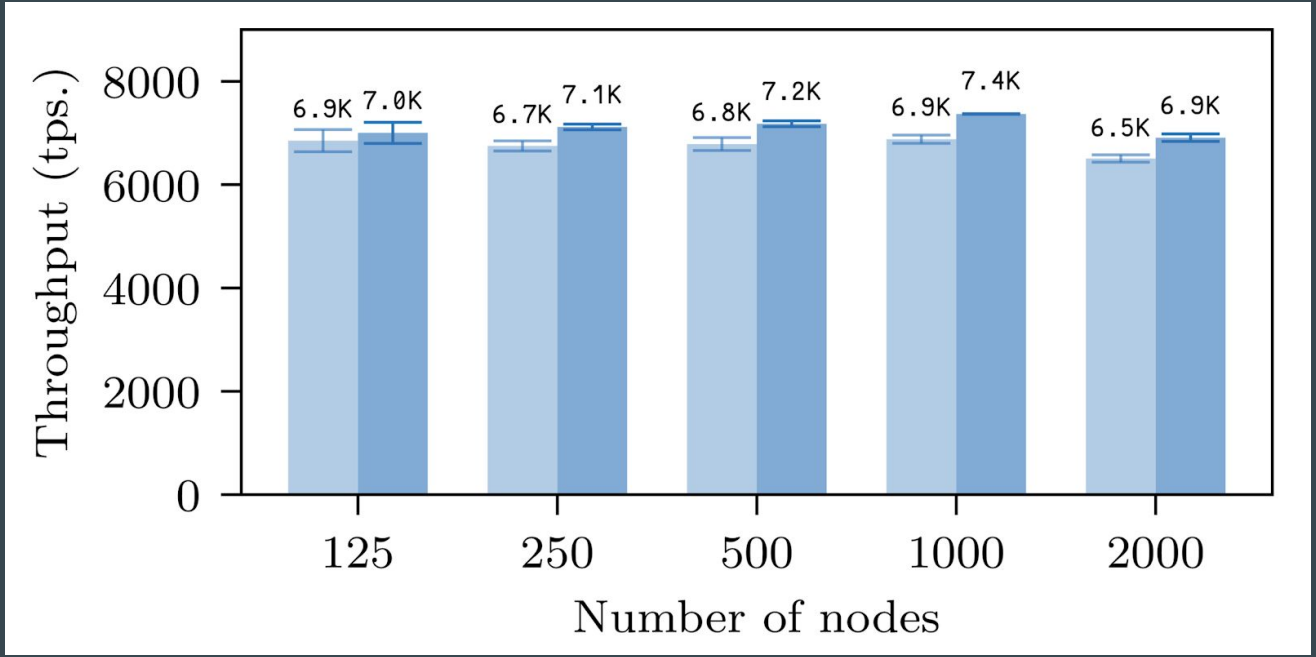
A truck full of hard drives can transport Petabytes (2^{50} bytes) of data.

Oh, very high throughput!

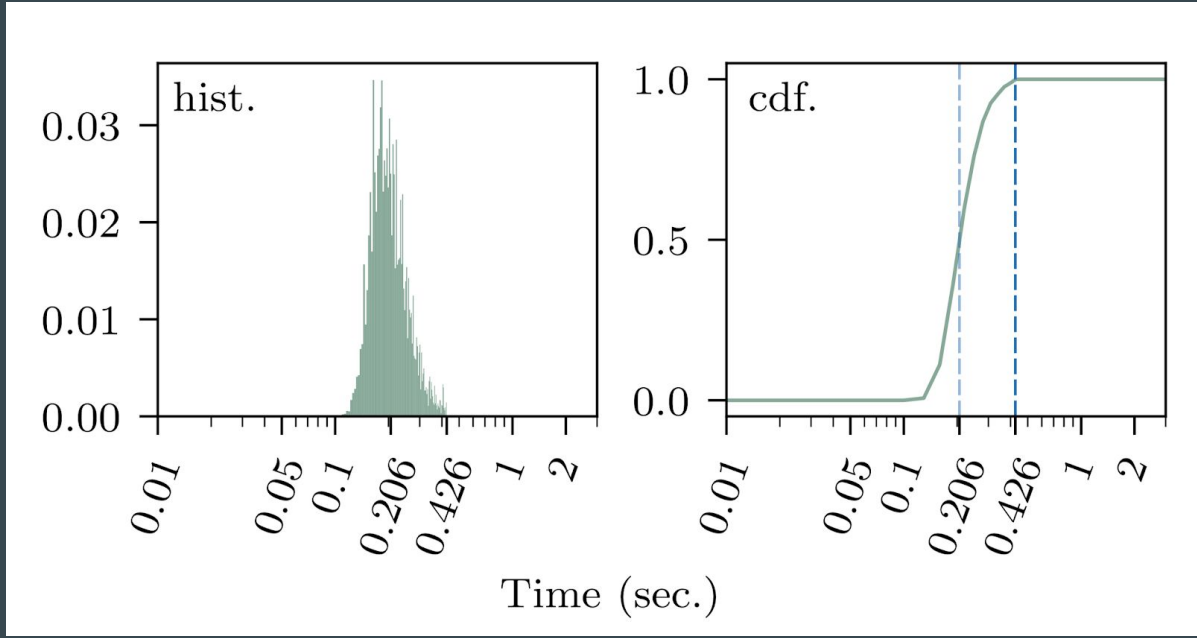
Well, the latency...

Talking about throughput without latency is meaningless.

Ava Throughput.



Ava Latency.


hist.
0.03
0.02
0.01
0.00
0.01
0.05
0.1
0.206
0.426
1
2
cdf.
1.0
0.5
0.0
0.01
0.05
0.1
0.206
0.426
1
2
Time (sec.)

Blockchain 3.0: Snow/Avalanche.

In an even more realistic setting:

- 2000 nodes in 20 cities across the globe
- All nodes directly participate in consensus
- Full signature verification

Our evaluation results:

- ~3400 tps
- ~1.35 sec

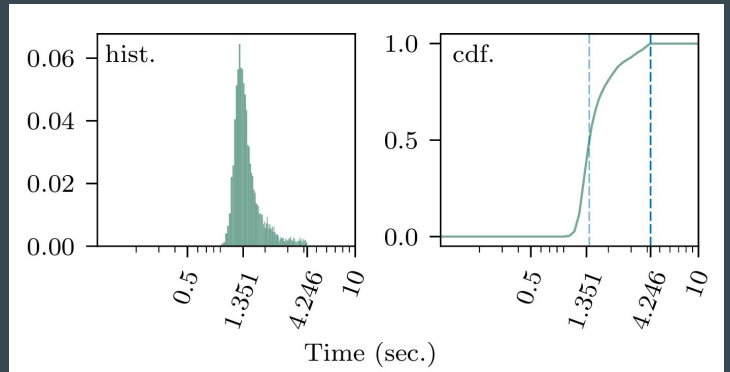


Fig. 19: Latency histogram/CDF for $n = 2000$ in 20 cities.

Blockchain 3.0: Snow/Avalanche.

That's interesting.

Tell me how this protocol works and why it is so efficient!

Gossip network is not only a way to deliver data,

but can also be used as a consensus!

The Snow Protocol Family: Let Rumors Spread.

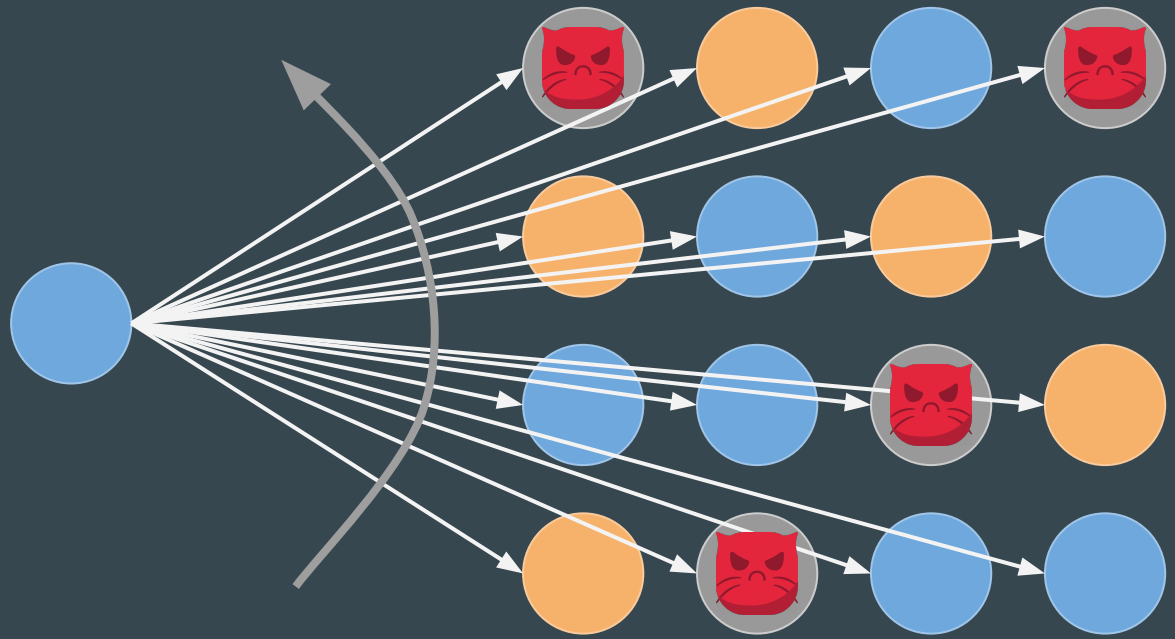
- “Gossip”: N people are discussing whether some rumor is true
- Everyone independently choose k others at random — “sample” the crowd

For Alice:

- Asks those k people whether the rumor is true — a “query”
- Updates the confidence in her current bias (true/false) by checking the majority opinion of each “query”

Full Broadcast to Partial Sampling.

Sample all

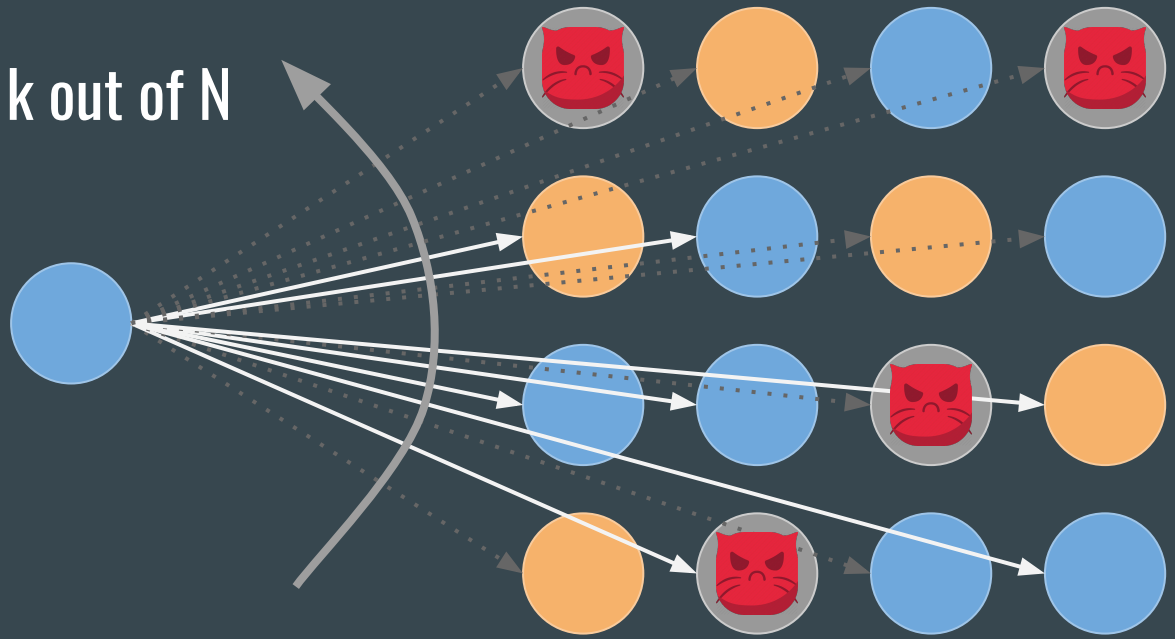


LABS

Full Broadcast to Partial Sampling.

~~Sample all~~

Only Sample k out of N



LABS

Alice asks
5 other people
randomly

Alice: “Blue is the majority answer!”

Bob asks
5 other people
randomly

Alice: “Blue is
the majority
answer!”

Alice asks
5 other people
randomly

Bob: “Yellow is
the majority
answer!”

Bob asks
5 other people
randomly

Alice: “Blue is the majority answer!”

Alice asks
5 other people
randomly

Bob: “Yellow is the majority answer!”

Bob asks
5 other people
randomly

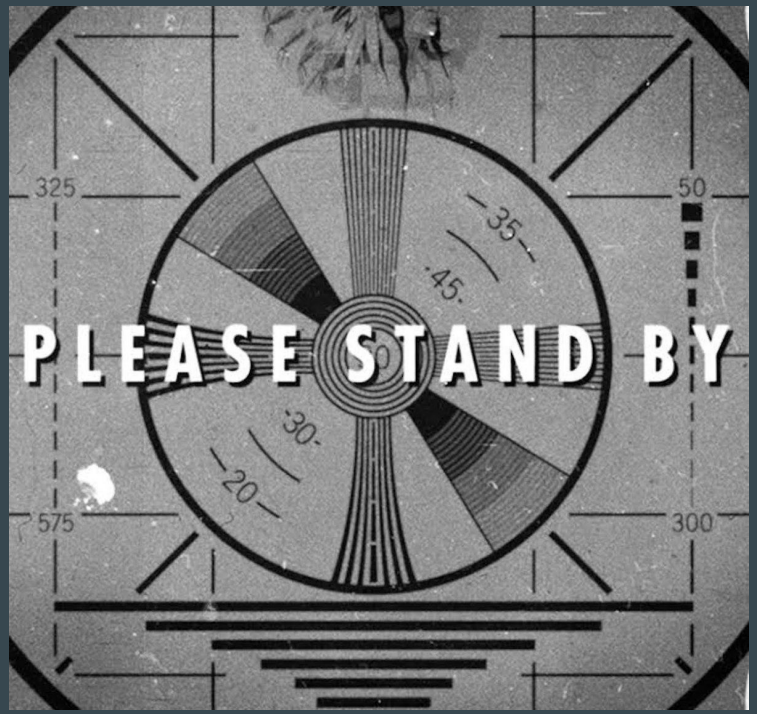
Alice: “Blue is the majority answer!”

Alice asks
5 other people
randomly

Bob: “Yellow is the majority answer!”

Protocol Demo

https://avalabs.org/snow-bft-demo



LABS

The Snow Protocol Family: Uniqueness

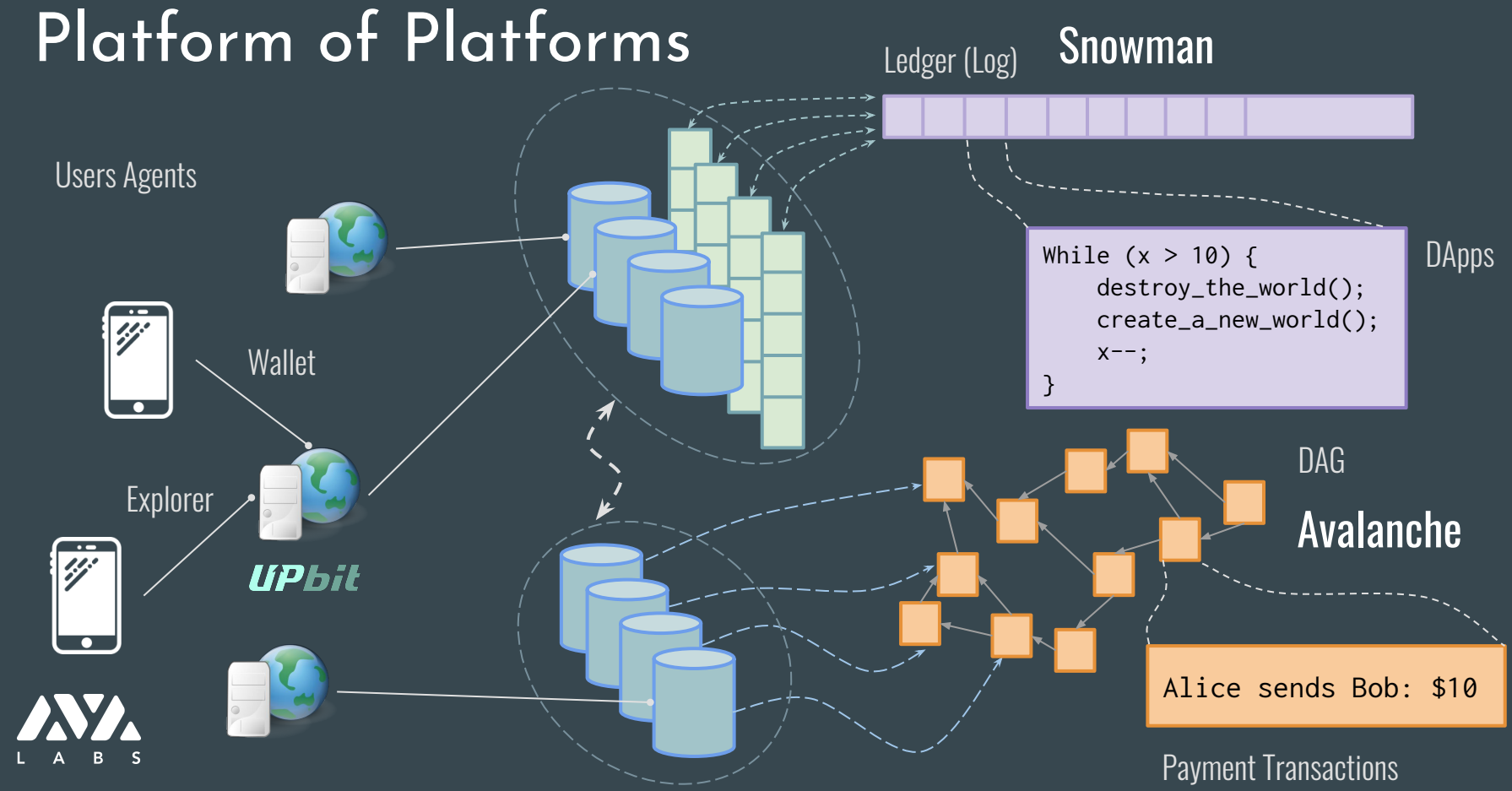
- The “magical” k: k = 10~20, for N = 1000, 10000, ...
- It could be viewed as a heavily relaxed quorum system
- It doesn’t require any PoW

- As fast as the network propagates
- Super scalable with respect to the number of nodes
- Loose membership
- Any anti-Sybil mechanism could be applied

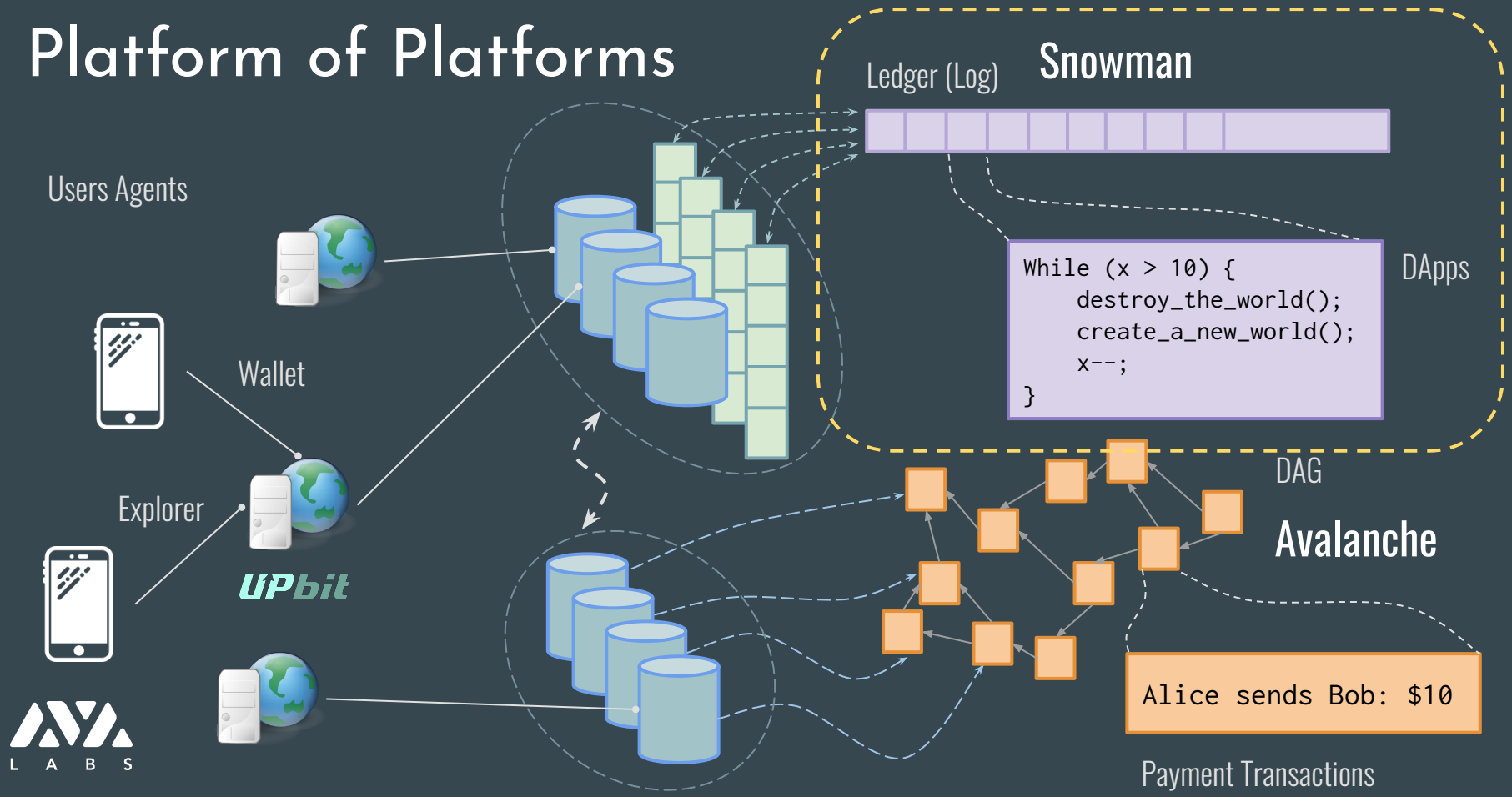
The Infrastructure for Blockchain 3.0

LABS

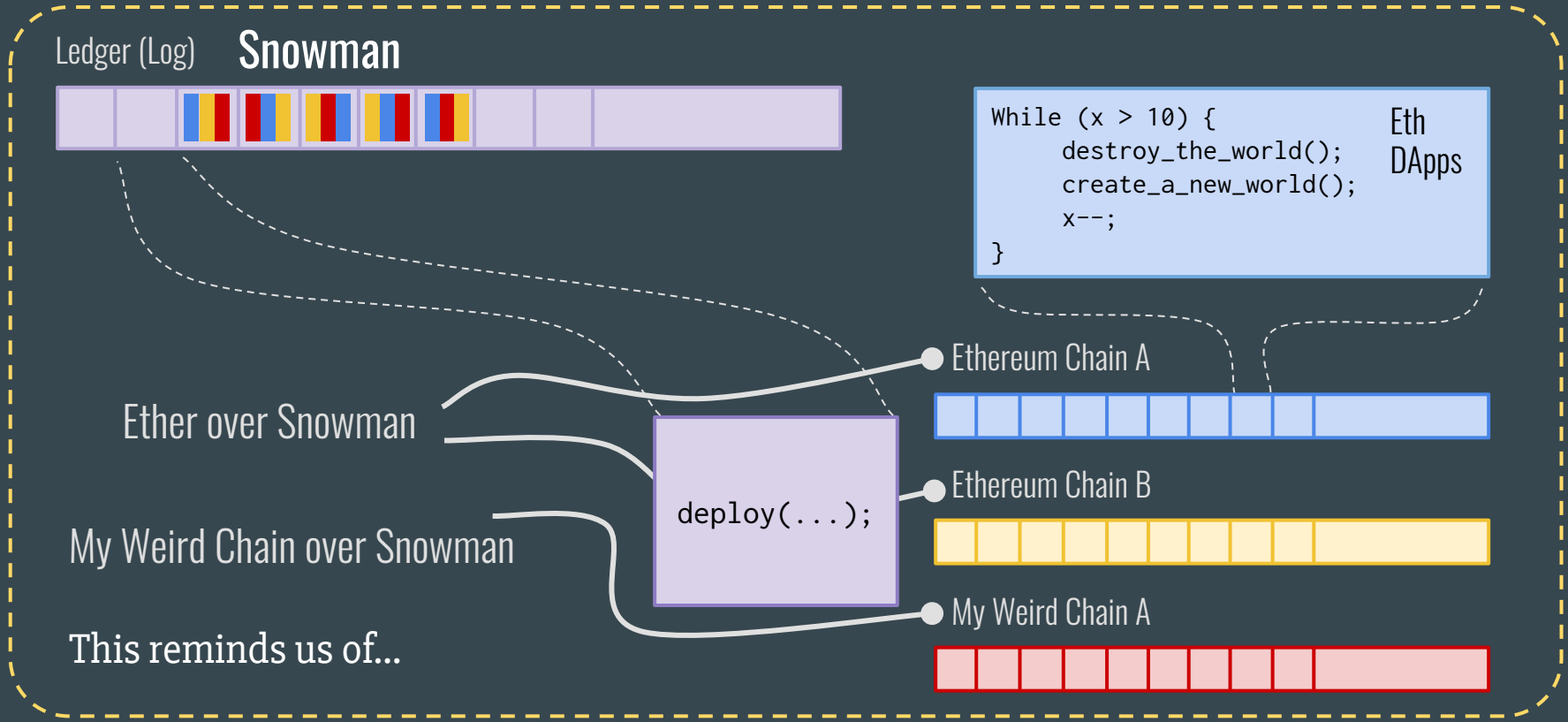
Platform of Platforms



Platform of Platforms



Platform of Platforms



I told you, sharding is just another trick. ;)

~~"This soup tastes sooo bad! :("~~
~~"But you can add some Gochujang so you don't feel too bad."~~

We're making our delicious Budae Jjigae,
...with Gochujang!

Public Testnet coming soon.
Our devs are working like a dog.
Stay tuned and contribute to our project in the future!

← avalabs.org

AVA LABS

Kakao (AVA Korea, 아바랩스)→