Turn Devices into Data Scientists

Simon Crosby, CTO, @swim

How will edge devices get "smart"?

Send data to the cloud?

- Too much data
- REST+ Big Data is too slow
- Streaming data pipelines are still a mystery

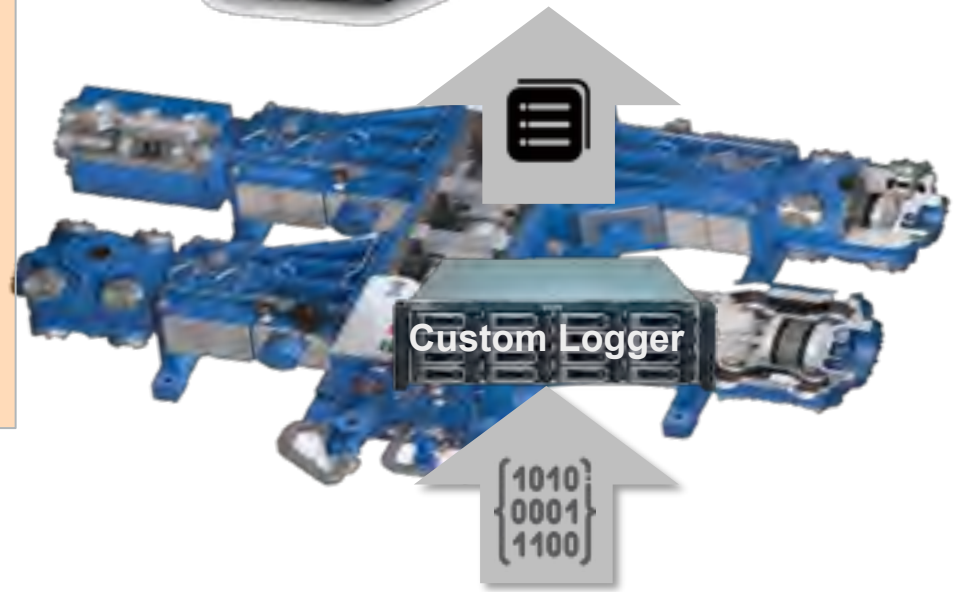
Train in the cloud, inference @ edge?

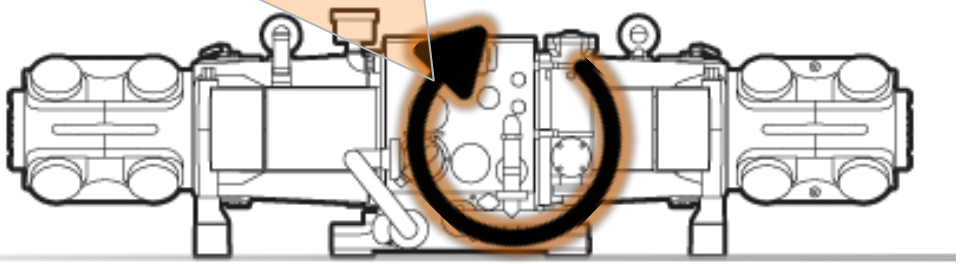
- Who builds the model?
- How is the model distributed to the edge?
- How to make models robust?

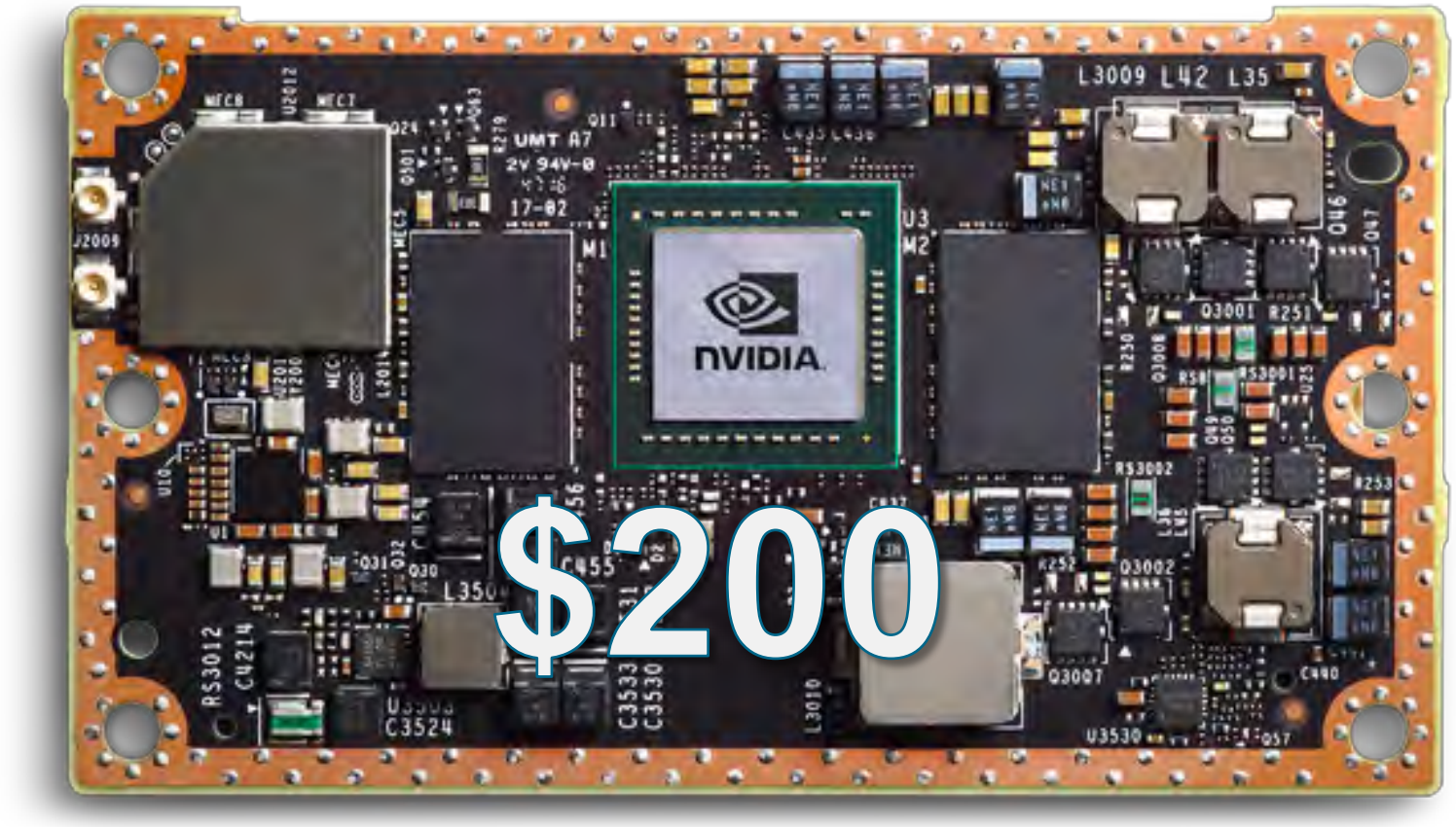
Big Data or Big Mistake?

mongoDB

- 70 samples per degree x 360° x 1000 RPM
- For 360 days/year
- > 50 PB / year
- x 40 Compressors
- ➢ Goal: 0.5% increased ROI per year on $1B CapEx







NVIDIA Jetson TX2

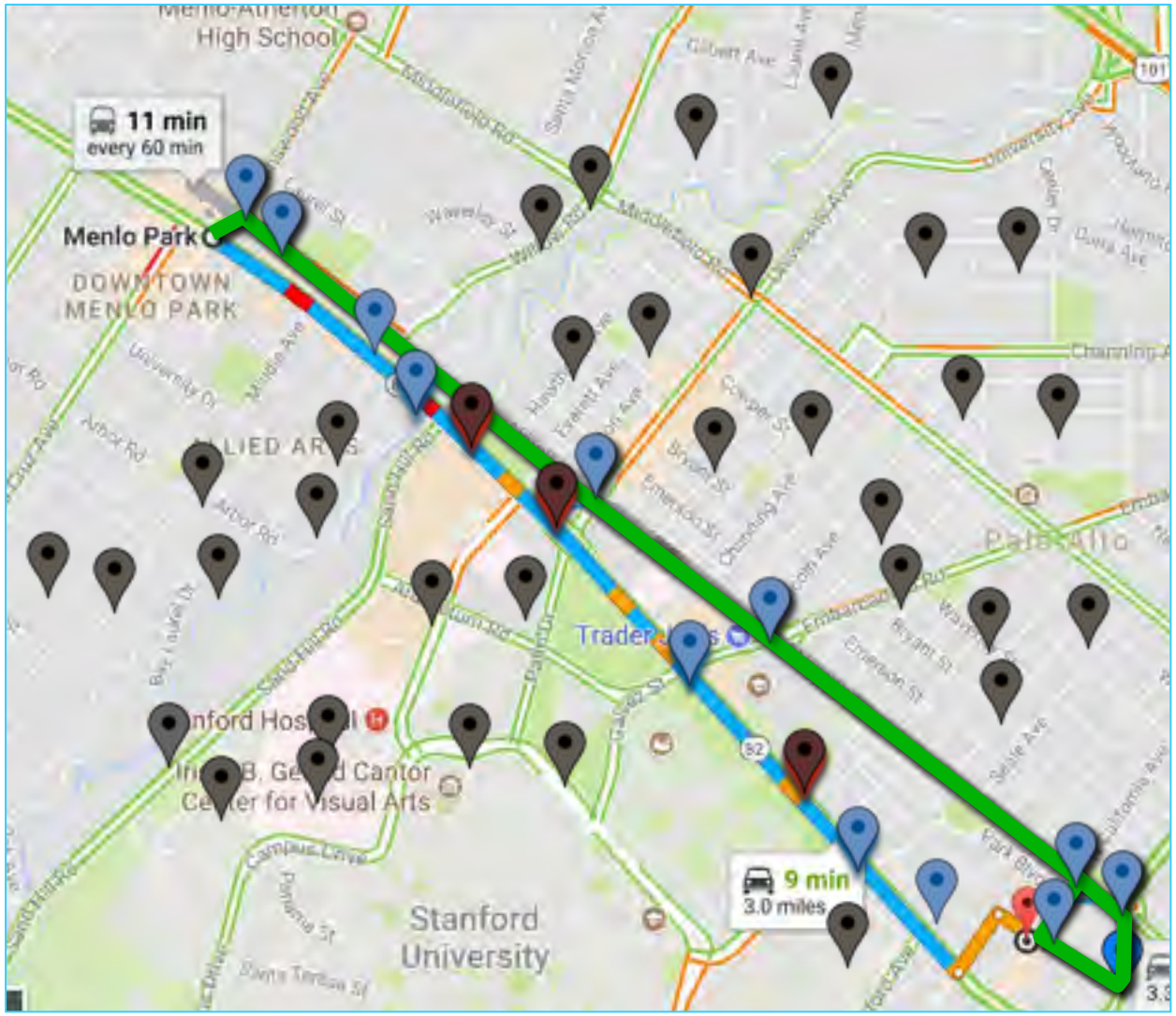
Quad **ARM®** A57/2 MB L2
NVIDIA Pascal™, 256 CUDA cores

…to route vehicles through a city without stops…

- Processing 4TB / day @ edge

  vs $5,000/month in the cloud

- Enabling a new market for insights



- To track & map millions of RFID tags in real-time
- With 70% cuts in bandwidth & storage, 50% cut in datacenter cost

How?

- Build a digital twin model of the real world directly from streaming data
- Digital twins collaborate to analyze, train & predict system behavior

vs

Edge ≠ Devices

Cloud apps are REST, stateless & *database centric*

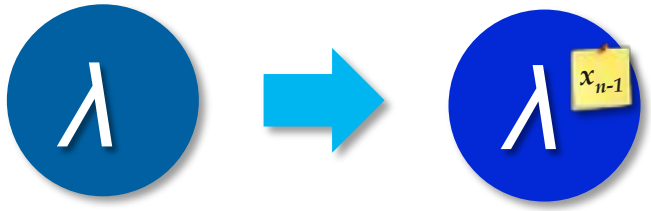
Edge ≠ Place

“Edge” applications are *data driven*

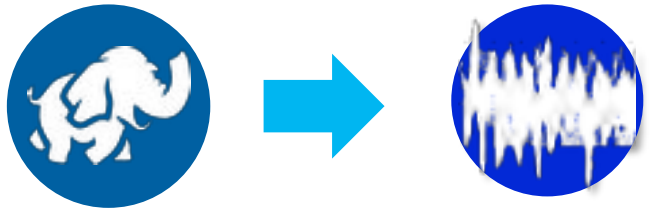
Database centric	Data Driven
Analyze Past Data	Analyze Present Data
Operate *on* Data Graphs	Operate *in* Data Graphs
Memory Limited	Time Limited
Highly Centralized	Horizontally Decentralized

“Edge” applications are *data driven*

Edge Apps Need a Different Paradigm

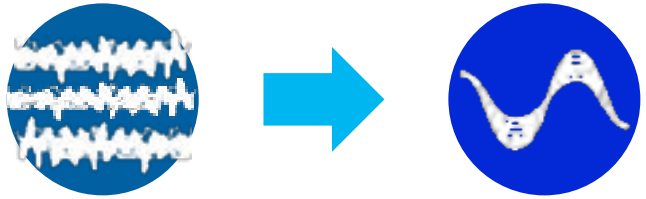


The real world is stateful



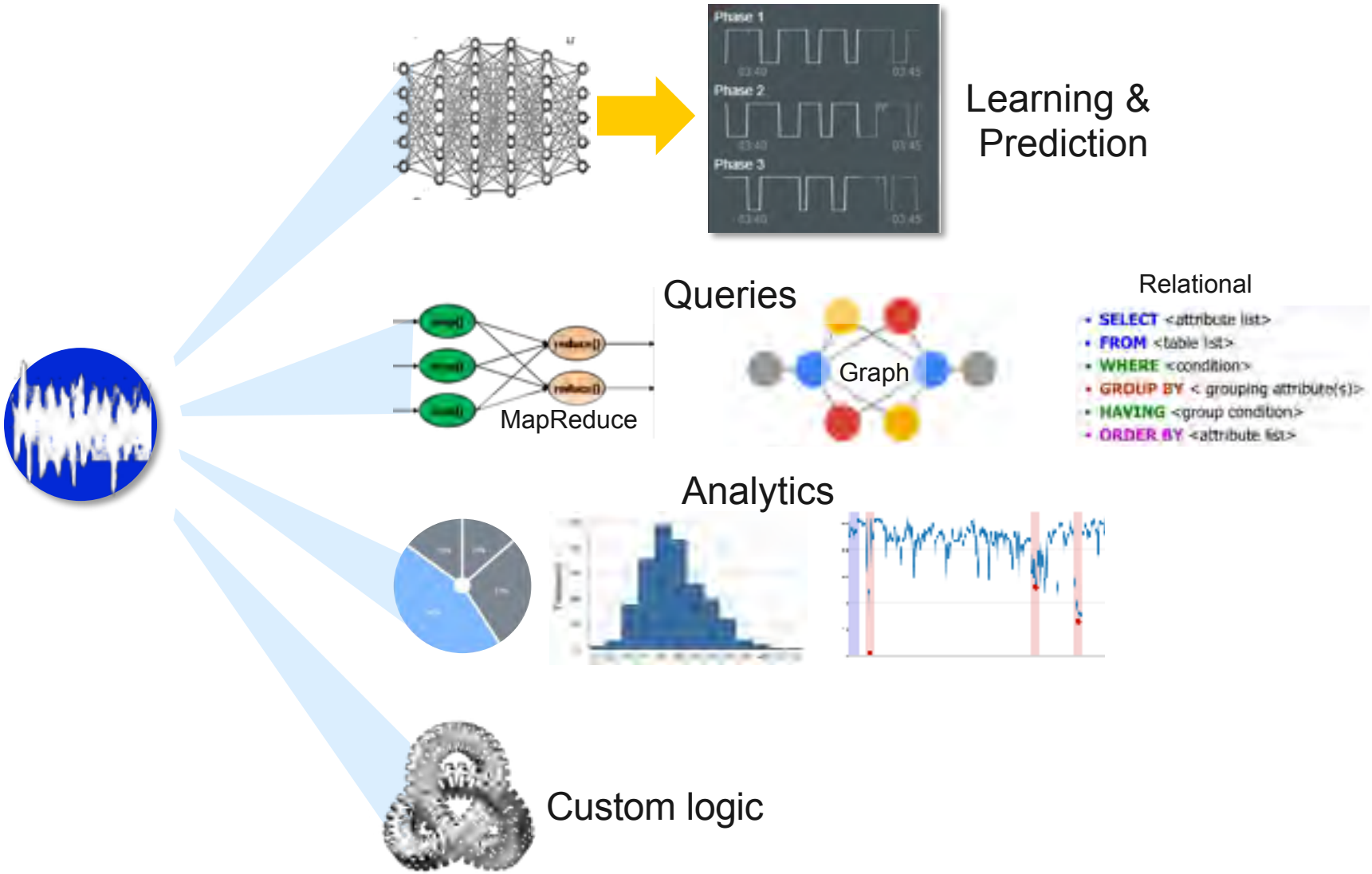
Vast amounts of data of ephemeral value

Need current insights to drive a real-time response



Dynamic discovery of real-world context is crucial

Data Driven Intelligence

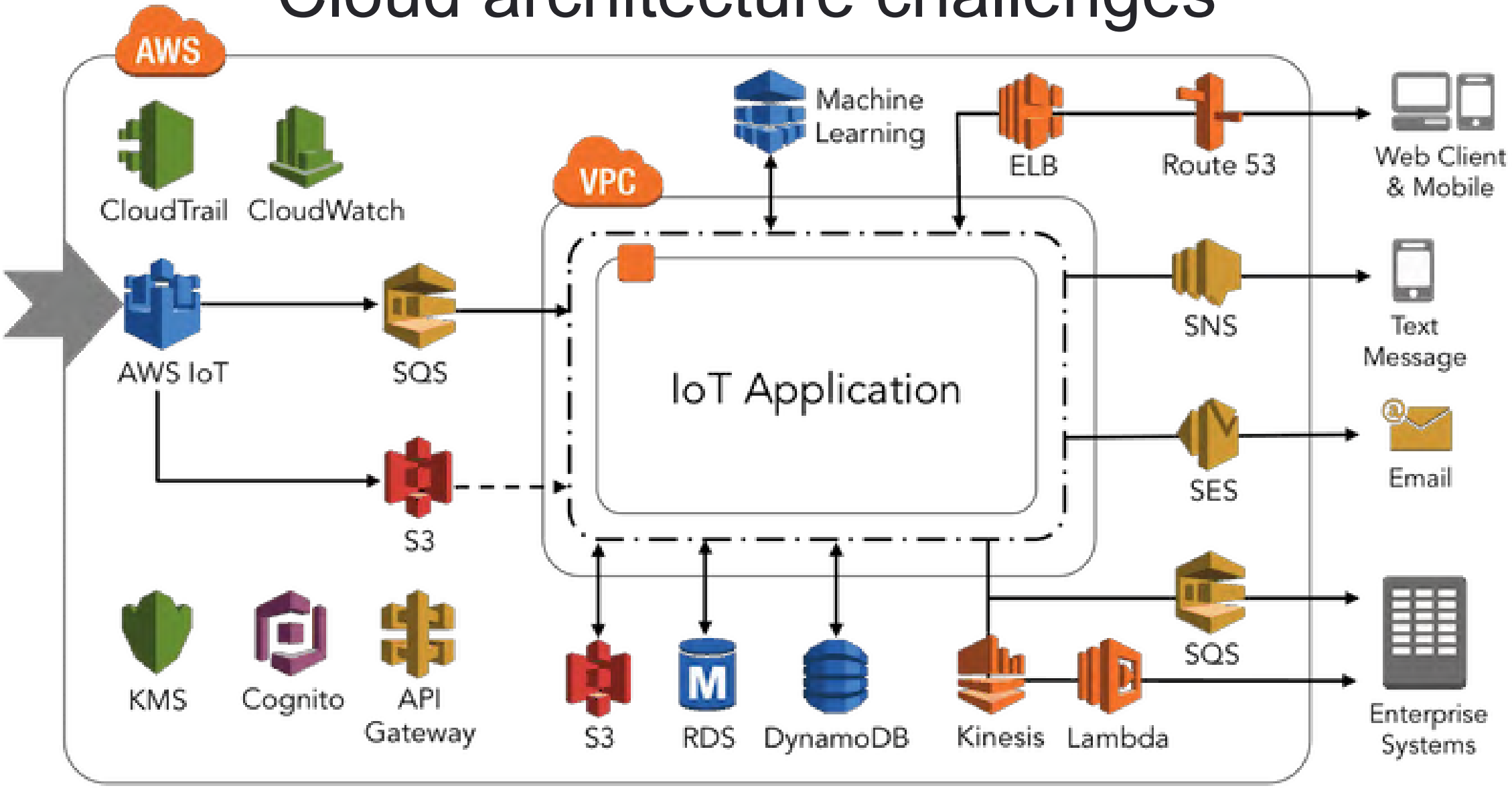


Swim Edge Intelligence

Swim is an <u>open source</u> *edge intelligence* platform that makes it easy to build stateful, distributed edge applications that stream insights in real-time

swim

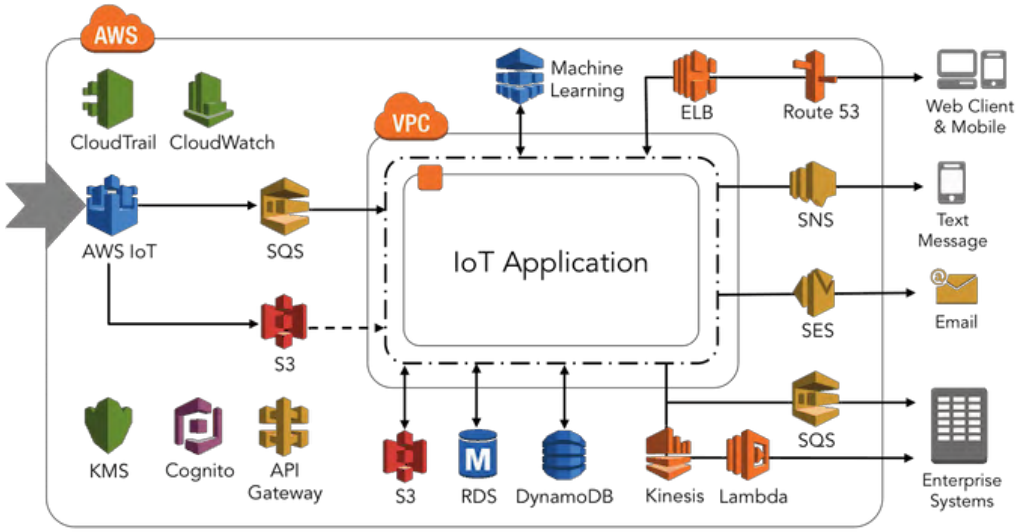
Cloud architecture challenges

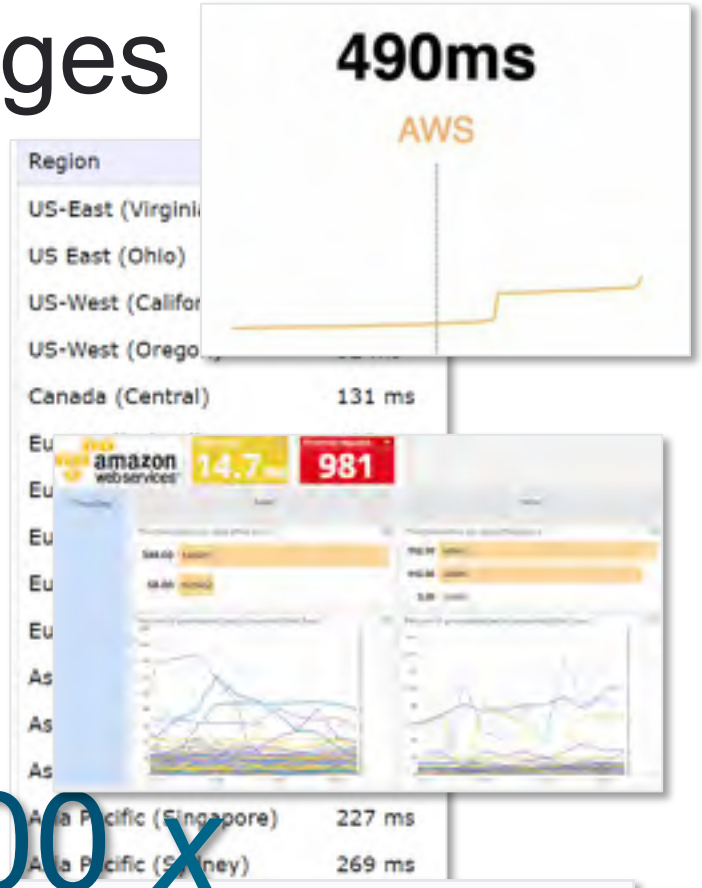


Cloud architecture challenges

- Networking 250 ms
- Lambda 500 ms
- Database lookup & store 25 ms
- Processing 0.1ms

775.1*ms*

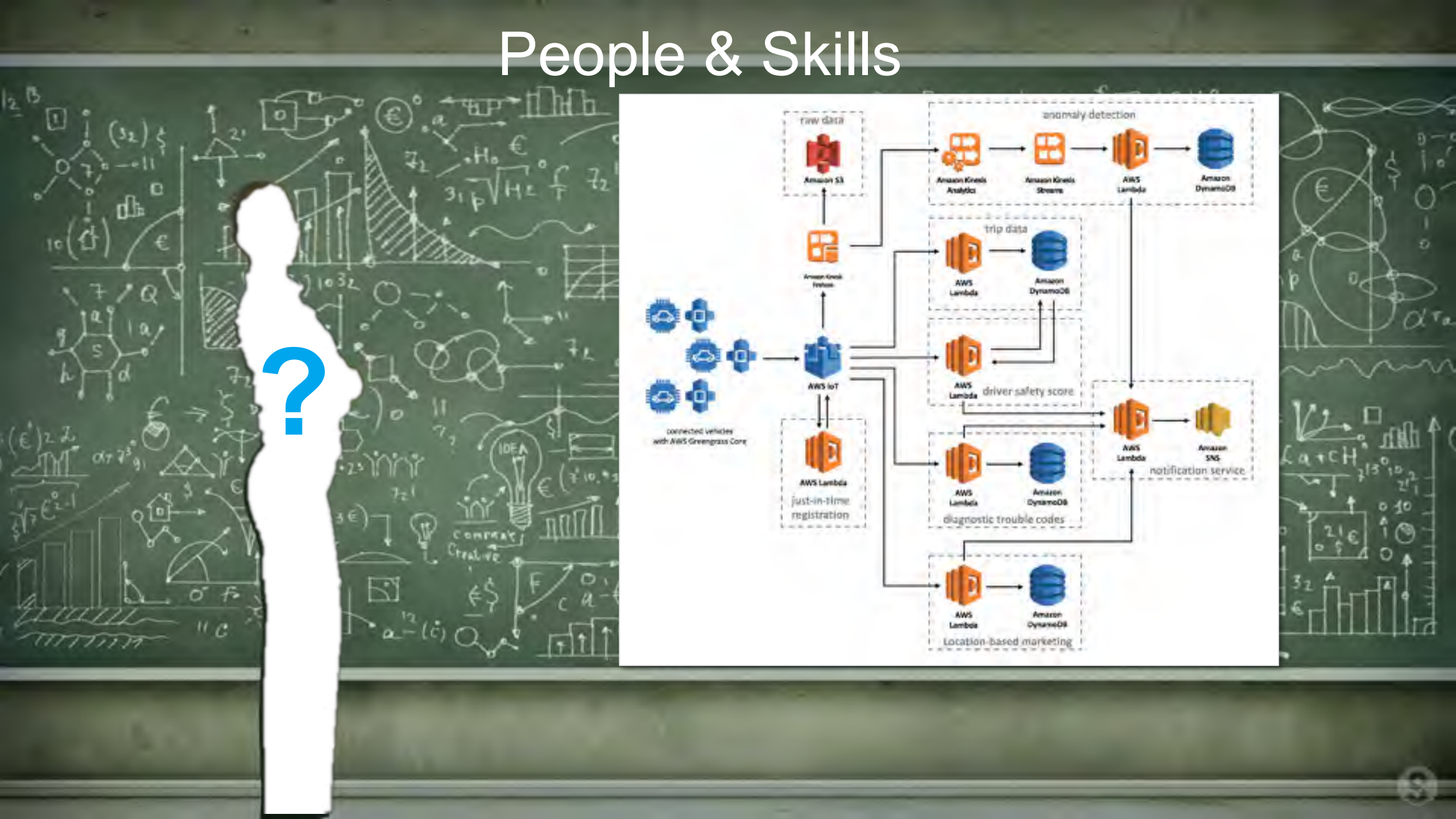




~10,000 x slower

Wastes 3BN cycles/ event

People & Skills



A New Edge Architecture

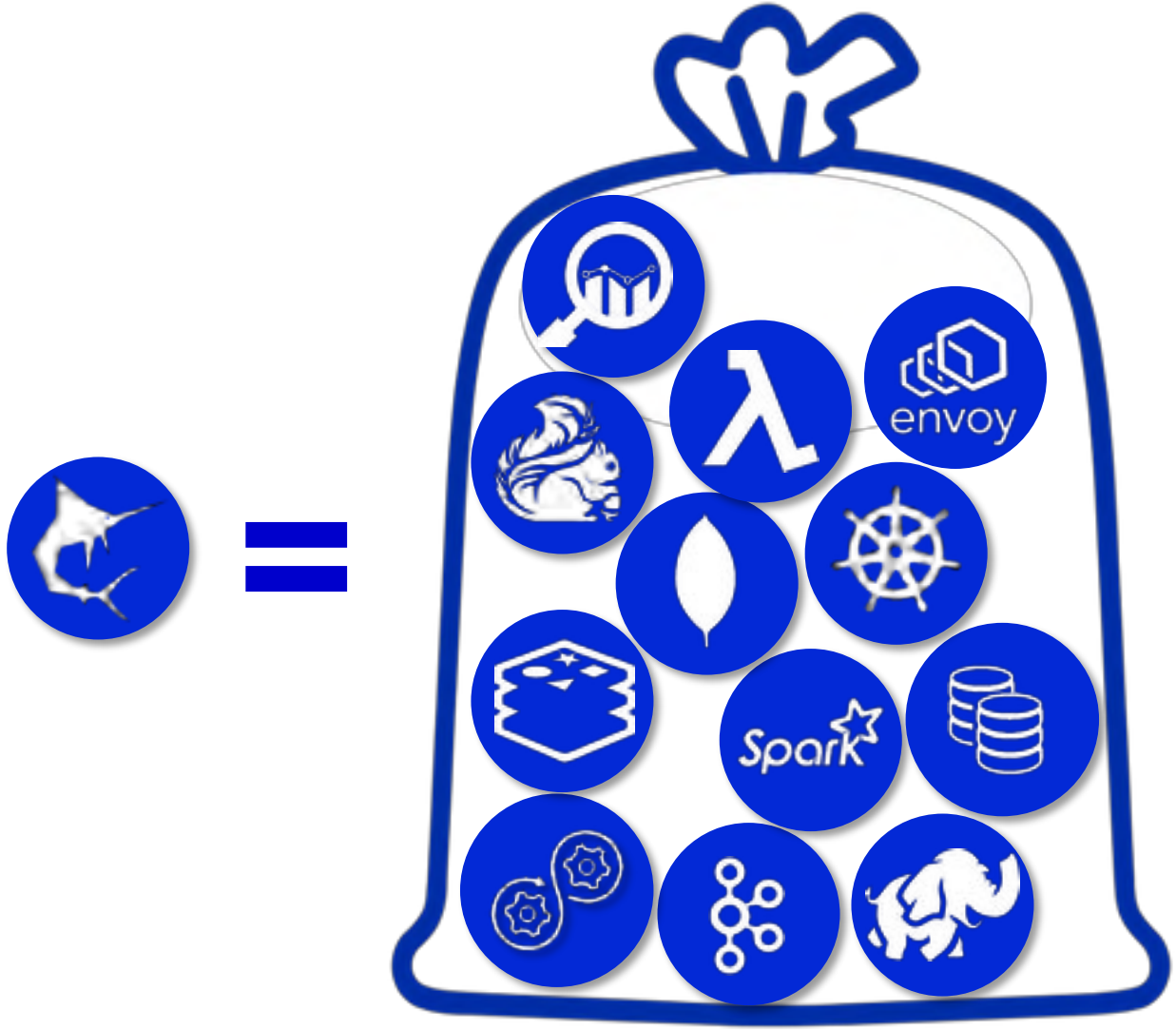
Stateful, distributed edge computing – a “web of things”

Active “digital twins“ are “things” that process their own data

Use data to build a digital twin model of real world relationships

Digital twins share state, collaborate, learn & predict in real-time

Stateful Means Vertically Integrated



Use Data to Build the Model

Developer defines entities & relationships (schema)

Data builds a stateful, distributed, digital twin model of the real-world

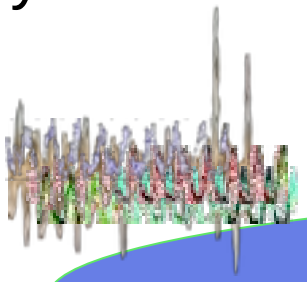
Twins collaborate to analyze, learn, predict and respond on the fly

Twins continuously stream real-time insights to UIs & applications

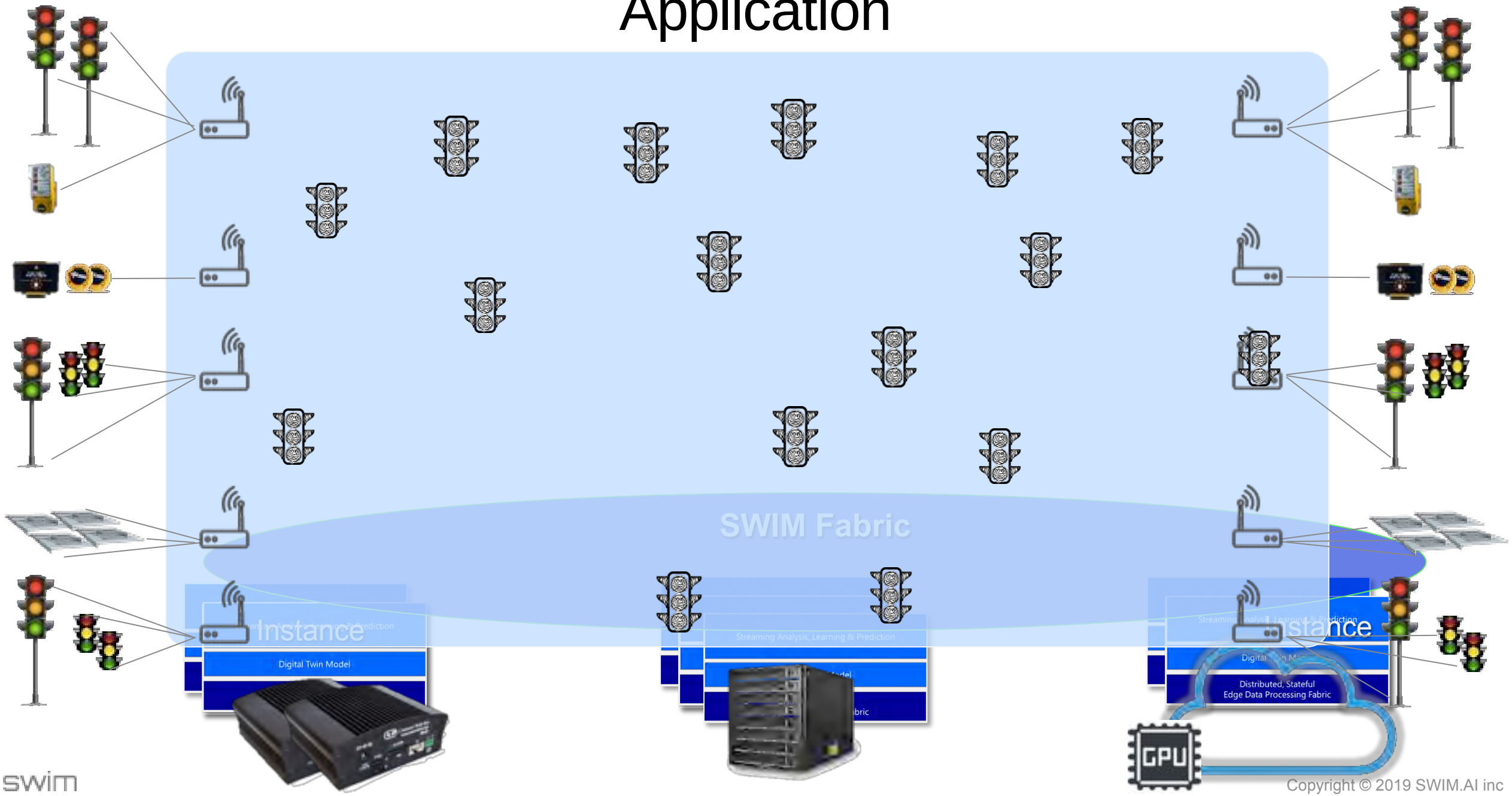
* Saves data you need to keep – just *not on the hot path*

Distribute the Stack “Edge to Cloud”

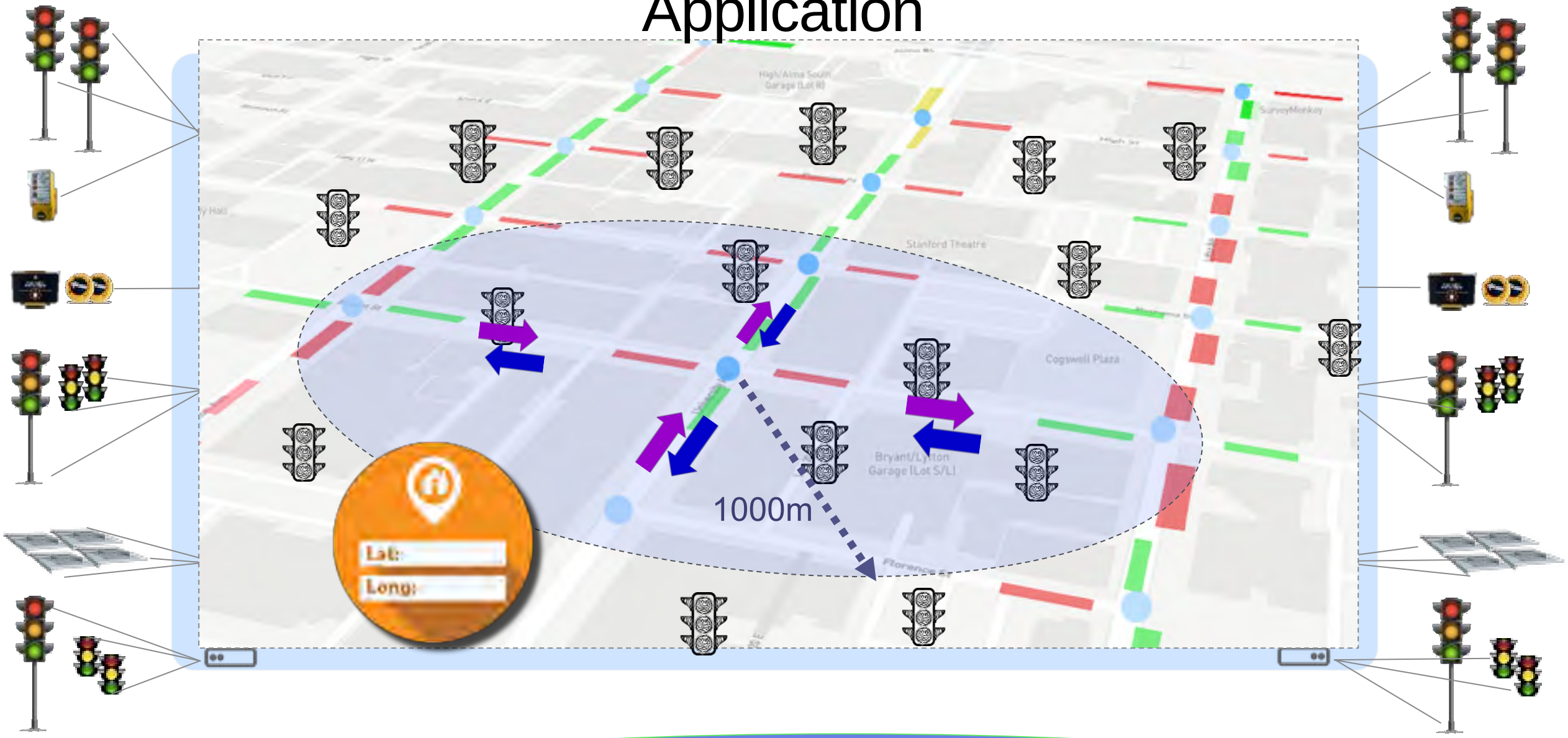
1. Build a resilient, self-managing fabric that spans edge, fog and cloud-hosted instances
2. Create a stateful “digital twin” for each real-world entity in the data
3. Each digital twin statefully reduces, labels & analyzes its data



Application

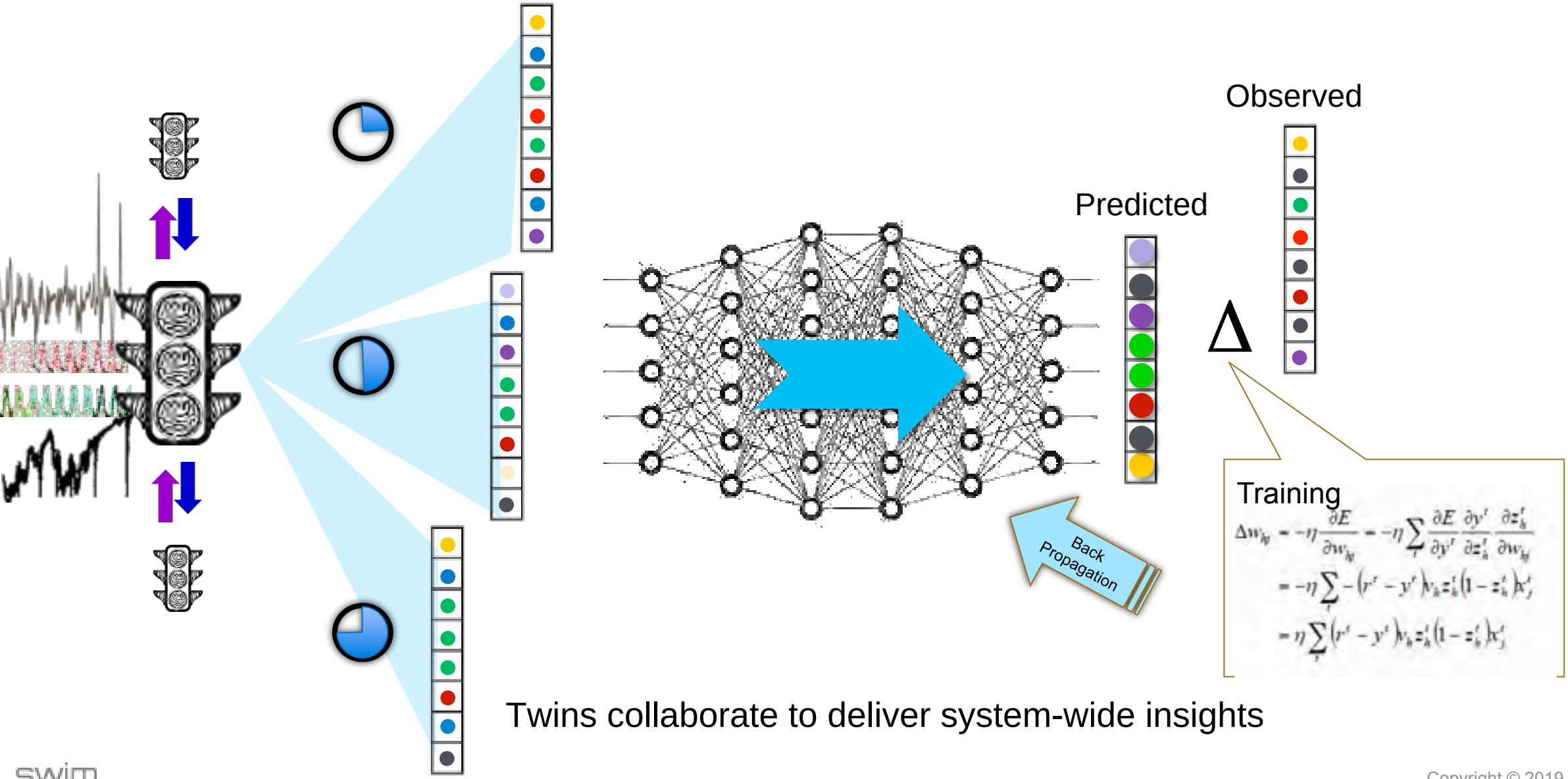


Application



Swim Fabric

swim



Twins collaborate to deliver system-wide insights

For the Win

1. Models built by data are “constructive” – they work in multiple settings with no need for re-calibration / training

2. Digital twins that learn use simpler models that can easily fit on small GPUs (eg: Jetson) or CPUs

3. Over- and under-fitting are not problematic – models are highly specific and aim to predict just a single system’s behavior

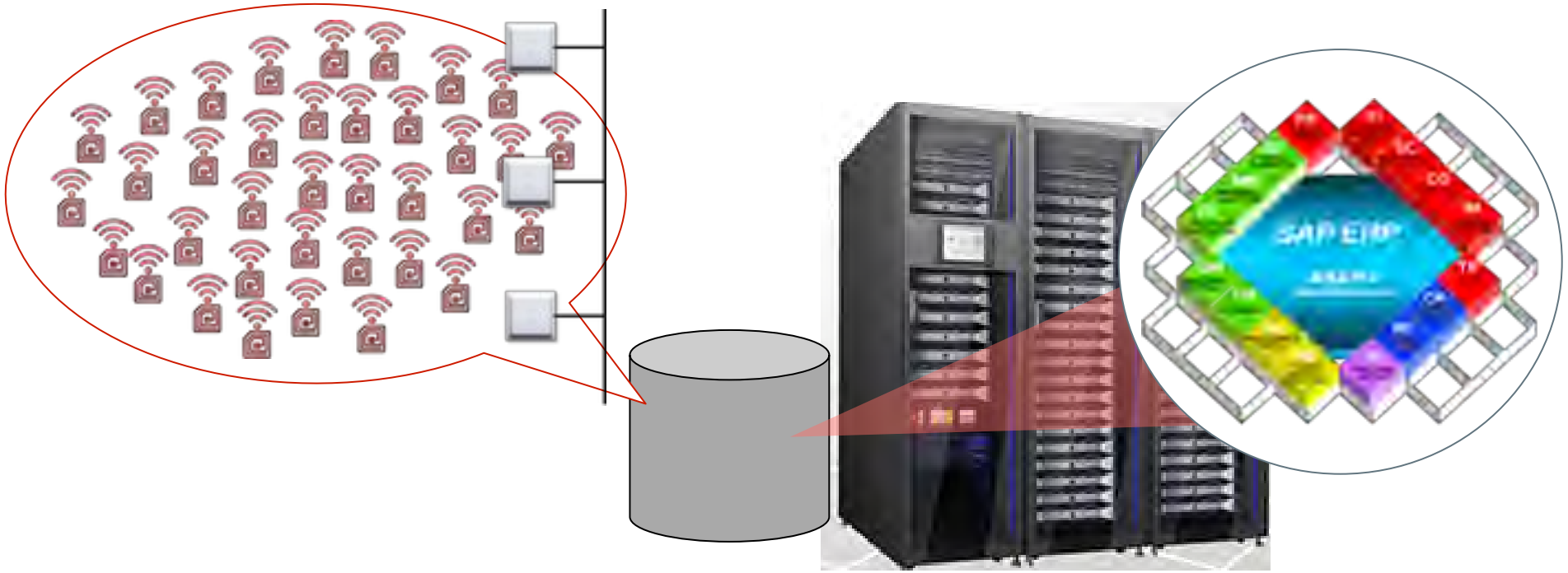
4. Learning at the edge on full-resolution data leaves less to chance

5. Train on more data than you could ever store

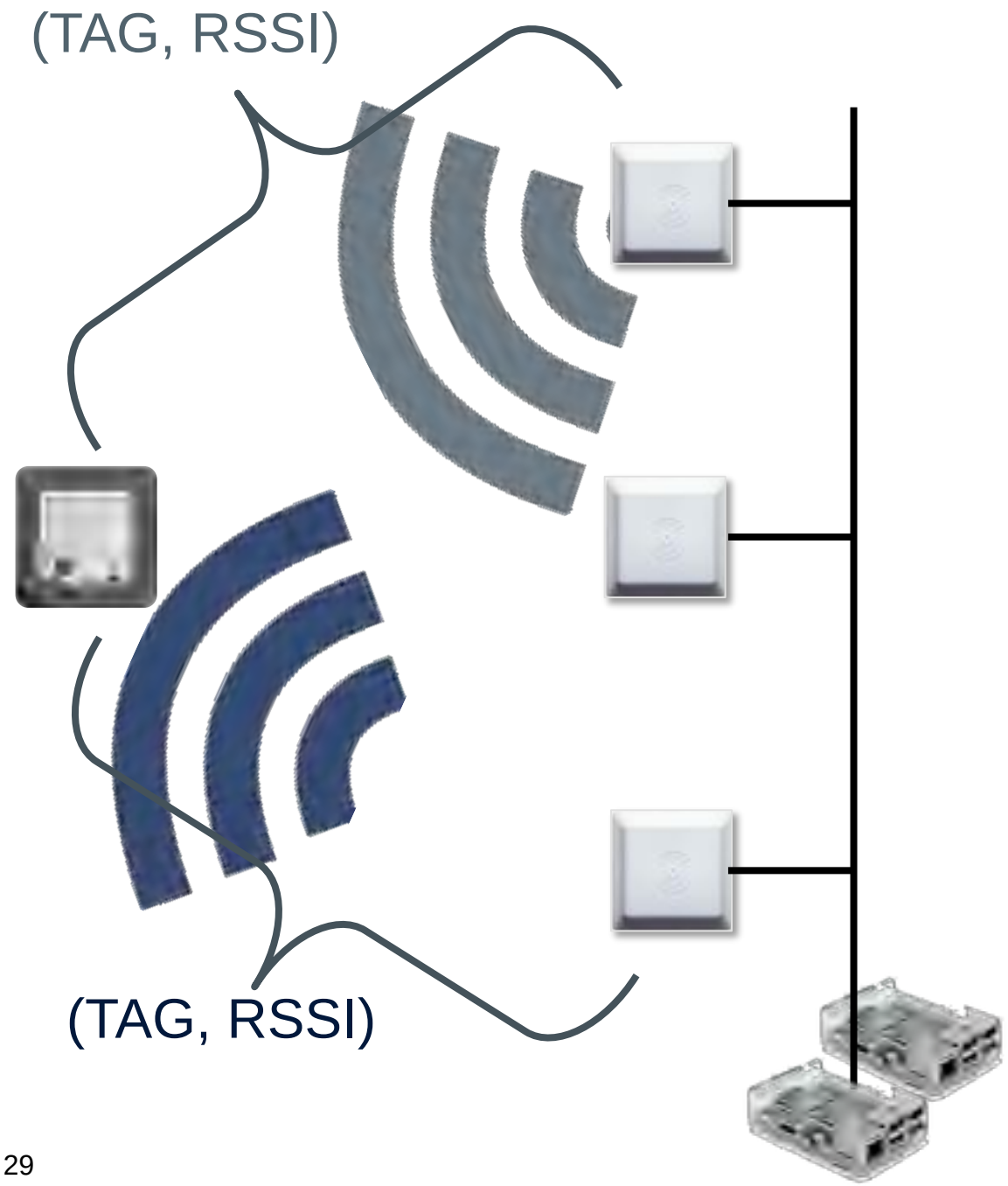
SWIM Empowers Developers

Swim Capabilities	Swim Innovations
Real-time application responses	Twins that process their data in real-time, linked to others with twin-twin backpressure
Streaming state updates	WARP streaming
Live UIs	Swim “in the browser”
Fast and easy to build	Persistence without a database, messaging without a broker, scheduling without a job manager, business logic without an app server
Economical to run	

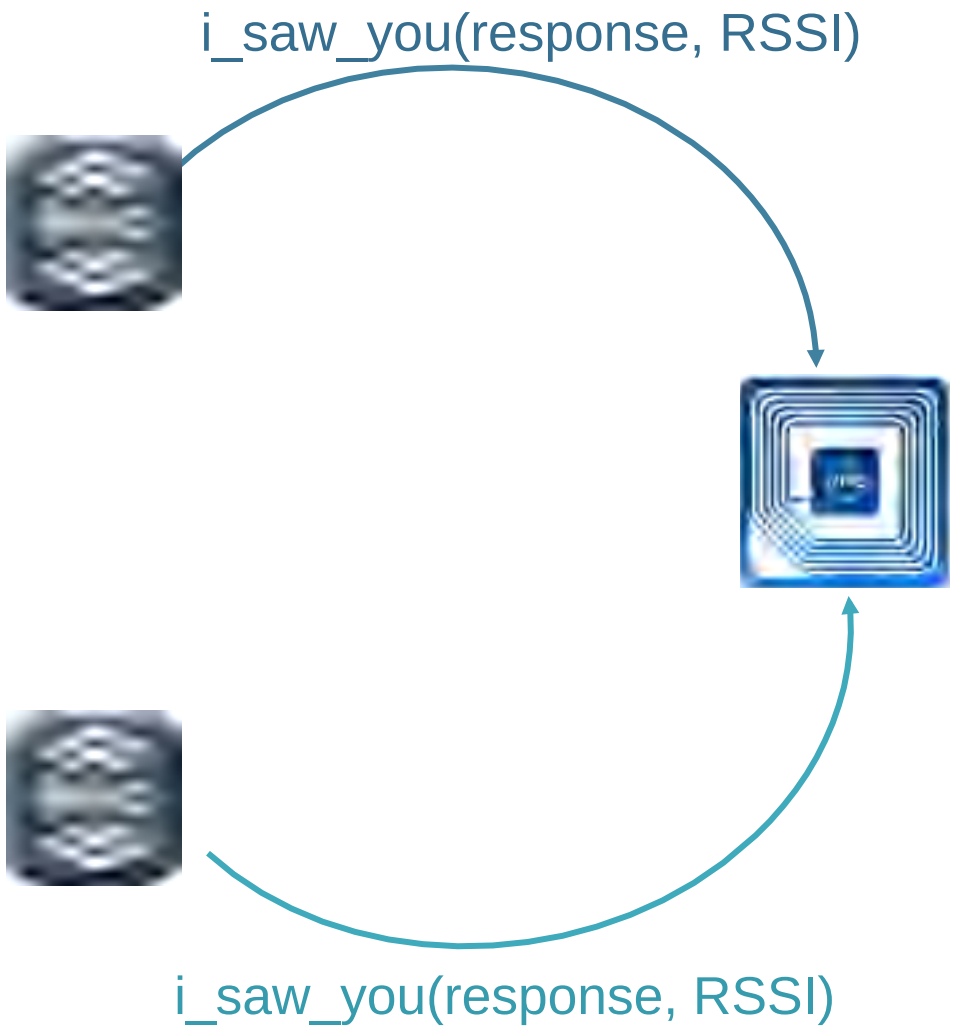
SWIM

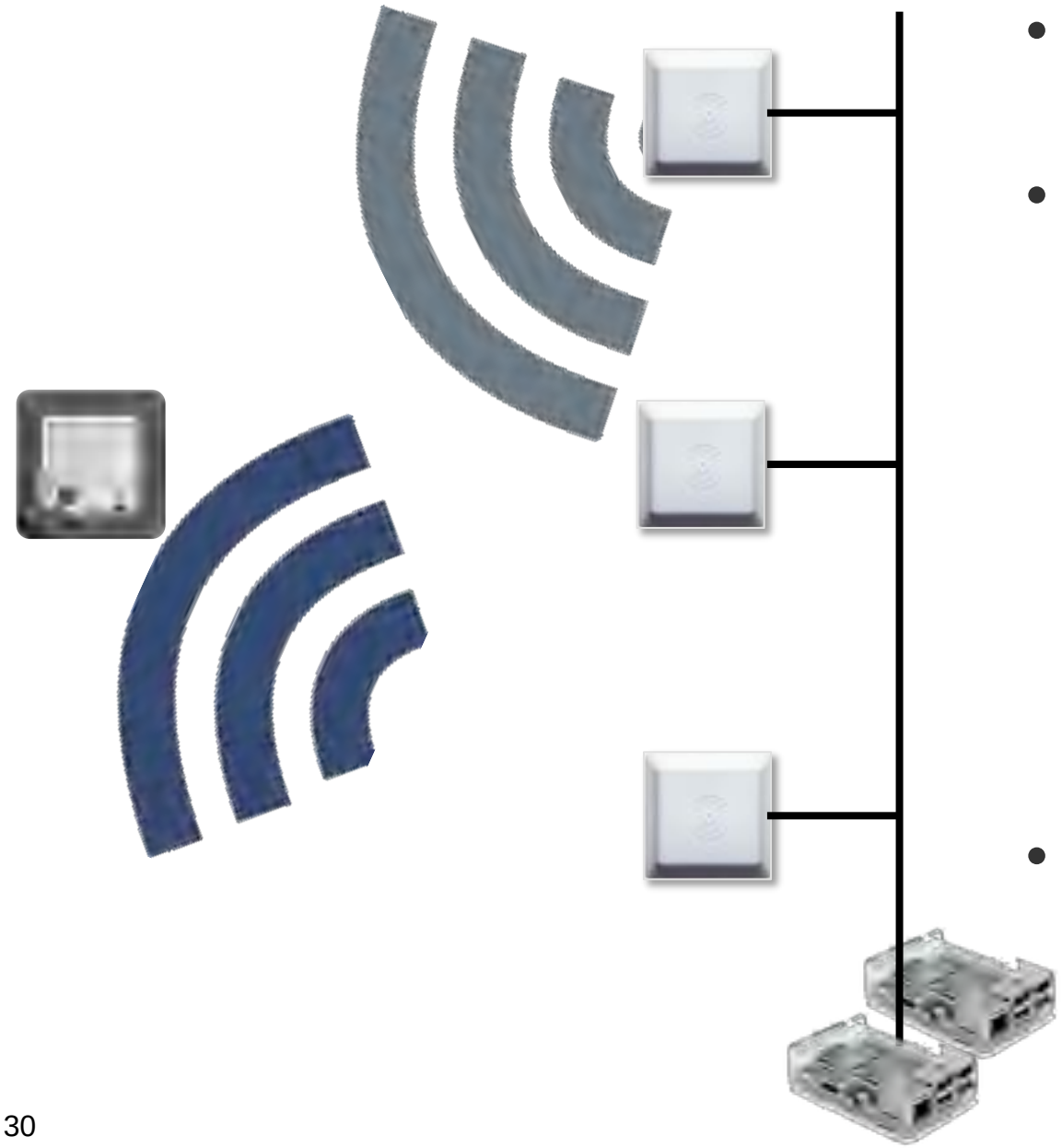


- 2000 readers and ~10,000 reads / sec
- Millions of tagged assets
- Each tag gets “seen” by multiple readers
- Tag read database of terabytes
- Computationally intense to process

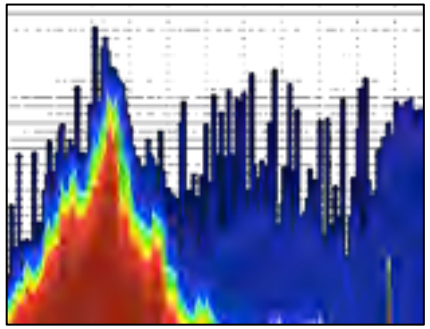


Digital Twins

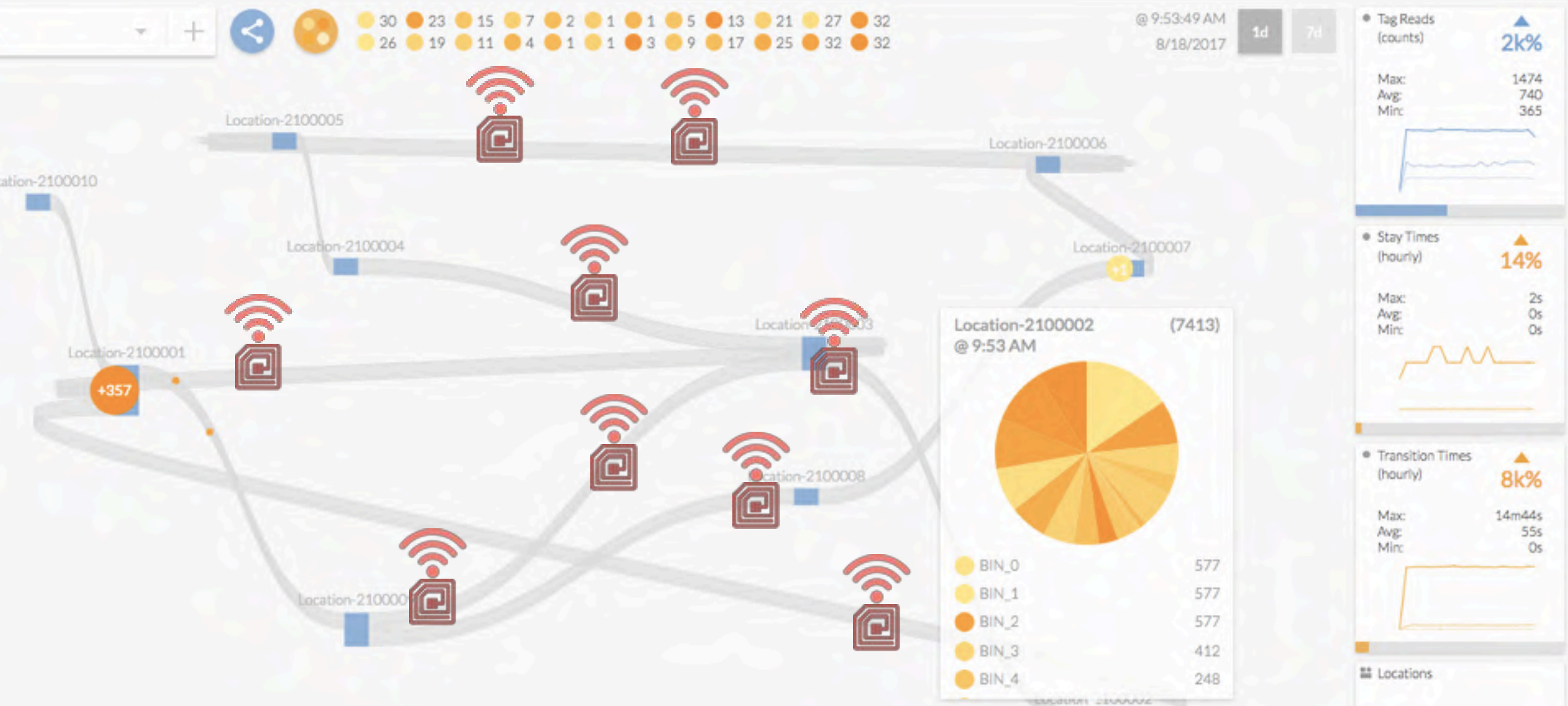




- **RSSI:** Received Signal Strength Indicator
- Signal strength variation means we need to “learn” the RF power distribution



- Then use **DeLaunay Triangulation** to compute position of each tag

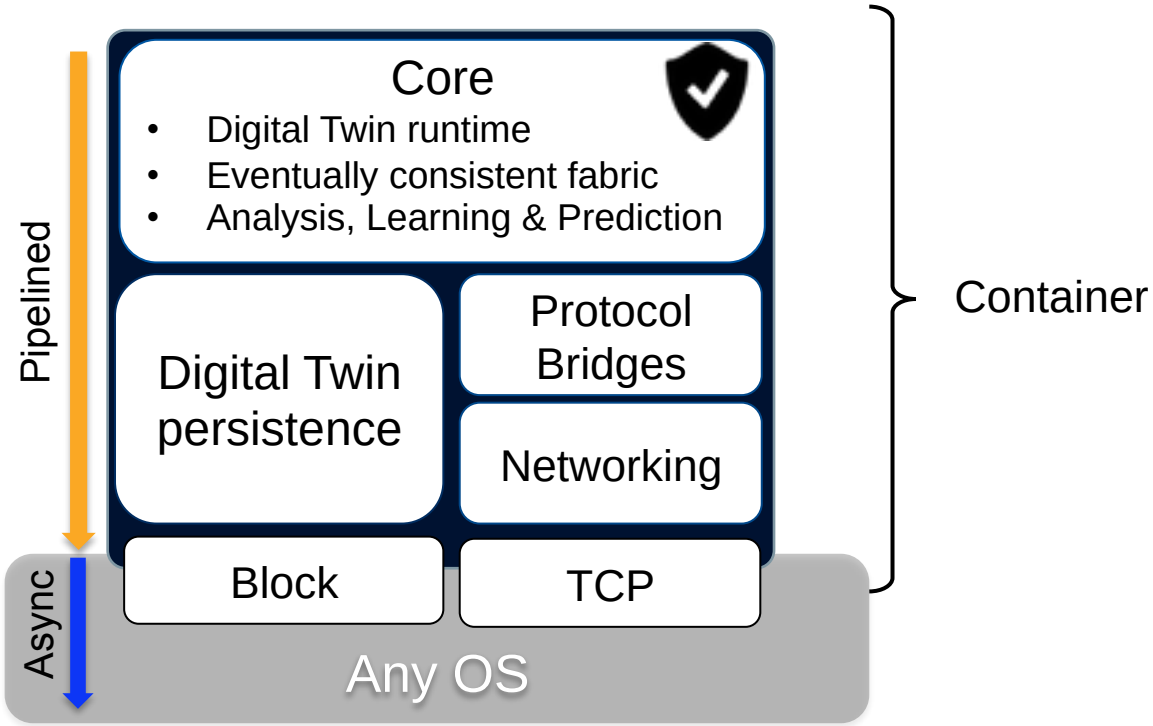


- Smart assembly line tracks & maps tags by linking to their updates
- Geo-Fence enables tracking of sub-assemblies

Swim

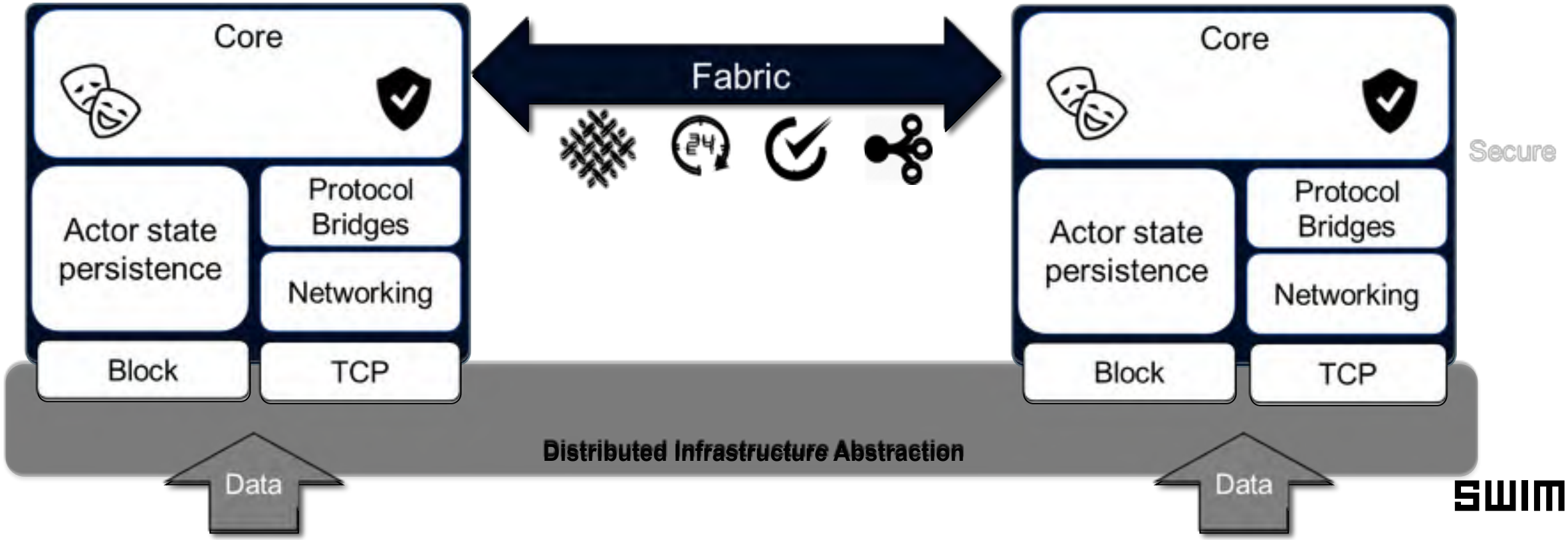
Swim Architecture

- Builds an efficient, stateful, lock-free, parallel edge data processing fabric across a set of compute instances – embedded, fog & cloud
- Swim is a 2MB library added to the JVM
- Security is fundamental – from boot to analysis and data custody



Swim Fabric

- Digital twins are stateful, persistent, active objects that process their own data
- Communication is real-time, non-blocking, and focused on eventual consistency
- Tasks automatically migrate to the optimal instance and are resilient to failure
- Instances share state changes via an eventually consistent protocol

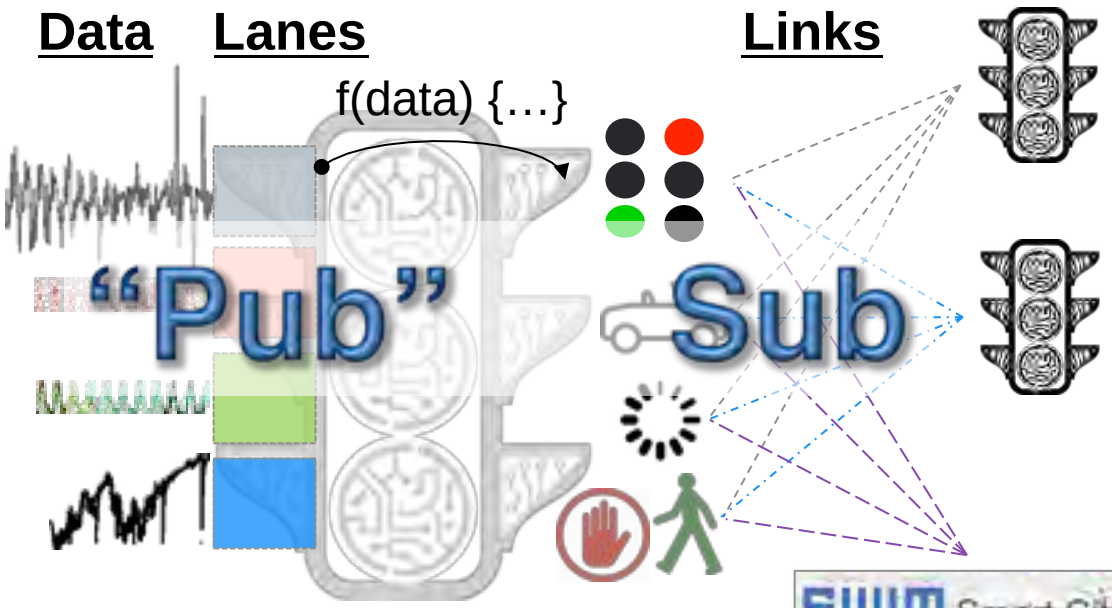


Developer Model

Fabric is a real-time runtime for distributed “web agents”

Digital Twins are stateful distributed objects created from data streams, on the fly

- **Lanes** are object members
  - Properties and methods eg: “average”
  - Hold state, have logic eg “reduce”
  - Streamed as updates over **links**
- **Links** are relationships between web agents
  - Build a graph that expresses real-world relationships`
  - “Subscribe” to Lanes and observe current state
  - Express computational constructs eg: “join” or “near”
- **Plane** is a collection of actor definitions (an app)
  - Includes naming, resolving, routing & security



Digital Twin

SWIM

Demos

- http://ripple.swim.ai
- http://traffic.swim.ai/dashboard/
- http://traffic.swim.ai/dark/