Continuous Intelligence
Apps That Always Have The Answer

Simon Crosby, CTO

swim.ai
simon@swim.ai
Big Data

⇒ “Store then analyze”
⇒ But data value is short-lived
⇒ And flows are boundless

• Expensive
• Slow
• Mismatched
Store then Analyze

Analyze & React

(maybe store later)
Continuous Intelligence

Always have the answer

Analyze, learn & predict on-the-fly

Data drives computation
⇒ Always have the answer

Predict pick & place nozzle failures
⇒ Analyze, learn & predict on-the-fly
• Find trucks that are in or approaching a banned area

• Find trucks with bad braking behavior and alert the nearest inspector

⇒ Data drives computation
Benefits

Get answers a million times faster...

Use 90% less infrastructure

Apps are easy to develop and operate

Do data science on live data
<table>
<thead>
<tr>
<th>Task</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execute an instruction</td>
<td>1 ns</td>
</tr>
<tr>
<td>Read 1MB from disk</td>
<td>20,000,000 ns</td>
</tr>
<tr>
<td>Ping US-W to US-E</td>
<td>80,000,000 ns</td>
</tr>
</tbody>
</table>
Moore's Law – The number of transistors on integrated circuit chips (1971-2018)

Moore’s law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are linked to Moore's law.
Databases...

- Can’t find *meaning* in data or build models
- Don’t track fluid real-world relationships
- Don’t analyze, learn or predict

They are poorly suited to streaming data
- Transactions & ACID properties are too heavyweight
- What’s the ‘*truth*’ and is it *useful*?
- ‘*Time*’ is fundamental, not just for ordering
- Relationships are fluid, mathematical & geospatial
- An event can cause cascading re-evaluations

Databases don’t run applications!
Event Streaming...

Streaming analytics (solved!)

polling → not real-time

Application

Client

Good event infra… but doesn’t run apps

- engine_temp: 290
- fan_temp: 188
- coolant_vol: 25
Stateful in-memory computing scales up and out

Compute on-the-fly driven by data

Use real-world context to find meaning

Use languages & ops tools everyone knows
swim: infor Things

- Creates a stateful, concurrent *Web Agent* for each data source, that continuously and statefully analyzes data from its real-world twin
- Each dynamically **links** to *related* agents, creating a fluid in-memory graph that tracks complex relationships
  - Containment, proximity - “neighbor” or ”is approaching”
  - Computed - “correlated to” or ”predicted to be within”
- Linked Agents can use each others’ state to continuously analyze, learn and predict
- And stream their insights to apps, real-time UIs etc

Get it at: swimos.org
Continuously Analyze, Learn & Predict

- Learn & Predict
- Relational Queries
- Graph Analysis
- Spark / Flink / Jupyter etc
Un-supervised Learning
Key Concepts: Agents, Lanes, and Links

- Universally addressable
- Stateful and persistent
- Cache coherent streaming APIs
- General purpose compute processes
- Low overhead (<1kB/agent)
- Self-sufficient distributed runtime
## Analogy

<table>
<thead>
<tr>
<th></th>
<th>Agent</th>
<th>Lane</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>OOP</td>
<td>Object</td>
<td>Member</td>
<td>Reference</td>
</tr>
<tr>
<td>REST</td>
<td>Endpoint</td>
<td>Method</td>
<td>Request</td>
</tr>
<tr>
<td>Database</td>
<td>Row</td>
<td>Column</td>
<td>Relation</td>
</tr>
<tr>
<td>Message Broker</td>
<td>Namespace</td>
<td>Topic</td>
<td>Subscription</td>
</tr>
<tr>
<td>Actor Model</td>
<td>Actor</td>
<td>Mailbox</td>
<td>Messages</td>
</tr>
<tr>
<td>Operating System</td>
<td>Process</td>
<td>File</td>
<td>File Handle</td>
</tr>
</tbody>
</table>
Architecture: A World Wide Web of Agents

- Streaming Data
- Event Sources
- State Changes

Real-Time UIs
Streaming APIs
Web Agents

Relational Data
NoSQL Data
Other Data

SwimOS
Edge
On-Prem
Cloud

Copyright © 2020 Swim.ai, inc.
Easy for Real Humans

Developer builds an object-oriented Java application & deploys with standard tools

Web Agents link to represent relationships and share state

... and stream their analysis and predictions in real-time over their links ...

Swim uses streaming data to build a stateful graph of concurrent Web Agents – one per source

They continuously analyze, learn & predict from their state and the states of linked Agents

... to deliver continuously updated insights to UIs, data scientists & applications
A Swim Application is an Active Graph
A Graph of Linked Web Agents
A Computational Graph That Computes as Data Flows
Where?

The “Edge” is not a place!

The “Edge” is *where your data comes from!*
Compute wherever you can – often in the cloud
Where?

Eg: Kubernetes

Digital Twin Model

Streaming Analysis, Learning & Prediction

Distributed, Stateful Edge Data Processing Fabric

Edge

Fog

Cloud
Swim Continuum: platform for continuous intelligence

➢ Distributed application runtime for real-time, data-driven computing
Where?
Where?
Continuous Intelligence

- Analyze, learn & predict on-the-fly
- Always have the answer
- React in real-time
- Get answers a *million times* faster…
- Use 90% less infrastructure
- Apps are easy to develop and run
- Do data science on live data
• 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
• **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