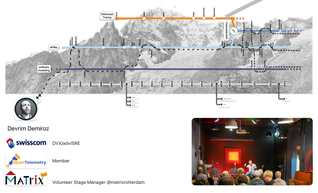


2017

Before unleashing the conclusion, lets take a look at from purposes point of view first.

It is always DNS like.. It is indeed dynamical control systems

Lets assume this model as big bang.

What is observability?

Observability is the ability to understand the internal state of a system by examining its outputs. In the context of software, this means being able to understand the internal state of a system by examining its telemetry data, which includes traces, metrics, and logs.

dynamical control systems

Essentially, with **observability**, we desire to create a **models** for different purposes.

- **predictive** model : What's Next
- **explanatory** model: Explaining Why(at,en,o,are,ich,hy)
- **exploratory** model: Searching, Finding , Understanding
- **generative** model: Making New
- **descriptive** model: Stories to Tell
- **prescriptive** model: Reflexes towards certain situations
- ....

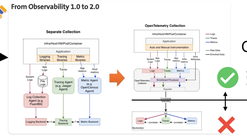
Signals

What is OpenTelemetry?

A short explanation of what OpenTelemetry is and isn't.

2019 2020 2021 2022 2023 2024

From Observability 1.0 to 2.0



OTel

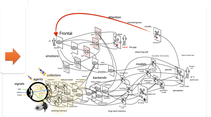
OpenTelemetry is an Observability framework and toolset designed to create and manage telemetry data such as traces, metrics, and logs. Crucially, OpenTelemetry is vendor- and tool-agnostic.

OpenTelemetry is not an observability backend like Jaeger, Prometheus, or other commercial vendors. OpenTelemetry is focused on the generation, collection, management, and export of telemetry.

vendor & open source products

From Observability 1.0 to 2.0

From Observability 2.0 to 3.0



I know the guy who diagrammed this.He is obsessed with OTel, looks like he is losing his mind 😂

In this neurological metaphor, signals — comprising metrics, logs, and traces — are the stimuli we perceive. They are then processed by various agents and collectors that mirror our sensory organs, translating these signals into a form that our brain, or in this case, our observability systems, can understand and use. —

what he actually said (auto translation from Portugese) .🙏

All metaphors are incorrect because they attempt to explain something using something else that is definitely not the thing itself. Somehow "useful".

I scoped overall concept to Visual Cortex Analogy for simplicity. Impressionist as diagram is incomplete, subject to change, intentionally left open-ended...

a collective intelligence story....

Thousands of engineers , software products and machines , we all unconsciously form and function parts of a brain at the very end. Emerged collective intelligence in observability is simply yet another form of brain.

ENTRY

emotions

Service Level Objectives (SLOs) are the emotional responses of the system.

They are the desired states of the system, which are defined by the user or the system itself.

SLOs are the ultimate goals of the system, and they are the key indicators of the system's health and performance.

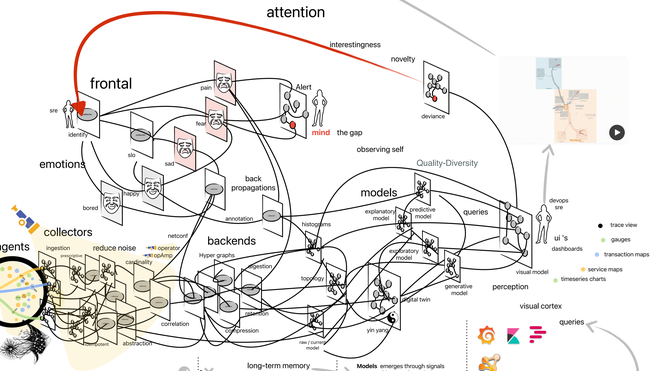
back propagations & actions

According to identified SLOs, the system takes actions to maintain or improve its performance. Alerts are triggered and so forth.

And/or reconfigures the model to adapt to the new conditions. These correspond to opAmp, operator in OpenTelemetry currently as well netconf in telco industry as examples.

Signals are observables about a system.

Signals are the critical pieces of information that shed light on the system under observation. These typically include logs, which record discrete events; metrics, which are numerical data points that track the operation of the system over time; and traces, which document the journey of requests through a distributed system.



agents

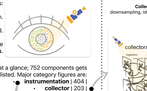
| The Eyes of Observability through OTel retina |

Just like the retina has receptors of different kinds and subkinds:

Rods: Low light vision.

Cones: Color vision, with three types; Blue/Green/Red.

Logging, **metrics**, **tracing**, and other types of **signals** are collected through various types of OTel **Agents**.



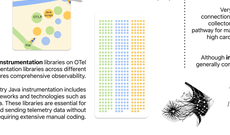
OpenTelemetry **Registry** at a glance; 752 components gets listed. Major category figures are:

- **instrumentation** | 404 |
- **collector** | 203 |
  - exporter | 72 |
  - tools | 20 |
  - resource | 14 |
  - otel | 34 |

Instrumentation Libraries

| Types of receptors|

Instrumentation libraries in OpenTelemetry, like those for Java or Apache HTTP, are analogous to retinal cells.



OpenTelemetry offers **404 instrumentation** libraries on OTel Registry. Wide range of instrumentation libraries across different platforms and languages ensures comprehensive observability.

For example, the OpenTelemetry Java instrumentation includes libraries for popular frameworks and technologies such as MongoDB, Spring, and Kafka. These libraries are essential for automatically capturing and sending telemetry data without requiring extensive manual coding.

Various types of telemetry data gathered suches language agents, library level replacements, k8s, OSs, containers, cloud resources etc.

High cardinality sensory data enters via agents into system observability.

working memory

Collector significantly reduces cardinality, sampling, downsampling, idempotent filters, correlation and abstraction ....

- **topology** can be calculated as early efficient possible.

92 receivers
47 exporters
24 processors
20 connectors
20 extensions,
3.6K stars.
As of 21 Feb 2024

Very similar to CNN , OTel agents first layer connections are wired into many variations of OTel collectors both vertical and horizontal along the pathway for mainly **reducing the noise** over incoming high cardinal data and executes many more vital functions.

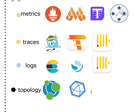
Although **ingestion** is not a brain function, but it is generally common to define collector layer similar to data pipelines.

collectors

| Short term (Working) memory |

OTel

long-term memory



open source products , CNCF , AGPL ,....

backends

Models emerges through signals to provide an optimum balance point towards perception.

The threshold should be just good enough to extract the meaning from the model.

Although solid static models are available for specific use cases, this is also the gray area that appears mostly when a user starts to query from the user interfaces, generally consisting of flexible dashboards.

models

In interaction with the data sources and the actual system under observation, the user views different models for different purposes.

- predictive model : What's Next
- explanatory model: Explaining Why(at,en,o,are,ich,hy)
- exploratory model: Searching, Finding , Understanding
- generative model: Making New
- descriptive model: Stories to Tell
- prescriptive model: Reflexes towards certain situations
- ...