
OPENTELEMETRY
METRICS 101
Reese Lee, New Relic

Reese Lee

Developer Relations Engineer
New Relic

- Previously Technical Support
- OpenTelemetry End User WG
  - Adoption and implementation
  - Feedback loop to improve the project
- Malaysia → Pacific Northwest
- Summited Mt. Hood (11,249’)
- Visited 14 countries

AGENDA

01 METRICS OVERVIEW

02 OPENTELEMETRY OVERVIEW

03 METRICS DIP

04 WHAT'S NEXT?

01

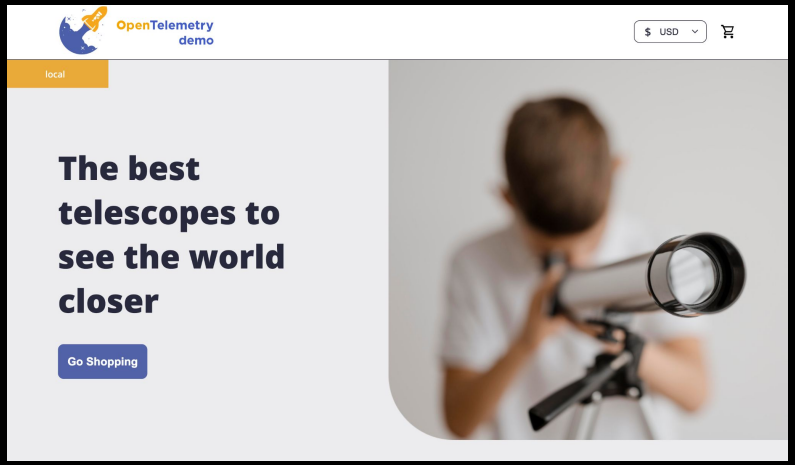
METRICS OVERVIEW

METRICS OVERVIEW

1. WHAT IS A METRIC?
2. WHY ARE METRICS USEFUL?

WHAT IS A METRIC?

A metric is a measurement about a service captured at runtime. Metrics represent aggregations of multiple measurements, and can be used to identify trends.



- Throughput
- Response time
- Error rate
- CPU utilization
- Number of active users
- Total processed orders
- Total processed orders of a specific item

} METRICS!

WHY ARE METRICS USEFUL?

DATA VOLUME REDUCTION

Reducing the volume of data

PERFORMANCE

Monitoring your system

ALERTS

Alerting on breached SLOs

VISUALIZATION

Powering graphs, charts, and dashboards

02

OPENTELEMETRY OVERVIEW

OPENTELEMETRY OVERVIEW

1. WHAT IS OPENTELEMETRY?
2. WHY OPENTELEMETRY FOR METRICS?

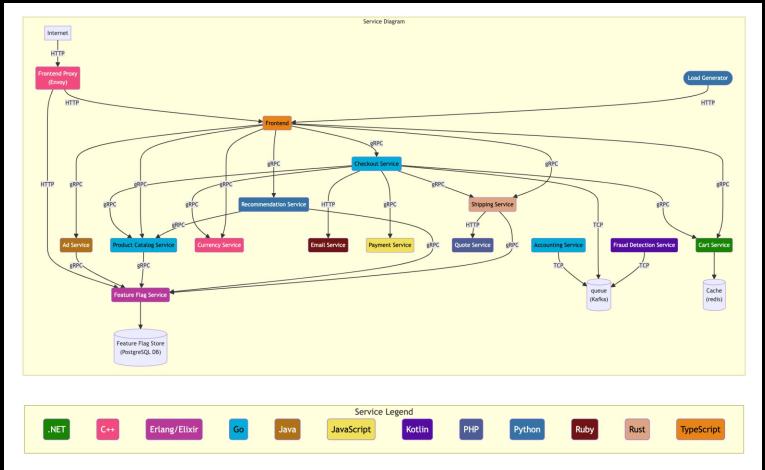
WHAT IS OPENTELEMETRY?

OpenTelemetry is...

- An observability framework built on an open standard
- The merging of OpenCensus and OpenTracing in 2019
- 2nd most active CNCF project in terms of contributions (after Kubernetes)
- Aims to standardize instrumentation and telemetry generation, collection, and transmission

WHAT IS OPENTELEMETRY?

OpenTelemetry provides a set of APIs and SDKs, tools and components (such as the Collector), instrumentation libraries, semantic conventions,and a protocol (called OTLP).



- Java
- .NET
- Python
- Ruby
- ... and more

} ONE STANDARDIZED SET OF TOOLS

Why OpenTelemetry for Metrics?

ABILITY TO CONNECT METRICS TO OTHER SIGNALS

- Exemplars
- Enrich metrics attributes via Baggage and Context

OPENCENSUS MIGRATION TO OPENTELEMETRY

- Original goal of OpenTelemetry (OpenCensus + OpenTracing

WORKS WITH EXISTING METRICS INSTRUMENTATION PROTOCOLS AND STANDARDS

- Minimum goal: Prometheus and Statsd

Freedom from vendor lock-in!

03

METRICS DIP

METRICS DIP

1. SESSION SCOPE
2. METRICS IN OPENTELEMETRY
3. ARCHITECTURE
4. METRIC INSTRUMENTS, TYPES, AND USE CASES
    a. What is an instrument?
    b. What instruments does OpenTelemetry provide?
    c. Why is instrument selection important?
    d. How do I choose an instrument?

SESSION SCOPE

10% **THIS SESSION**
High-level overview of metrics concepts

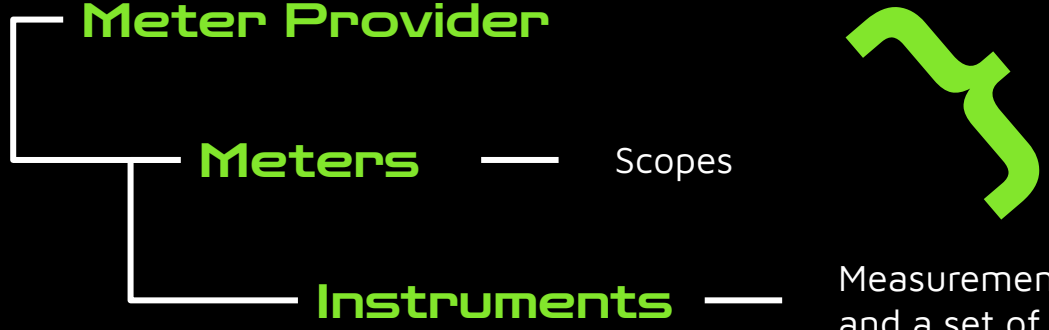
40% **DEEP DIVE**
There is so much more we could get into!

50% **EVERYTHING ELSE!?**
And perhaps even beyond that... (e.g., implementation)

METRICS IN OPENTELEMETRY

API
Used to instrument code

SDK
Used to implement the API



METRICS IN OPENTELEMETRY

AGGREGATION

The process of combining multiple measurements into a single point

TEMPORALITY

Related to whether the reported values of additive quantities include previous measurements

- Cumulative
- Delta

MONOTONICITY

- Monotonic
- Non-monotonic

Related to whether the value is always increasing, or always increasing and decreasing at the same time

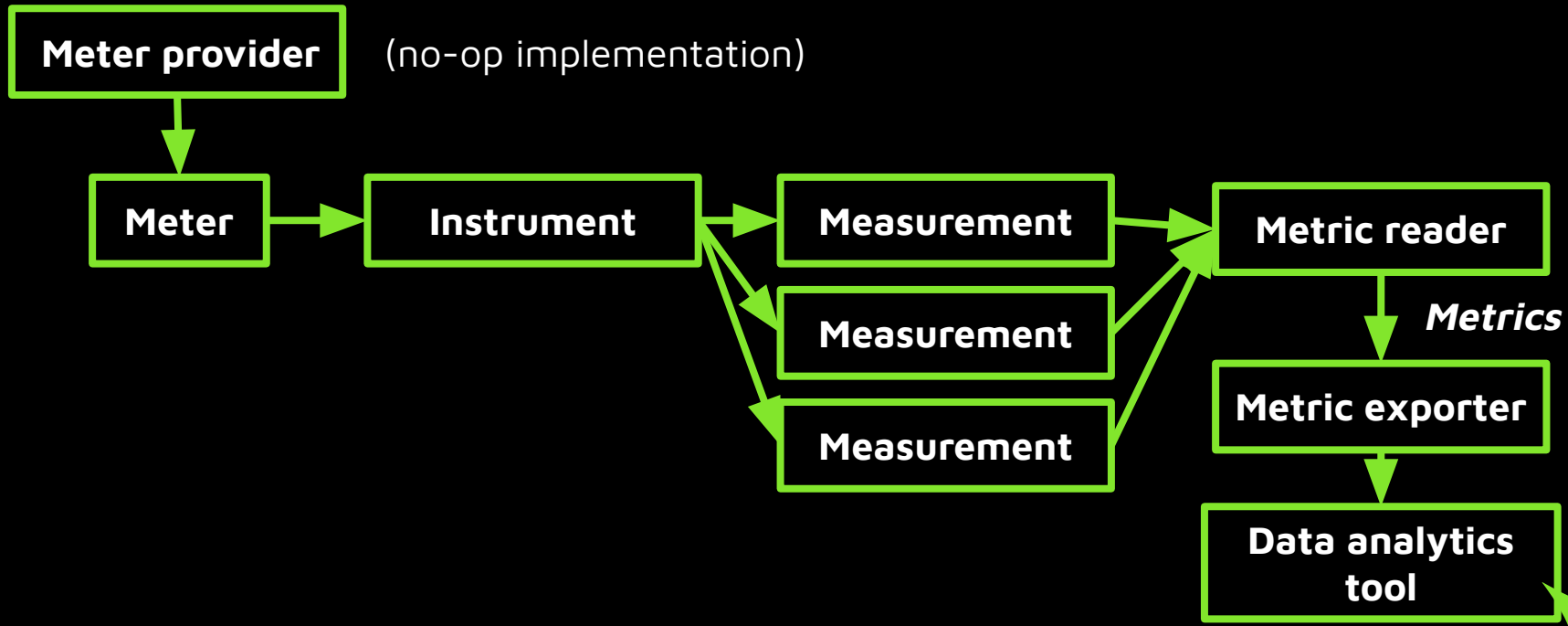
DIMENSION

An attribute associated with a metric, can be used to filter and aggregate data

CARDINALITY

How many unique dimensions are associated with a metric

ARCHITECTURE



METRIC INSTRUMENTS, TYPES, AND USE CASES

What is an instrument?

Instruments report measurements and have the following fields:

- Instrument name
- Kind
- Measure of unit (optional)
- Description (optional)

WHAT INSTRUMENTS DOES OPENTELEMETRY PROVIDE?

	INSTRUMENT	SYNCHRONOUS	ADDITIVE	MONOTONIC	AGGREGATION
1	Counter	■	■	■	Sum
2	Up/down counter	■	■	□	Sum
3	Async counter	□	■	■	Sum
4	Async up/down counter	□	■	□	Sum
5	Histogram	■	□	□	Histogram
6	Gauge	□	□	□	Last value

WHY IS INSTRUMENT SELECTION IMPORTANT?

Default aggregation reflects the intended use of the measurements

Instrument type

→ measurements are aggregated

→ the type of metric that is exported

→ impacts the way you can query and analyze it.

How do I choose an instrument?

Analysis

How do you want to analyze the data?

Sync or async

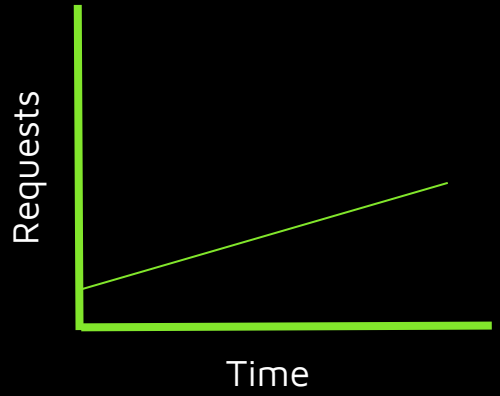
Do you need the measurement synchronously, or can it be reported on a set interval?

Monotonicity

Are the values monotonic?

COUNTER

Synchronous	Additive	Monotonic	Default aggregation	Example usage
✅	✅	✅	**Sum**	**Number of bytes sent, total orders processed, total cart adds, total cart add failures, total checkouts, total checkout failures**

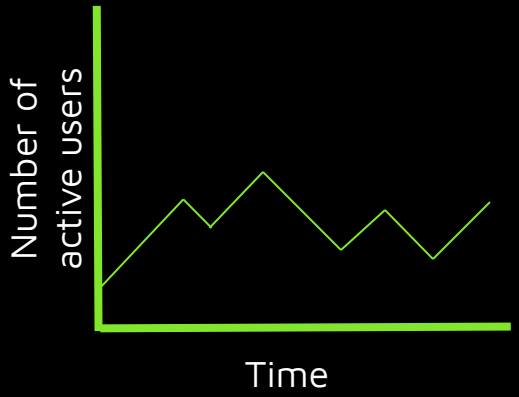


Use when...

- you want to count things and compute the rate at which things happen
- the sum of the things is more meaningful than the individual values

UP/DOWN COUNTER

Synchronous	Additive	Monotonic	Default aggregation	Example usage
✅	✅	❌	Sum	Number of open connections, number of active users, queue size, memory in use

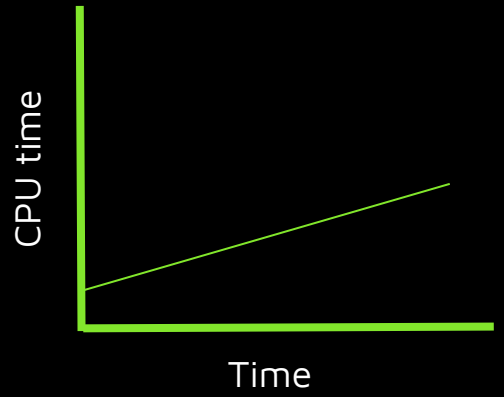


Use when...

- you want to process positive and negative increments

ASYNC COUNTER

Synchronous	Additive	Monotonic	Default aggregation	Example usage
❌	✅	✅	**Sum**	**CPU time, cache hits and misses, total network bytes transferred**

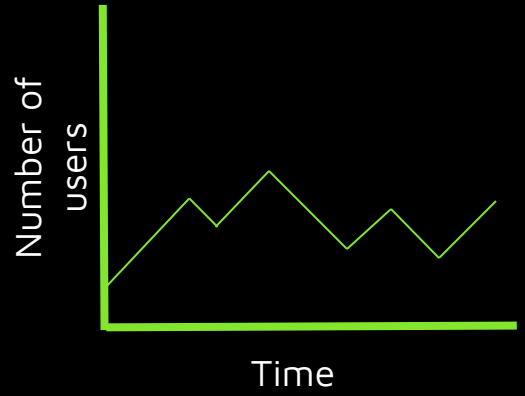


Use when...

- you need a sum of your measurements, but they may be too expensive to report synchronously, or it is more appropriate to record on set intervals

ASYNC UP/DOWN COUNTER

Synchronous	Additive	Monotonic	Default aggregation	Example usage
❌	✅	❌	**Sum**	**Memory utilization, process heap size, number of active shards, changes in the number of active users**

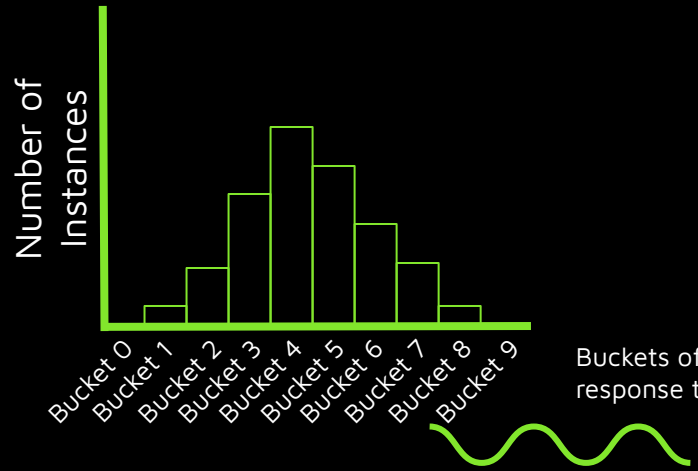


Use when...

- you need a non-monotonic additive counter to report on set intervals

HISTOGRAM

Synchronous	Additive	Monotonic	Default aggregation	Example usage
✅	❌	❌	**Explicit bucket histogram**	**HTTP server response times, client duration, request rate**

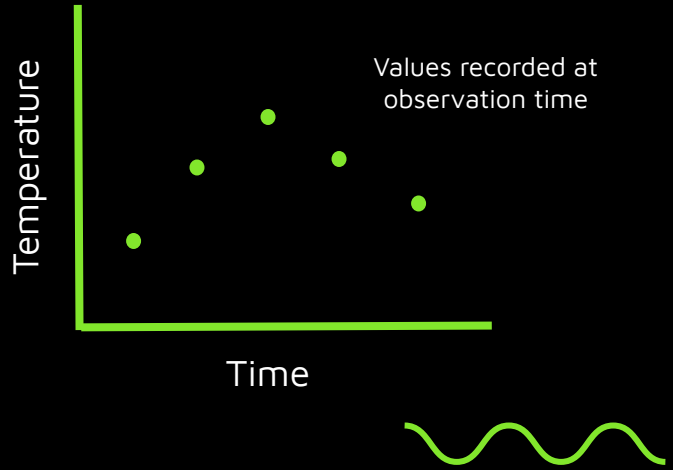


Use when...

- you want to analyze the distribution of measurements to identify trends
- you want to calculate the min, max, and average response time

GAUGE

Synchronous	Additive	Monotonic	Default aggregation	Example usage
✗	✗	✗	Last value	CPU utilization, temperature of hardware at this point in time, average memory consumption



Use when...

- you want to report data that's not useful to aggregate across dimensions and you have access to measurements asynchronously
- you want finer-grain control of when a non-additive measurement is made, particularly when its purpose is a distribution

METRIC INSTRUMENTS, TYPES, AND USE CASES

What is an instrument?

Instruments report measurements and have the following fields:

- **Instrument name** telescopes_sold
- **Kind** counter
- **Measure of unit (optional)** telescope
- **Description (optional)** “Total telescopes sold”

METRICS IN OPENTELEMETRY

VIEW

- Allows you to customize the metrics output by the SDK:
  - Process or ignore instruments
  - Override aggregation strategy
  - Attributes

04

WHAT'S NEXT?

WHAT'S NEXT?

1. RECAP
2. WHAT TO EXPLORE NEXT?
3. CREDITS, REFERENCES & CONTACT INFO

RECAP

1. What a metric is, and why they’re useful for observability
2. What OpenTelemetry is, and the utility and customization options it provides in metric generation and collection
3. Metric concepts as they apply in OpenTelemetry
4. OpenTelemetry metric instruments, and how to choose one

	Synchronous	Additive	Monotonic	Default aggregation	Example usage
Counter	✅	✅	✅	Sum	Number of bytes sent, total orders processed
Up down counter	✅	✅	❌	Sum	Number of open connections, number of active users
Histogram	✅	❌	❌	Histogram	Response times, search results latency
Async counter	❌	✅	✅	Sum	Cache hits and misses, CPU time
Async up down counter	❌	✅	❌	Sum	Memory utilization, number of active users
Gauge	❌	❌	❌	Last value	CPU utilization, hardware temperature

WHAT TO EXPLORE NEXT?

- Instrumentation and implementation - try it out yourself!
- Views API
- Data point types
- Adding metric attributes (or dimensions)
- Push- vs pull-based exporting
- Application runtime metrics
- OpenTelemetry collector metrics processors
- Infrastructure metrics
- ... and so much more!

CREDITS & REFERENCES

CREDITS

- Jack Berg, New Relic
- Vijay Samuel, eBay

REFERENCES

- Exponential Histograms: Better Data, Zero Configuration – Jack Berg
- Cloud-Native Observability with OpenTelemetry – Alex Boten
- OpenTelemetry docs
- OpenTelemetry Metrics Primer for Java Developers – Asaf Mesika

THANK YOU!

@reesesbytes

Reese Lee on CNCF Slack

Booth #212!