CATS Metrics Definition

draft-ietf-cats-metric-definition-03

https://github.com/ietf-wg-cats/draft-ietf-cats-metric-definition/commits/main/

https://datatracker.ietf.org/doc/draft-ietf-cats-metric-definition/

Kehan Yao, China Mobile

Cheng Li, Huawei Technologies

L. M. Contreras, Telefonica

Jordi Ros-Giralt, Qualcomm Europe, Inc

Hang Shi, Huawei Technologies

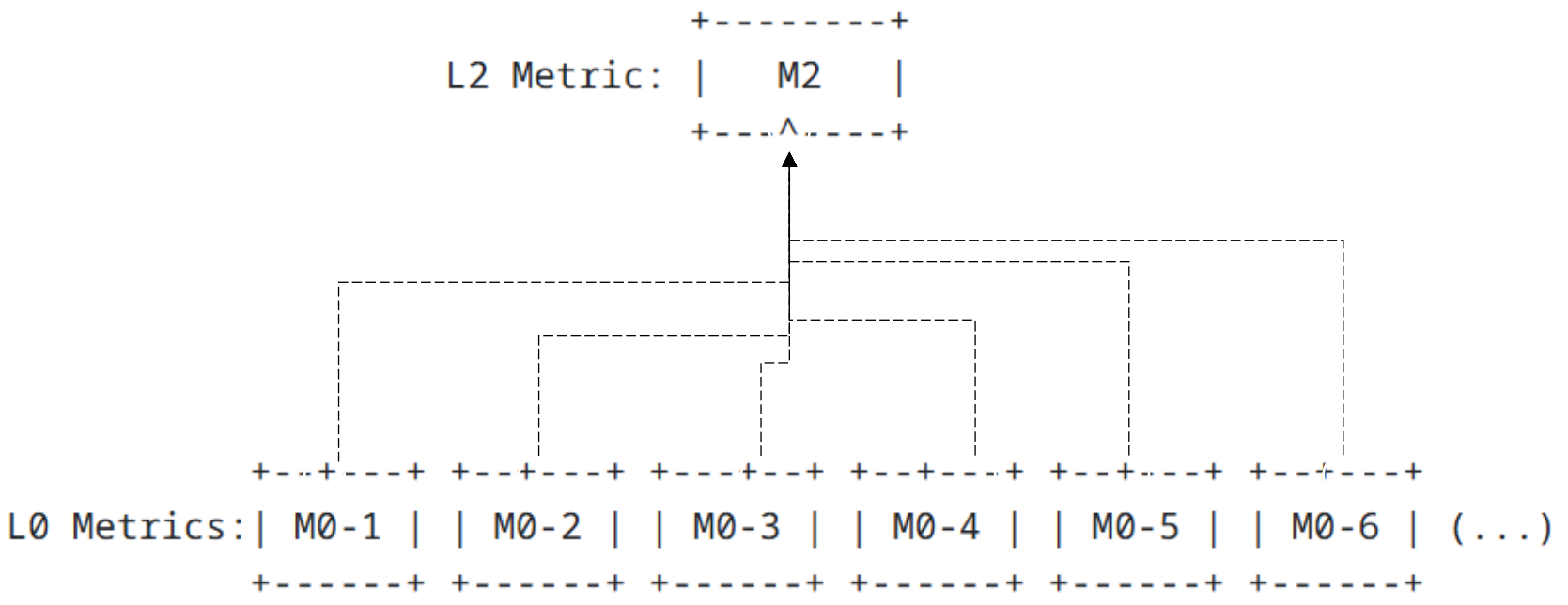
Updates Summary Since IETF 122

- **Address comments from Julien and Dirk (Minutes of IETF 122)**
  - Add a new sub-section “Design Principles - Why Three Metric Levels?” in Section 3 to better align with the requirements draft, as well as providing explanations on why we don’t define more or less metric levels
- **Address comments from Chairs (Minutes of IETF 122)**
  - Removed the part “Implementation Guidance on Using CATS Metrics”, since this part has been added into the CATS framework document.
- **Section 5. Comparison among Metric Levels.** Analysis of encoding complexity, extensibility, and stability.

Design Principles - Why Three Metric Levels?

- **Conceptually, it is necessary to define at least two fundamental levels of metrics:**
  - One comprising all raw metrics
  - The other representing a simplified form—consisting of a single value that encapsulates the overall capability of a service instance.
- **Considering extending a bit of flexibility across various CATS use cases:**
  - Need balanced approaches - adding metric levels
  - But not adding too many levels - trade-offs among encoding complexity, accuracy, scalability, extensibility.

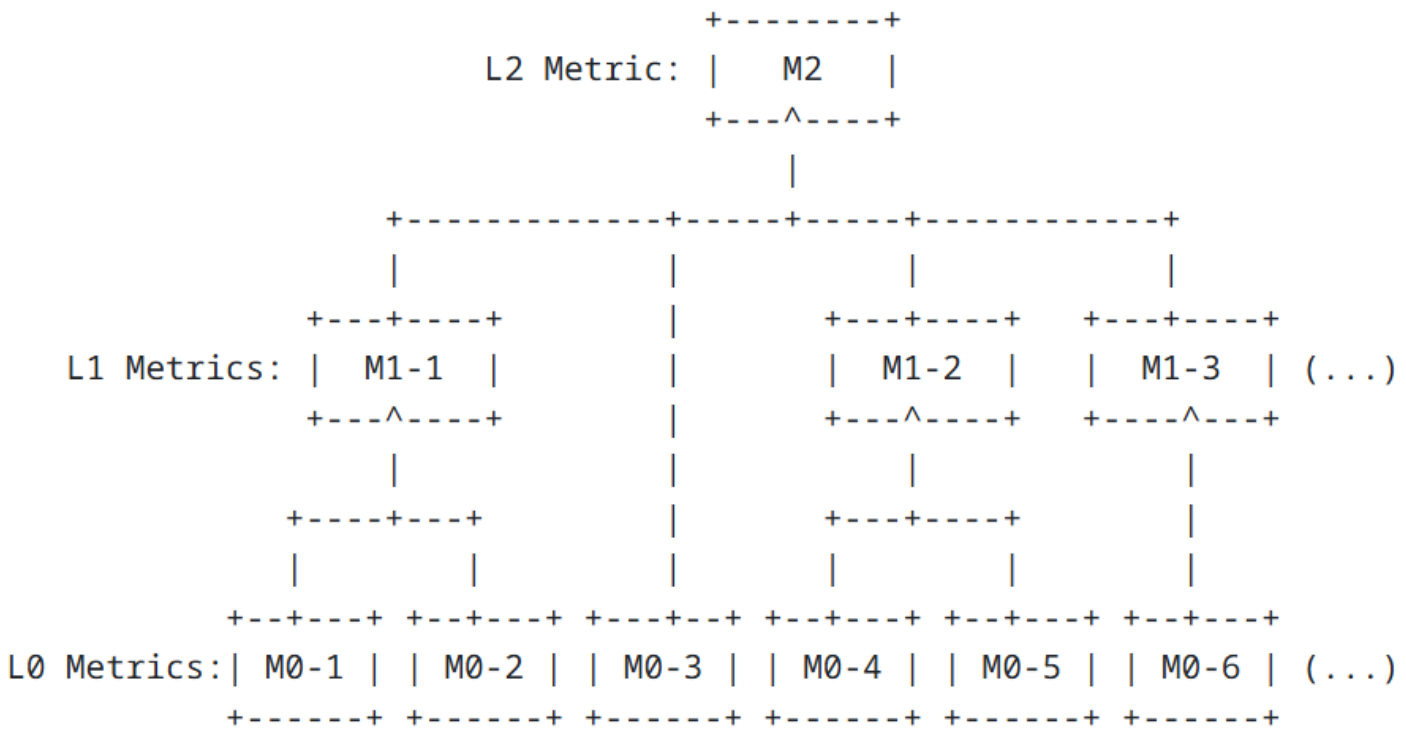
Design Principles - Why Three Metric Levels?



One single top metric

Hundreds or thousands of L0 metrics

Design Principles - Why Three Metric Levels?



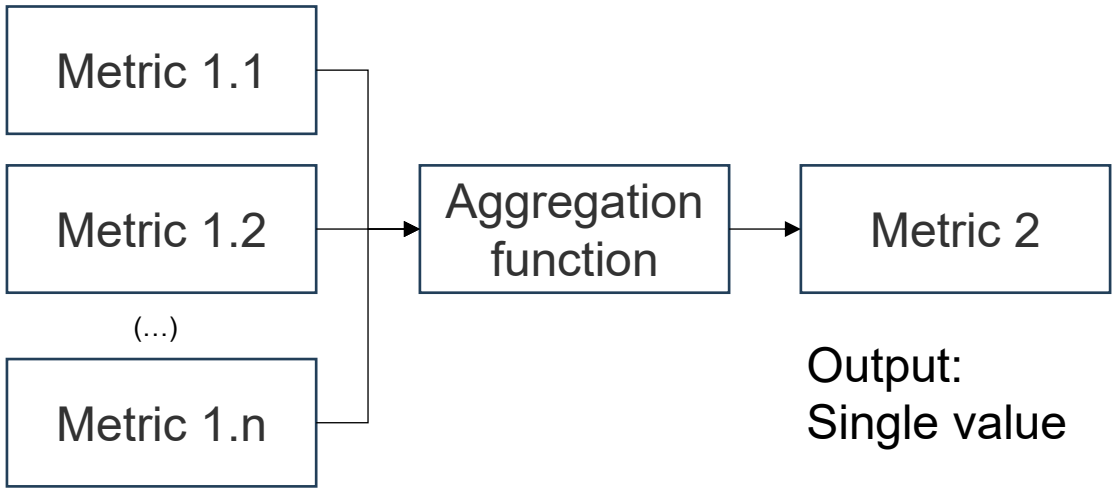
One single L2 metric

A reduced number of L1 metrics: enables significant flexibility with small additional cost

Hundreds or thousands of L0 metrics

Aggregation and Normalization Functions

Aggregation function:



Input:
Multiple values

Output:
Single value

Example: mean average

Normalization function:

Input:
Value with units

Output:
Unitless score

Example: sigmoid function

Discussions

https://mailarchive.ietf.org/arch/msg/cats/4yyZWIVo_j_PIFeTePaRwiz-3pw/

1. Should we introduce a new L1 category called "Service Metrics"?

Currently, L1 includes three resource-level metric types: computing, communication, and composed. However, some services may need to be scheduled based on service-level metrics like system availability, Queries Per Second (QPS) or Tokens Per Second (TPS). Do you agree that we should normalize and group these under a new L1 category for service metrics?

2. Should this document standardize metric sizes?

Currently, the document only provides examples for each metric level. Should we standardize the fields and their sizes—especially for L1 and L2—to ensure consistency?

Next Steps

- Facilitate discussions over these issues in the mailing list
- Refine the document and update soon.

THANKS!

Report on hackathon results

Luis M. Contreras

(on behalf of the hackathon team)

CATS Metrics Definition
draft-ietf-cats-metric-definition-03

Alejandro Muñiz, Luis M. Contreras, Guillermo Sánchez | *Telefónica GCTIO*

Alex T. de Cock Buning | Universidad Carlos III de Madrid

David Remón | Eclipse Foundation

Alberto del Rio | Universidad Politécnica de Madrid

IETF Hackathon @ IETF 123, Madrid

CODECO
Cognitive Decentralised Edge Cloud Orchestration

Joint exposure of compute and network metrics for path selection

- Enhancement to the CODECO framework
- Collects real-time network performance and infrastructure metrics
- Goal:
  - 3 aggregation levels to reach a view adapted to a custom optimization problem.
  - Taking decisions with a final single scoring.

CODECO
Cognitive Decentralised Edge Cloud Orchestration

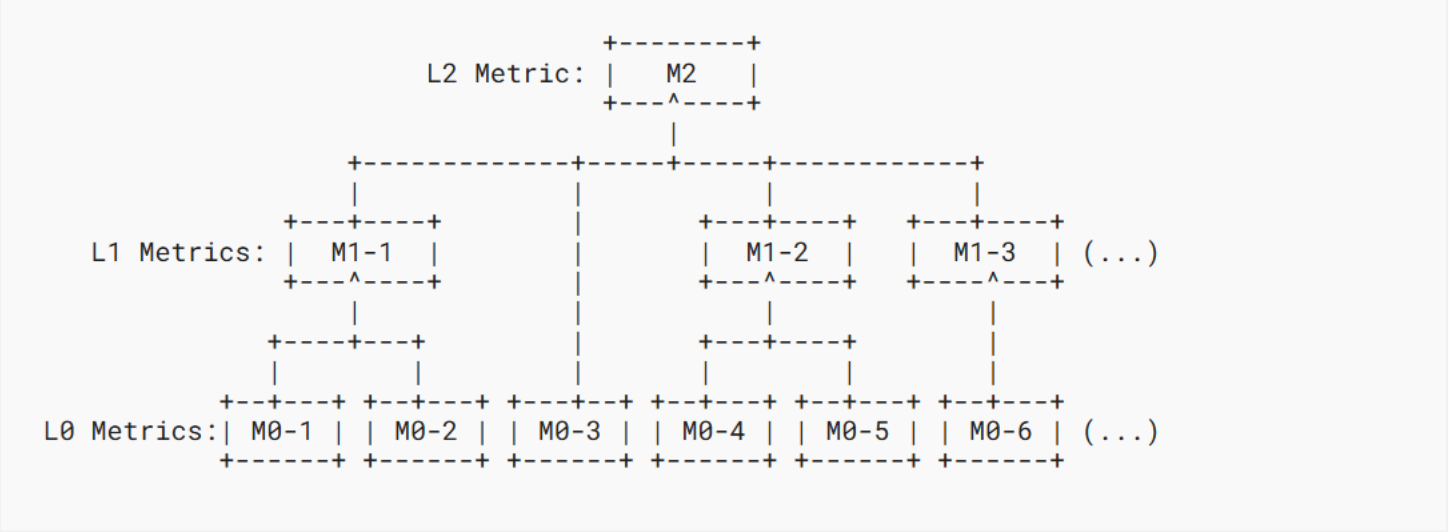


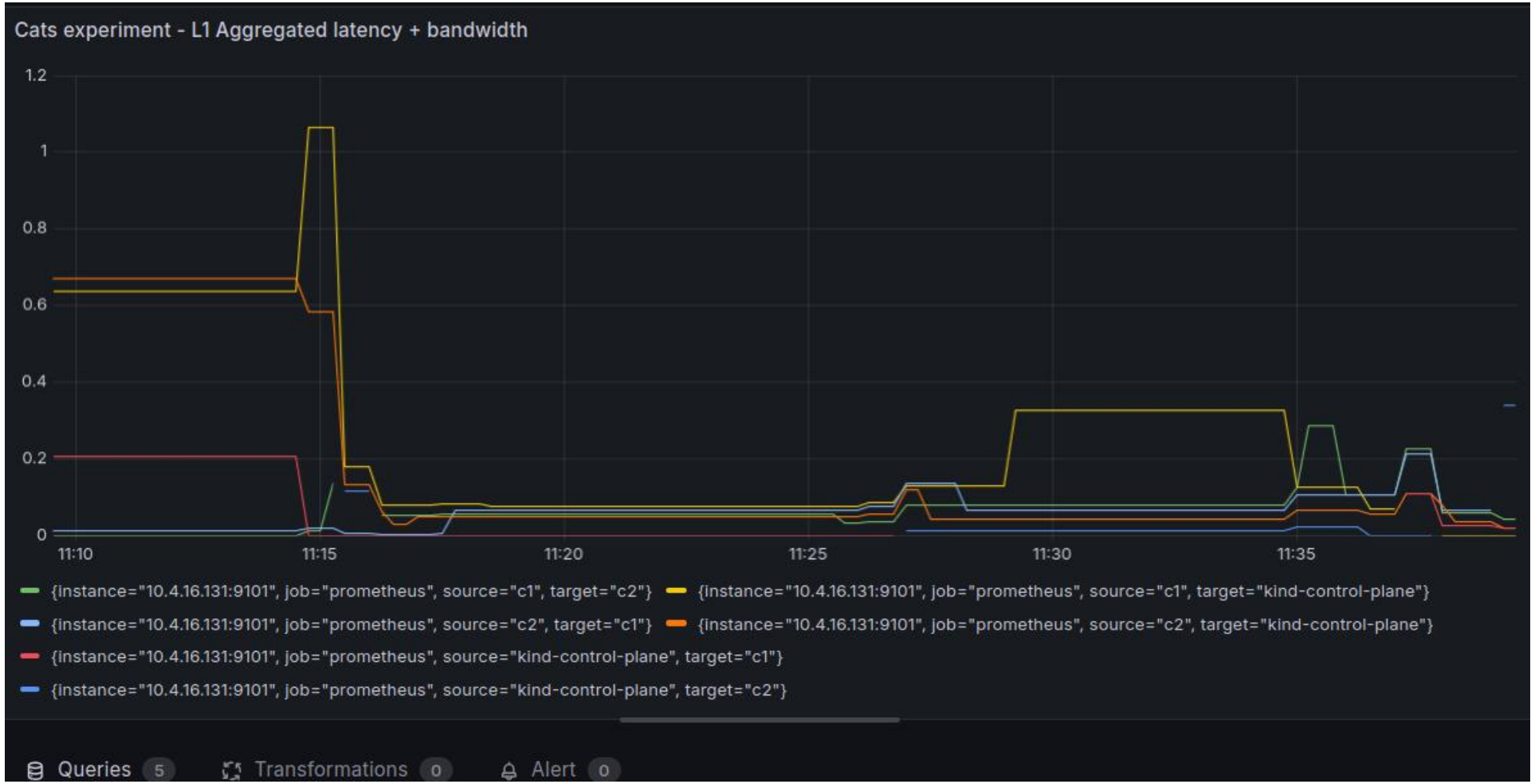
Figure 1: Logic of CATS Metrics in levels

Result L0


Cats experiment - L0 Latency
1000000
900000
800000
700000
600000
500000
400000
10:50
10:55
11:00
11:05
11:10
11:15
11:20
11:25
11:30
11:35
11:40
11:45
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="c1", target="c2"}
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="c1", target="kind-control-plane"}
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="c2", target="c1"}
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="c2", target="kind-control-plane"}
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="kind-control-plane", target="c1"}
{__name__="netma_latency_nanos", instance="10.4.16.131:9101", job="prometheus", source="kind-control-plane", target="c2"}
Queries 5
Transformations 0
Alert 0


Cats experiment - L0 CPU
0.26
0.24
0.22
0.2
0.18
0.16
0.14
0.12
0.1
0.08
11:05
11:10
11:15
11:20
11:25
11:30
{instance="10.4.16.131:9100", job="prometheus"}

Results L1


Cats experiment - L1 Aggregated latency + bandwidth
1.2
1
0.8
0.6
0.4
0.2
0
11:10
11:15
11:20
11:25
11:30
11:35
{instance="10.4.16.131:9101", job="prometheus", source="c1", target="c2"}
{instance="10.4.16.131:9101", job="prometheus", source="c1", target="kind-control-plane"}
{instance="10.4.16.131:9101", job="prometheus", source="c2", target="c1"}
{instance="10.4.16.131:9101", job="prometheus", source="c2", target="kind-control-plane"}
{instance="10.4.16.131:9101", job="prometheus", source="kind-control-plane", target="c1"}
{instance="10.4.16.131:9101", job="prometheus", source="kind-control-plane", target="c2"}
Queries 5
Transformations 0
Alert 0

Results L2


Cats experiment - L2 Final score number
14
12
10
8
6
4
2
0
12:05
12:10
12:15
12:20
12:25
12:30
c1
c2
kind-control-plane

Backup slide

- Should we include "service metrics", for example, in L1?
- Should we standardize the metric field and size?
- Define normalization function and aggregation functions.
- Meaning of the scores.
- Difference between L2 and L1 metrics when they are actually in use.
- Customized metrics.