AWS User Groups
Kuala Lumpur / Malaysia

aws
COMMUNITY DAY
MALAYSIA

Designing Observable Microservice Architectures for End-to-End ML Pipelines on AWS

Ananda Dwi Rahmawati

AWS Container Hero

aws
COMMUNITY DAY
MALAYSIA

AWS
User Groups
Kuala Lumpur / Malaysia

Ananda Dwi Rahmawati

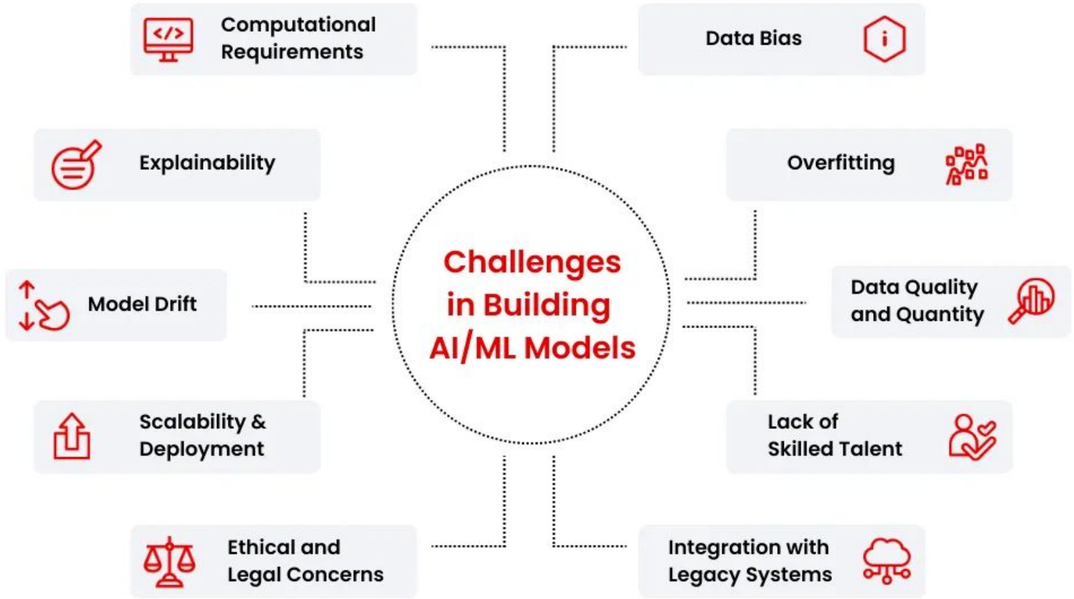
- Cloud & DevOps Engineer, Singapore
- AWS Container Hero
- Master of Computer Science - University of Texas at Austin
- https://linktr.ee/misskecupbung

Ananda Dwi Rahmawati

AWS Container Hero

Operating ML models, presents several challenges:

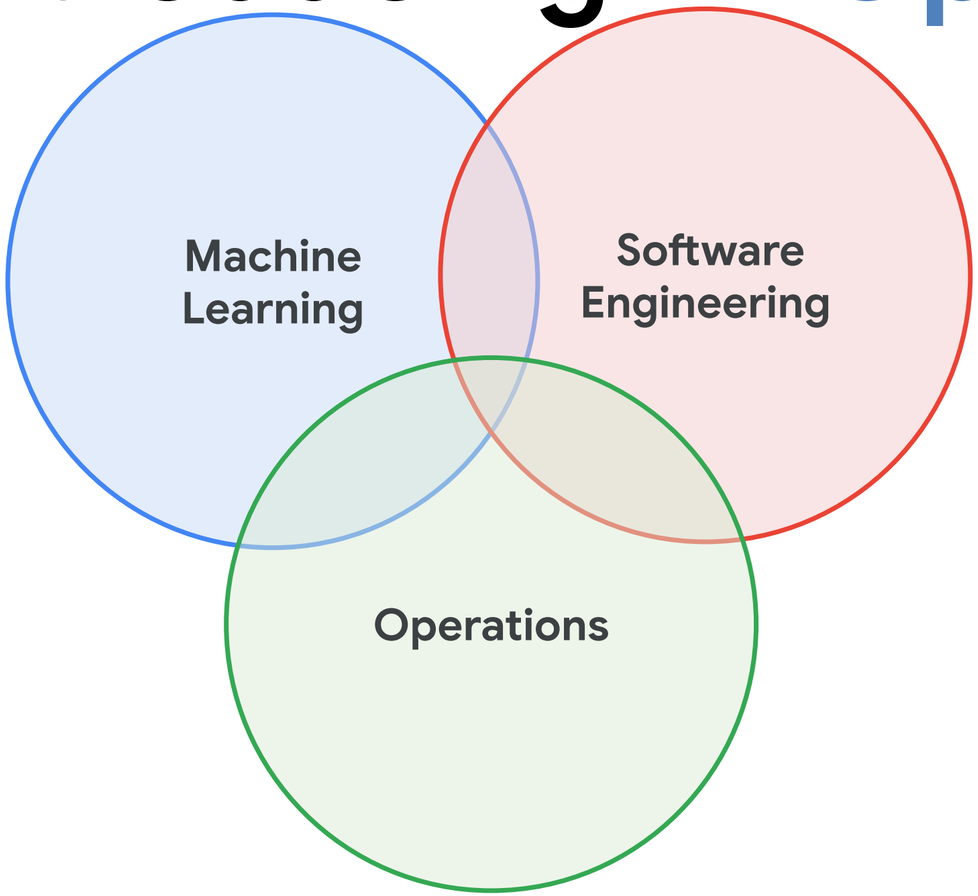
- **Model drift**: As real-world data changes, models become less accurate, requiring frequent retraining.
- **Resource management**: ML workloads have varying demands, making efficient allocation crucial.
- **Data quality**: Consistent, reliable input data is essential for model performance.
- **Compliance**: Meeting governance and regulatory requirements is challenging.
- **Versioning**: Tracking models, datasets, and experiments is difficult at scale.



Source: https://www.dqlabs.ai/blog/impact-of-data-quality-on-model-performance/

Introducing MLOps

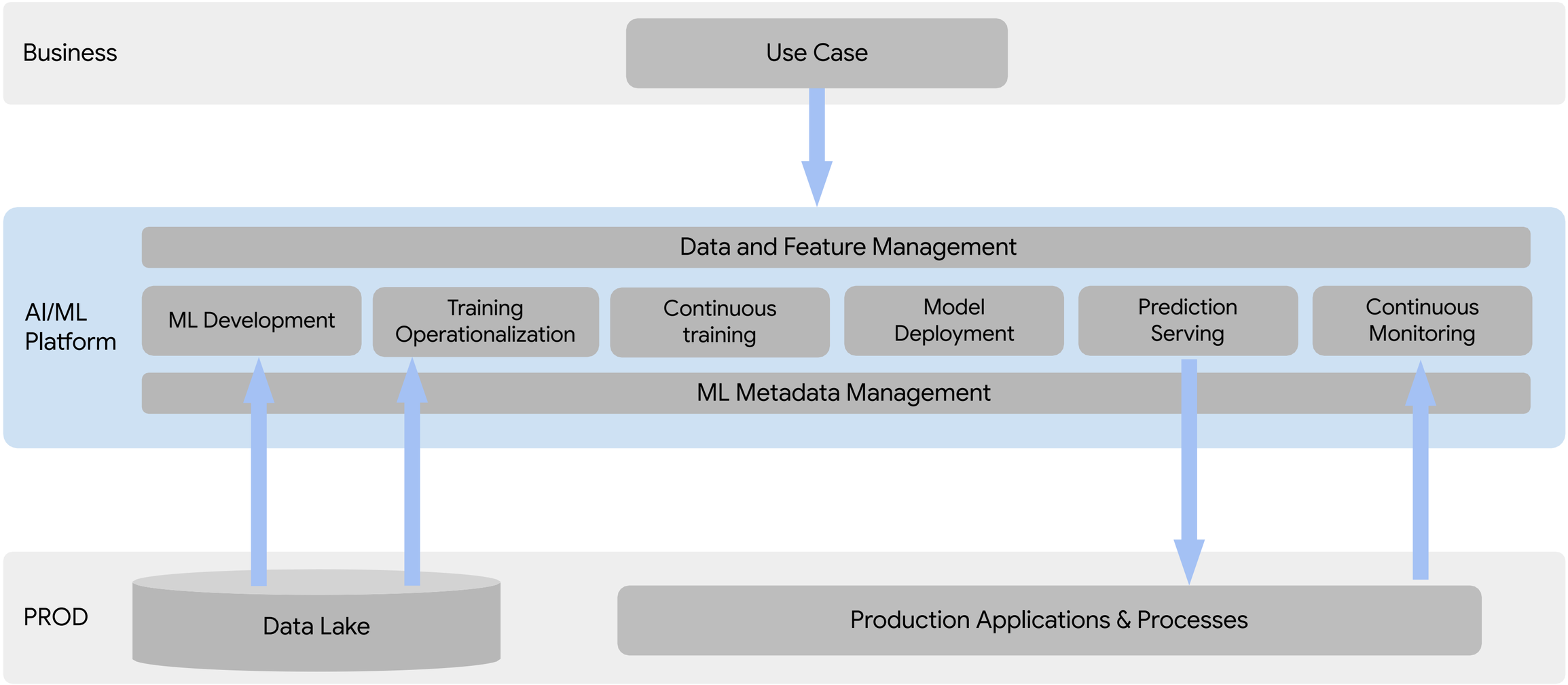
- Model development
- Model evaluation
- Parameter tuning



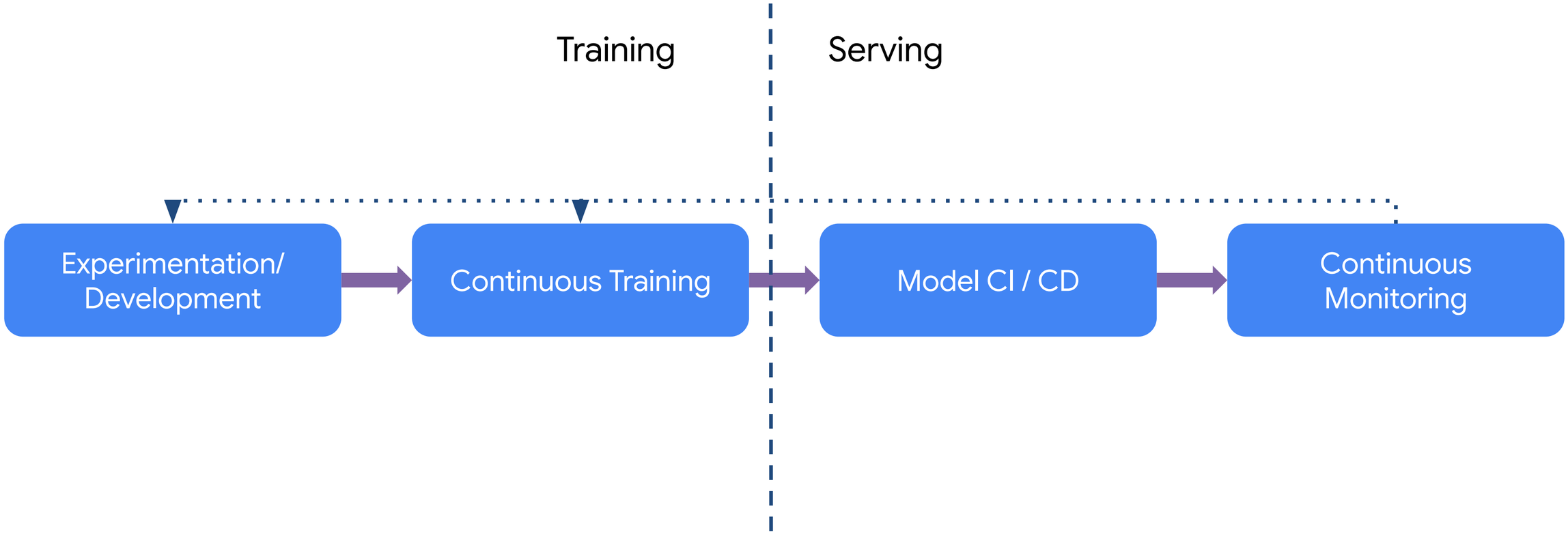
- Data engineering
- Pipeline development
- Integration of model into business application

- Model deployment
- Metadata management
- Logging and monitoring

MLOps: quick recap

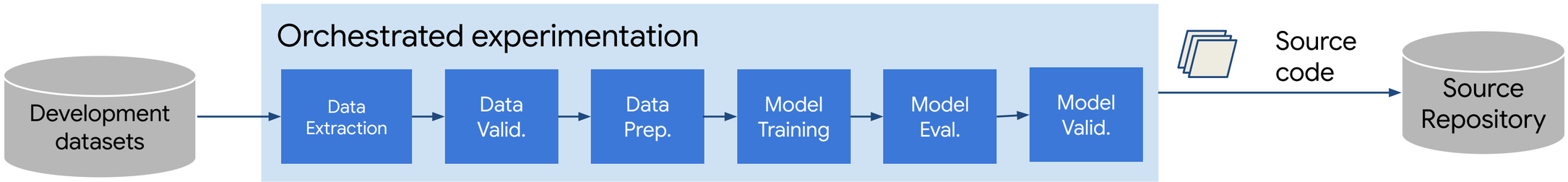


ML Solution Lifecycle



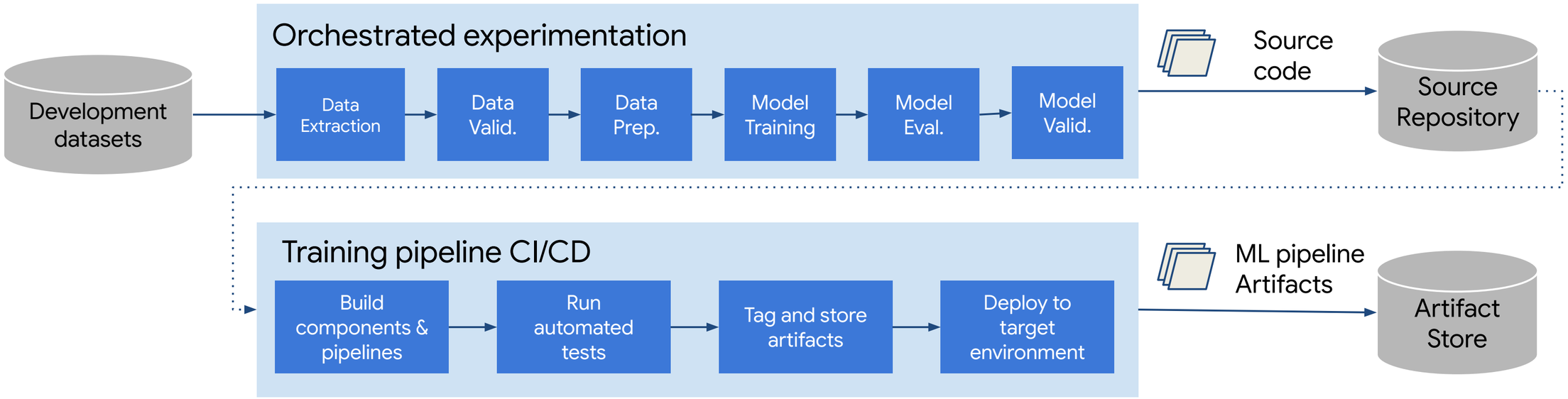
Reliable and repeatable training

Automated E2E Pipelines



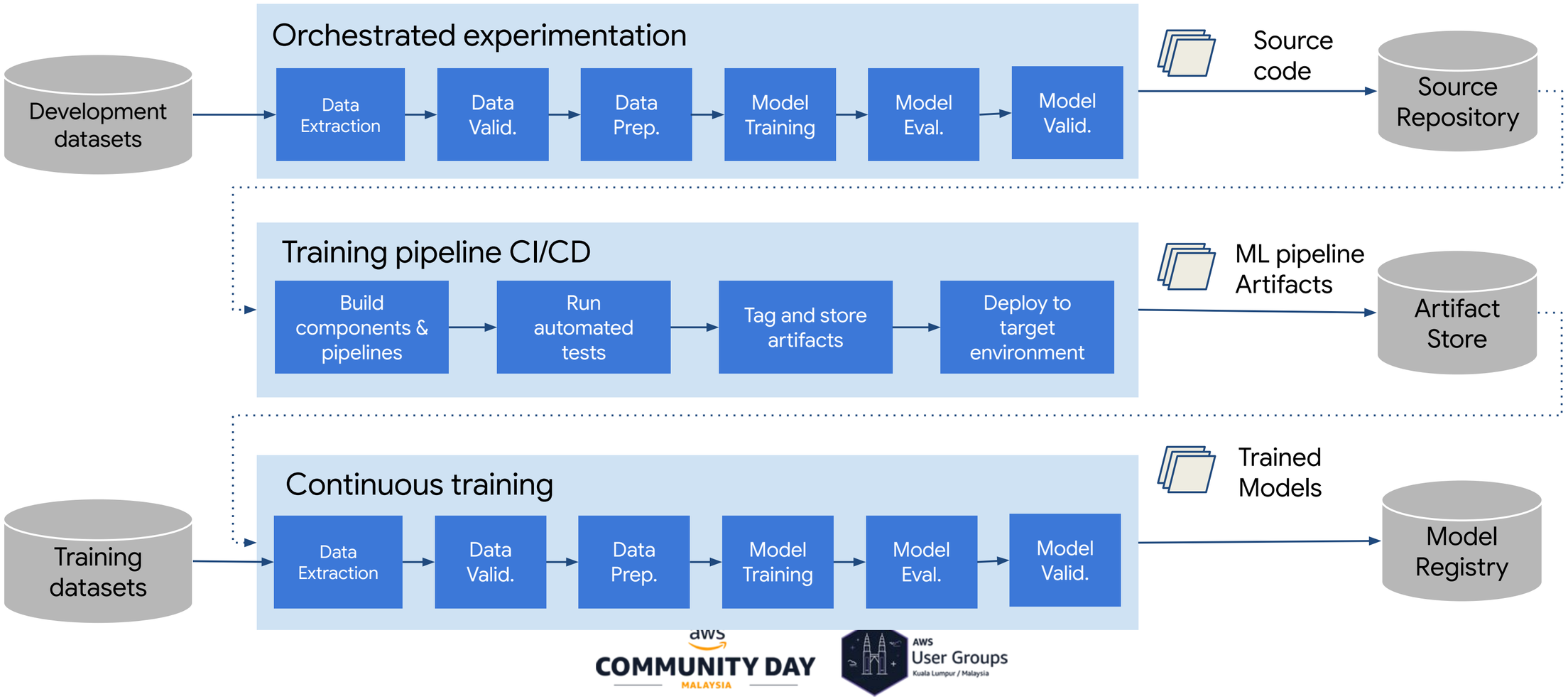
Reliable and repeatable training

Automated E2E Pipelines



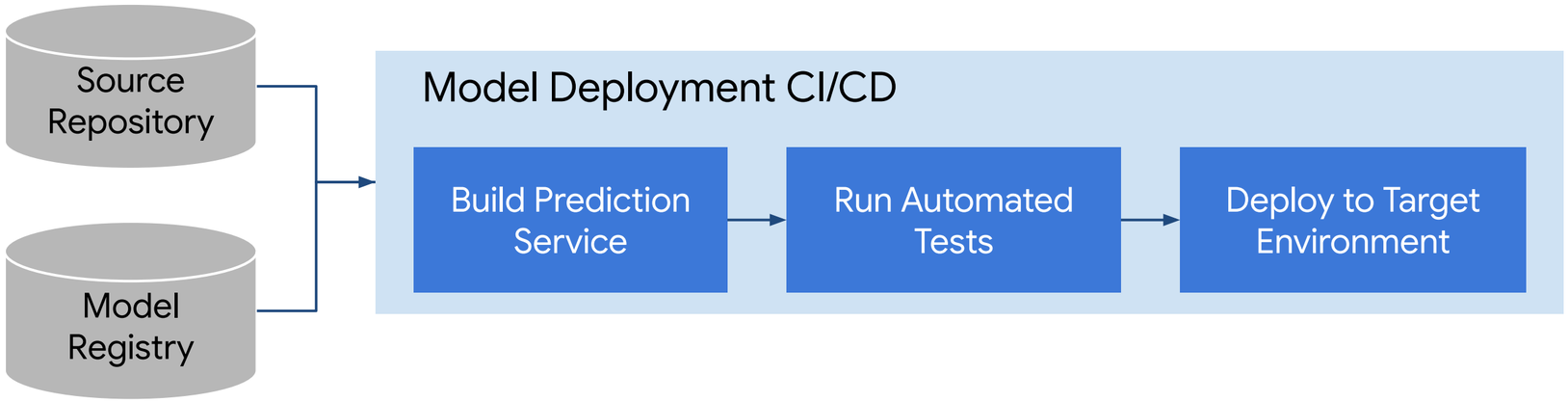
Reliable and repeatable training

Automated E2E Pipelines



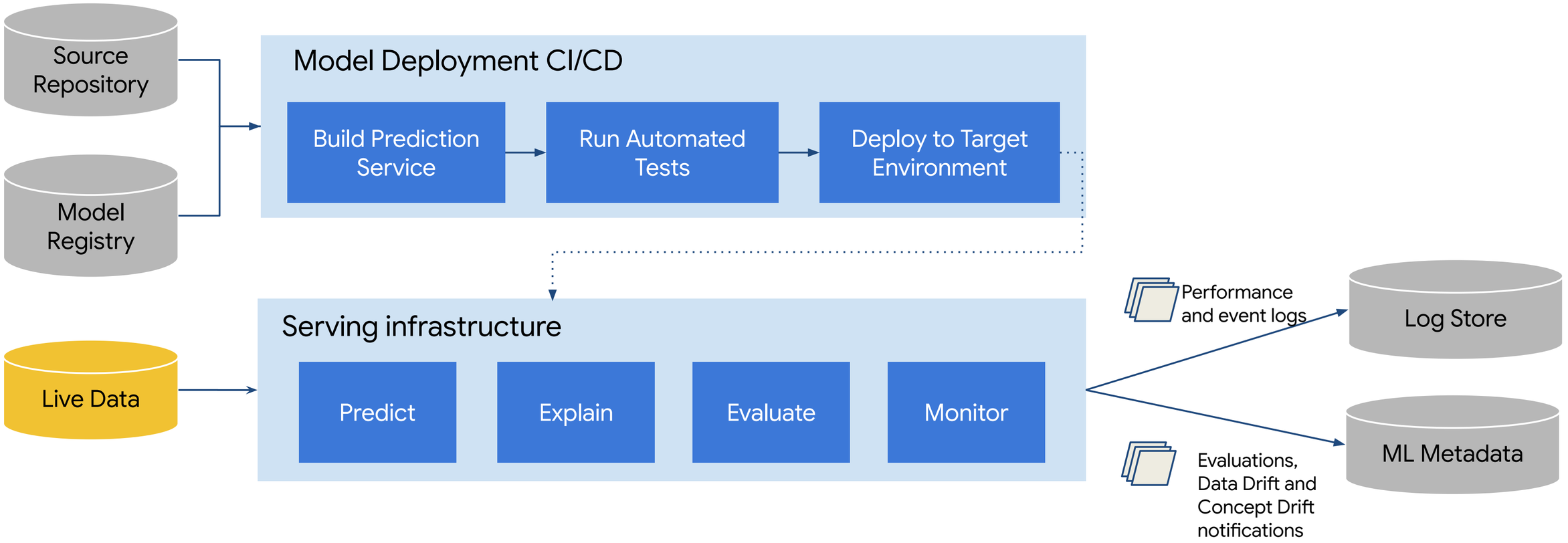
Reliable and monitored serving

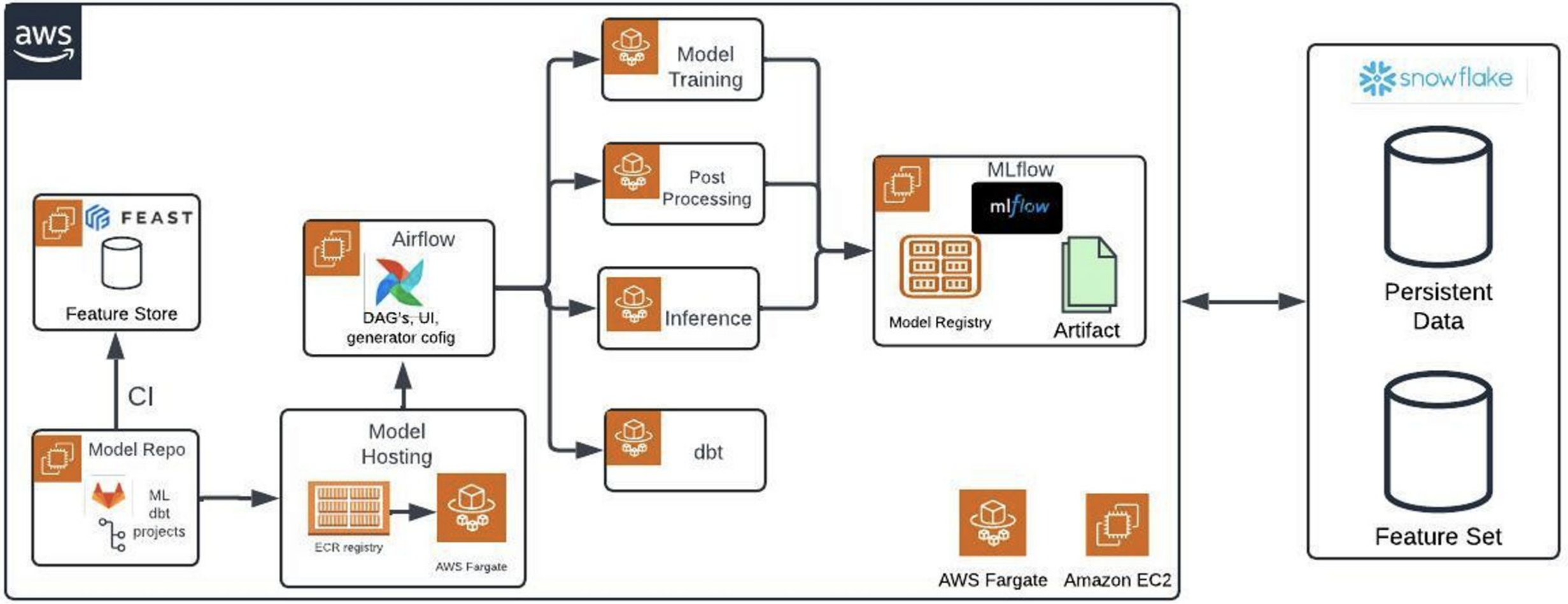
Automated E2E Pipelines



Reliable and monitored serving

Automated E2E Pipelines





Architecture overview:

- Airflow for workflow orchestration Feast for feature
- management dbt for accelerated data transformation
- MLflow for experiment tracking and model management

Source:
https://aws.amazon.com/blogs/machine-learning/building-an-efficient-mlops-platform-with-oss-tools-on-amazon-ecs-with-aws-fargate/

aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

“In control theory, **observability** is a measure of **how well** internal states of a system can be inferred from knowledge of its external outputs.”

“**Monitoring** tells you whether a system is working;

Observability lets you understand why isn't working.”

Unique ML Characteristics

Resource Patterns:

- Sustained high GPU usage during training vs consistent CPU usage in traditional apps
- Specialized GPU node scheduling vs typical short-lived batch jobs
- Variable computational demands requiring dynamic resource allocation

Monitoring Focus:

- **Model-specific metrics**: Accuracy, F1 scores (irrelevant for standard applications)
- **Data drift monitoring:** Track shifts in user preferences and data patterns
- **Continuous feedback loops:** Analyze interactions for targeted improvements
- **Granular observations**: Sometimes per-prediction monitoring vs standard application metrics

Source: https://www.dqlabs.ai/blog/impact-of-data-quality-on-model-performance/

Observability

Observability = **gaining insights into ML model behavior & infrastructure**.

Enables Teams to:

- Quickly identify and diagnose issues
- Optimize resource usage
- Ensure compliance
- Monitor model performance and detect drift
- Track data quality and integrity

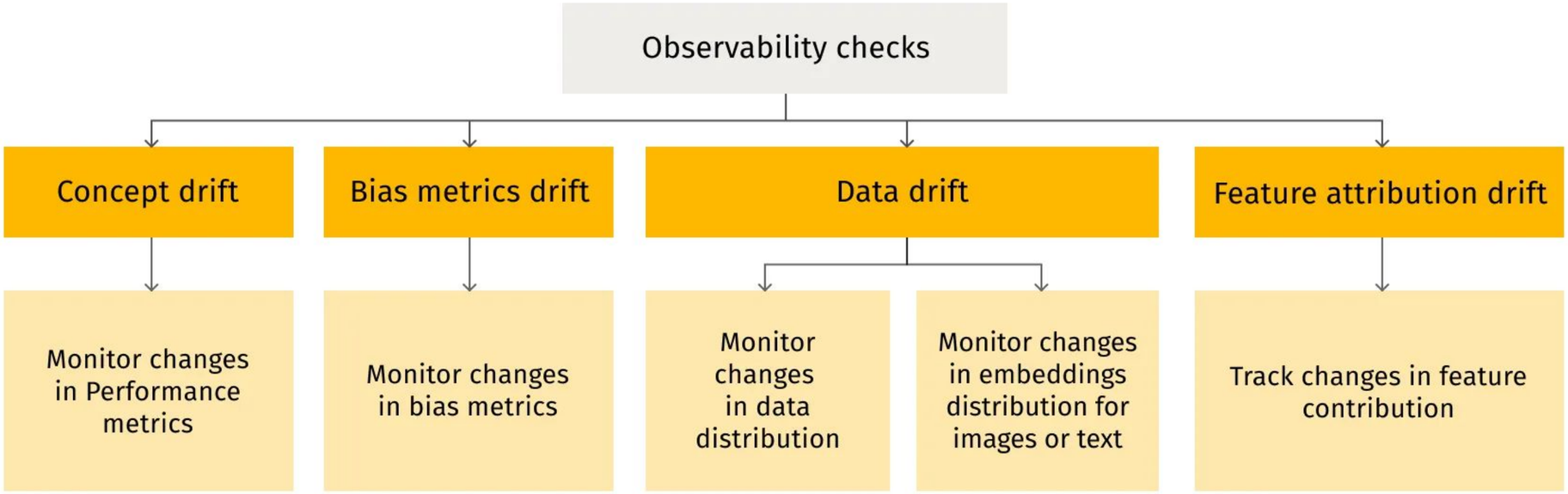
Feedback Loop:

- Continuous monitoring and retraining using real-world data
- Helps models adapt to user behavior, new data patterns, and emerging trends
- Drives better decision-making, user experience, and business value

Source: https://www.cloudlaunchpad.app/blog/observability-in-aws-eks

The Need for ML Observability in MLOps

Pillar	What It Covers	AWS Services / Methods
Data Quality	Detect schema mismatches, cardinality shifts, out-of-range values; track distribution drift in features. (Grid Dynamics)	Use AWS SageMaker + Model Monitor; batch & streaming ingestion; baseline jobs. (Grid Dynamics)
Fairness / Bias	Pre- and post-training bias detection; monitoring predictions' distribution across sensitive groups. (Grid Dynamics)	AWS Clarify for bias metrics; set facets; integrate into model evaluation pipelines. (Grid Dynamics)
Explainability	Understanding which features drive predictions (global & local); detecting unjustified dependencies. (Grid Dynamics)	Methods like SHAP, LIME; use feature attribution drift jobs; visualize top features & heatmaps. (Grid Dynamics)
Model Performance / Drift	Monitoring model accuracy, recall, F1, etc.; detecting concept drift; comparing predictions vs ground truth. (Grid Dynamics)	AWS Model Monitor, performance / drift jobs; optionally use open-source libs like *nannyml*. (Grid Dynamics)



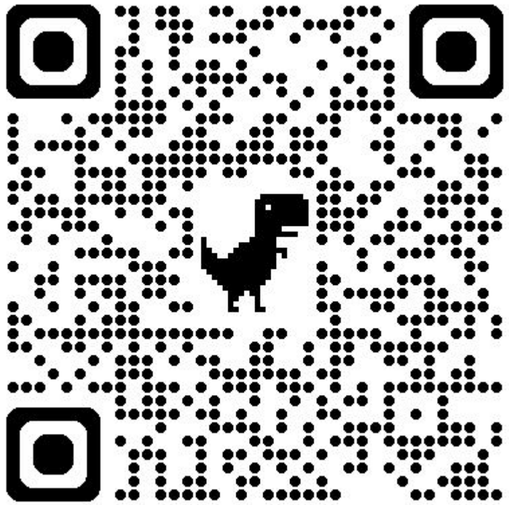
- Model accuracy
- Model score drifts
- Monitor drift between train baseline and production scoring

- Model fairness

- Monitor feature drifts across tabular (structured) and unstructured (images, video) data

- Explainability

Let’s Demo



Container Insights | Service: EKS | Add to dashboard | View in maps | View performance dashboards

Clusters state summary (3)

As of May 13, 2024, 11:37 AM (UTC+01:00)

Clusters with alarms

0 In alarm | 0 OK

Clusters without alarms

2 High utilization | 1 Low utilization

Performance and status summary

Last 1 min

Clusters CPU (avg)
Utilization **13%**
Reserved **14%**

Clusters Memory (avg)
Utilization **30%**
Reserved **2%**

Clusters GPU (avg)
Utilization **0%**
Memory **0%**

Clusters NeuronCore (avg)
Utilization **72%**

Pods (sum)
Desired **66**
Ready **65**

Nodes (sum)
Unavailable **0**
Available **6**

Control plane summary

Last 3 hours

122
Max API server requests

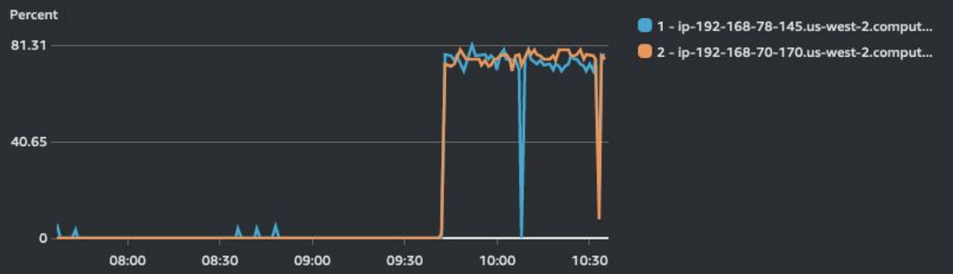
28.6
Average API server requests latency

93
Total number of stored objects

0
Average admission controller latency

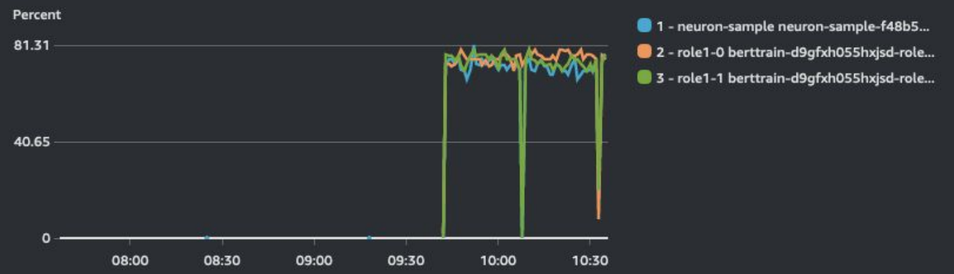
Top 10 Nodes per metric NeuronCore Utilization

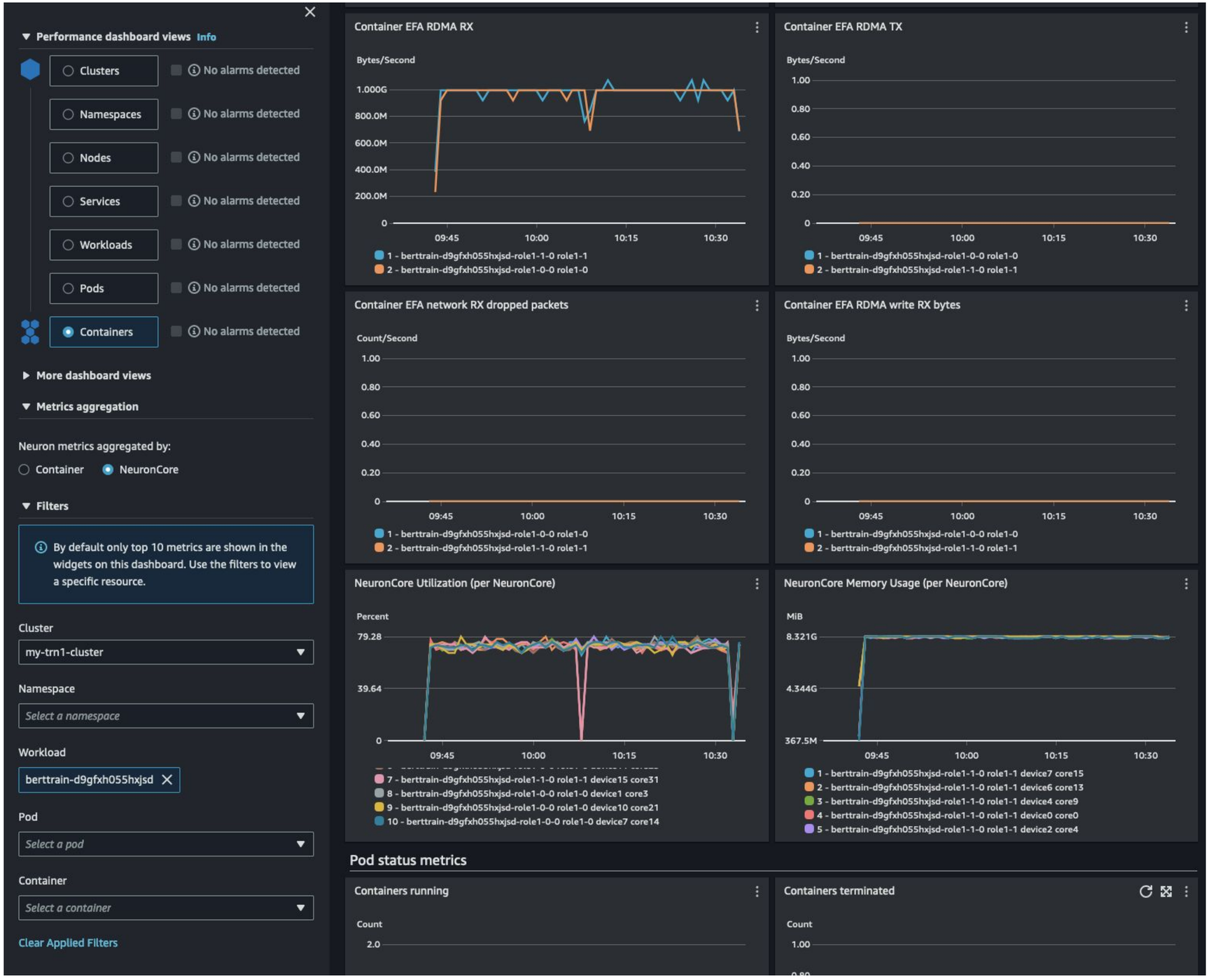
Last 3 hours



Top 10 Containers per metric NeuronCore Utilization

Last 3 hours

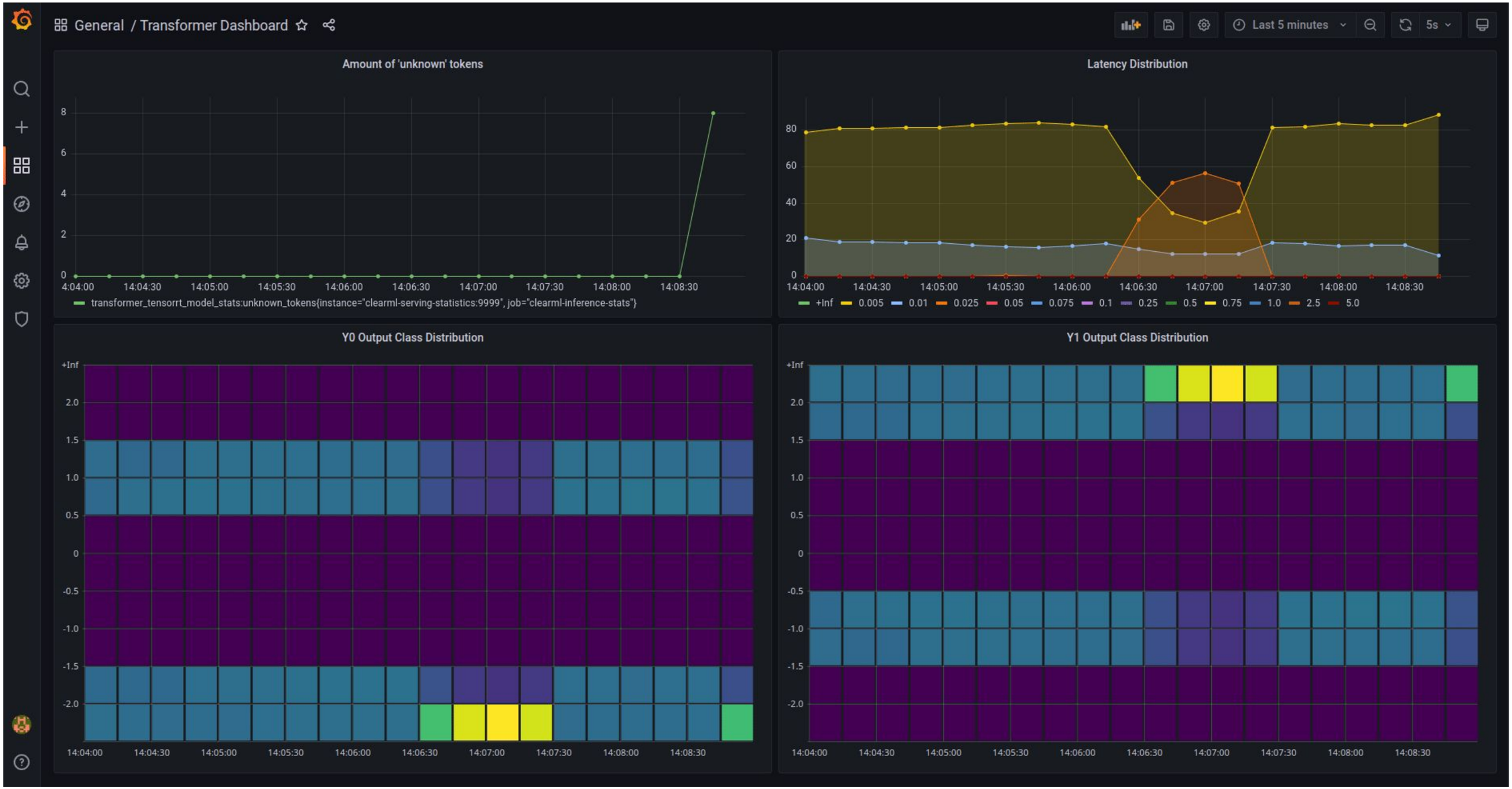



Performance dashboard views Info
Clusters
No alarms detected
Namespaces
No alarms detected
Nodes
No alarms detected
Services
No alarms detected
Workloads
No alarms detected
Pods
No alarms detected
Containers
No alarms detected
More dashboard views
Metrics aggregation
Neuron metrics aggregated by:
Container
NeuronCore
Filters
By default only top 10 metrics are shown in the widgets on this dashboard. Use the filters to view a specific resource.
Cluster
my-trn1-cluster
Namespace
Select a namespace
Workload
berttrain-d9gfxh055hxjsd
Pod
Select a pod
Container
Select a container
Clear Applied Filters
Container EFA RDMA RX
Bytes/Second
1.000G
800.0M
600.0M
400.0M
200.0M
0
09:45
10:00
10:15
10:30
1 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1
2 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0
Container EFA RDMA TX
Bytes/Second
1.00
0.80
0.60
0.40
0.20
0
09:45
10:00
10:15
10:30
1 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0
2 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1
Container EFA network RX dropped packets
Count/Second
1.00
0.80
0.60
0.40
0.20
0
09:45
10:00
10:15
10:30
1 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0
2 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1
Container EFA RDMA write RX bytes
Bytes/Second
1.00
0.80
0.60
0.40
0.20
0
09:45
10:00
10:15
10:30
1 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0
2 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1
NeuronCore Utilization (per NeuronCore)
Percent
79.28
39.64
0
09:45
10:00
10:15
10:30
7 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device15 core31
8 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0 device1 core3
9 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0 device10 core21
10 - berttrain-d9gfxh055hxjsd-role1-0-0 role1-0 device7 core14
NeuronCore Memory Usage (per NeuronCore)
MiB
8.321G
4.344G
367.5M
09:45
10:00
10:15
10:30
1 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device7 core15
2 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device6 core13
3 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device4 core9
4 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device0 core0
5 - berttrain-d9gfxh055hxjsd-role1-1-0 role1-1 device2 core4
Pod status metrics
Containers running
Count
2.0
Containers terminated
Count
1.00

aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

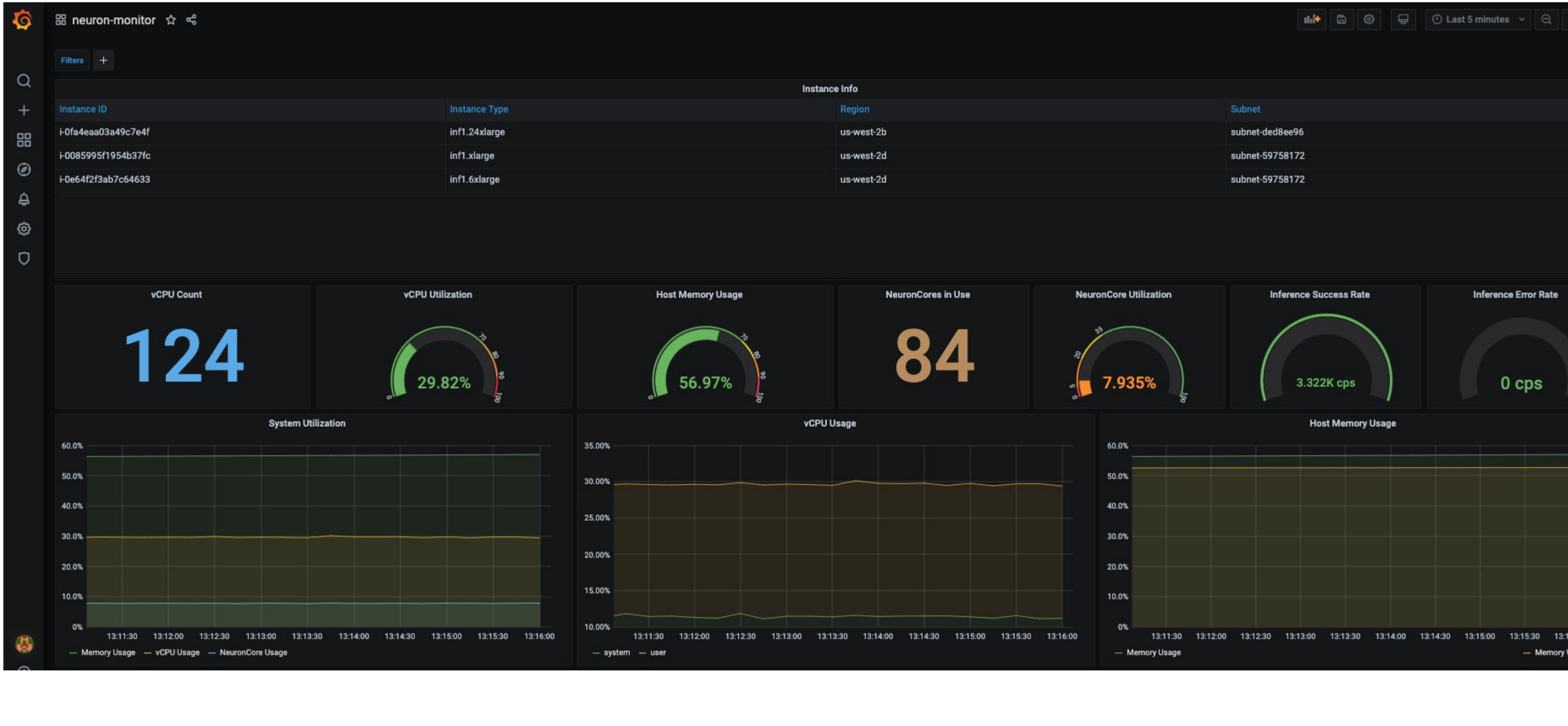
Component



aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

Component

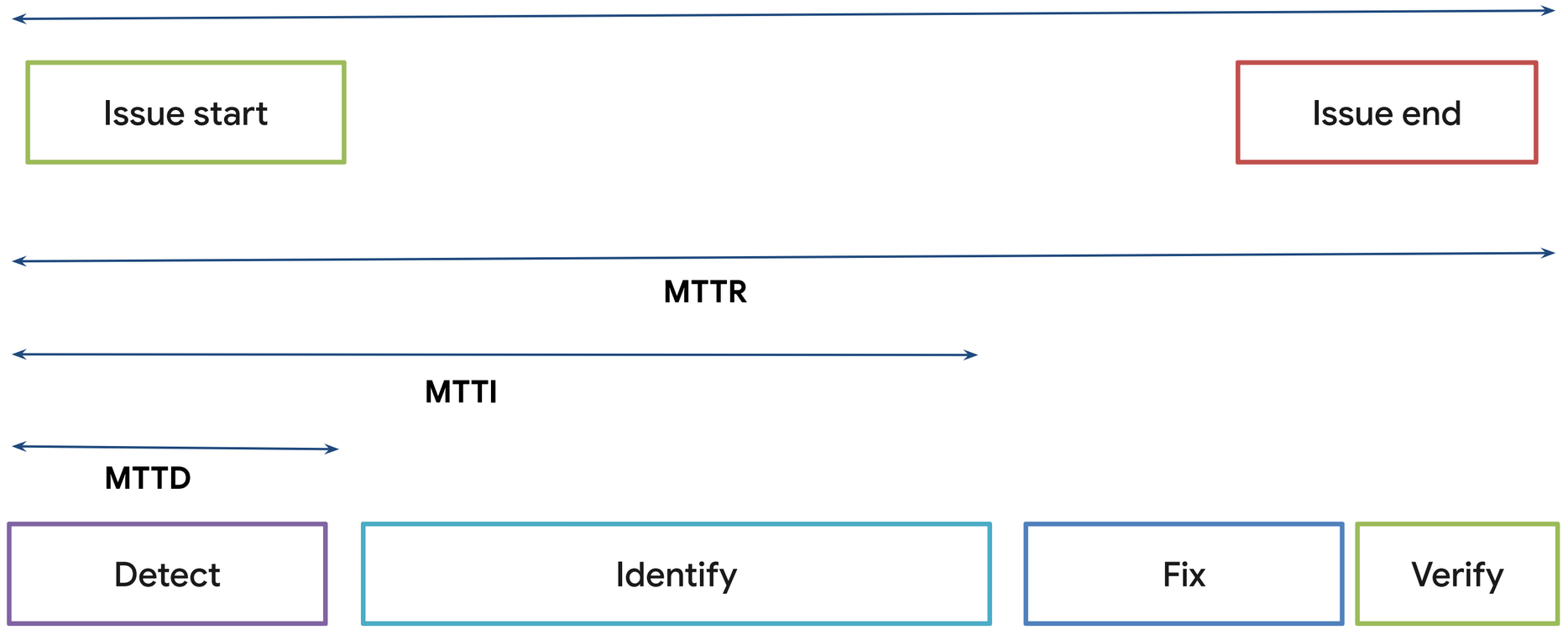


aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

Component

Issue Timeline

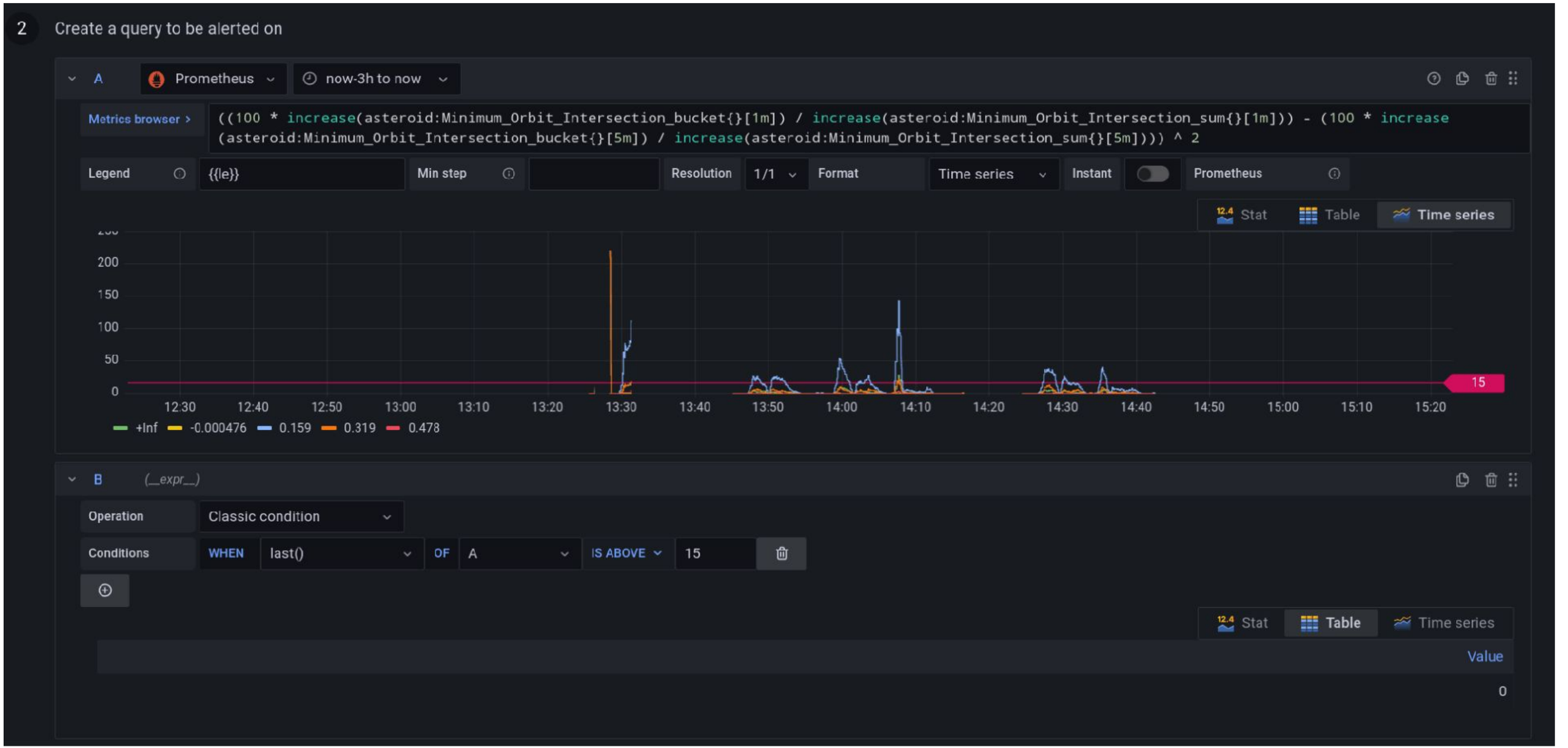


How to Measuring

- Changes made to monitoring configuration
- "Out of hours" alerts
- Team alerting balance
- False positives & negatives

- Alert creation
- Alert acknowledgement
- Alert silencing and silence duration
- Unactionable alerts
- Usability: alerts, runbooks, dashboards
- MTTD, MTTR, impact

2 Create a query to be alerted on

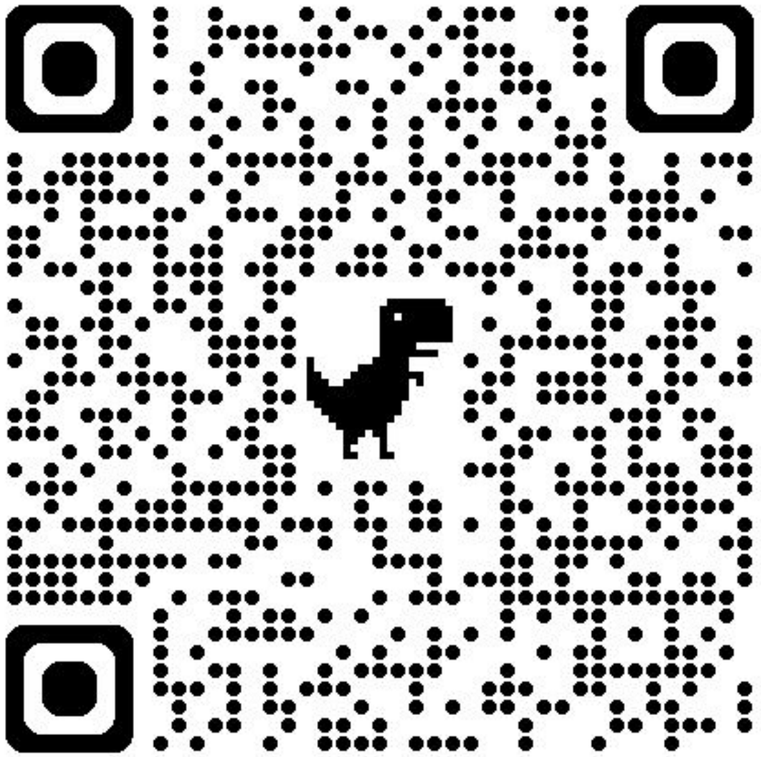


aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

Component

Feedback



aws COMMUNITY DAY MALAYSIA

AWS User Groups Kuala Lumpur / Malaysia

Thank you

aws
COMMUNITY DAY
MALAYSIA

AWS
User Groups
Kuala Lumpur / Malaysia