Unlocking Network Data: A New Take on Observability
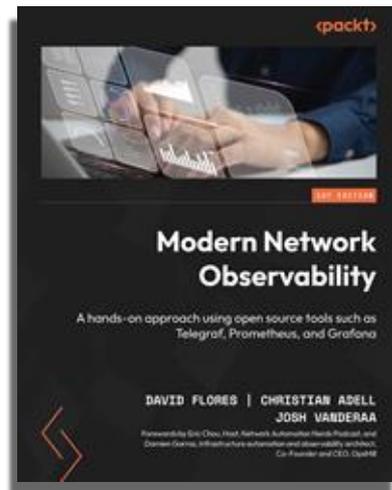
INEX (March 12, 2026)

About me

- Network Automation and Observability at CoreWeave
- Experience in Service Provider, Financial, Hyperscalers, and Integrators.
- Focused on building open source automation and network observability platforms
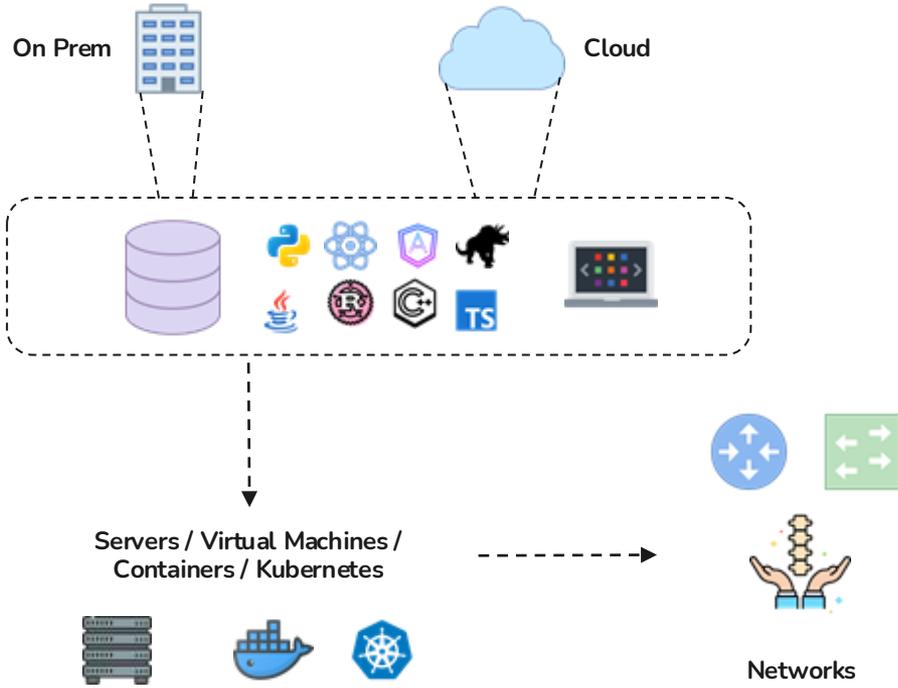- Co-Author of "Modern Network Observability"

@netpanda.bsky.social - Bluesky

@davidban77 - Github / X

David Flores - LinkedIn

Network Monitoring Today

On Prem

Cloud

Servers / Virtual Machines /
Containers / Kubernetes

Networks

- MRTG, PRTG, Cacti – Familiar but limited.
- **Networks: The backbone of modern infrastructure.**
- Complexity is growing: cloud, on-premises, hybrid, microservices.
- Challenge? some metrics and disparate logging isn't enough for deep insights.
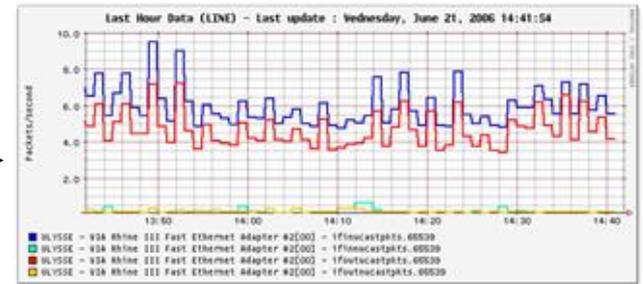
What Is Observability?

- **Observability vs. Monitoring**: "What happened" vs. "Why it happened."
- Extracting actionable insights from **multiple system signals**.
- Distributed systems and modular architectures require new strategies.

Metrics

```
ifOutOctets=3814i, ifInOctets=469i
```
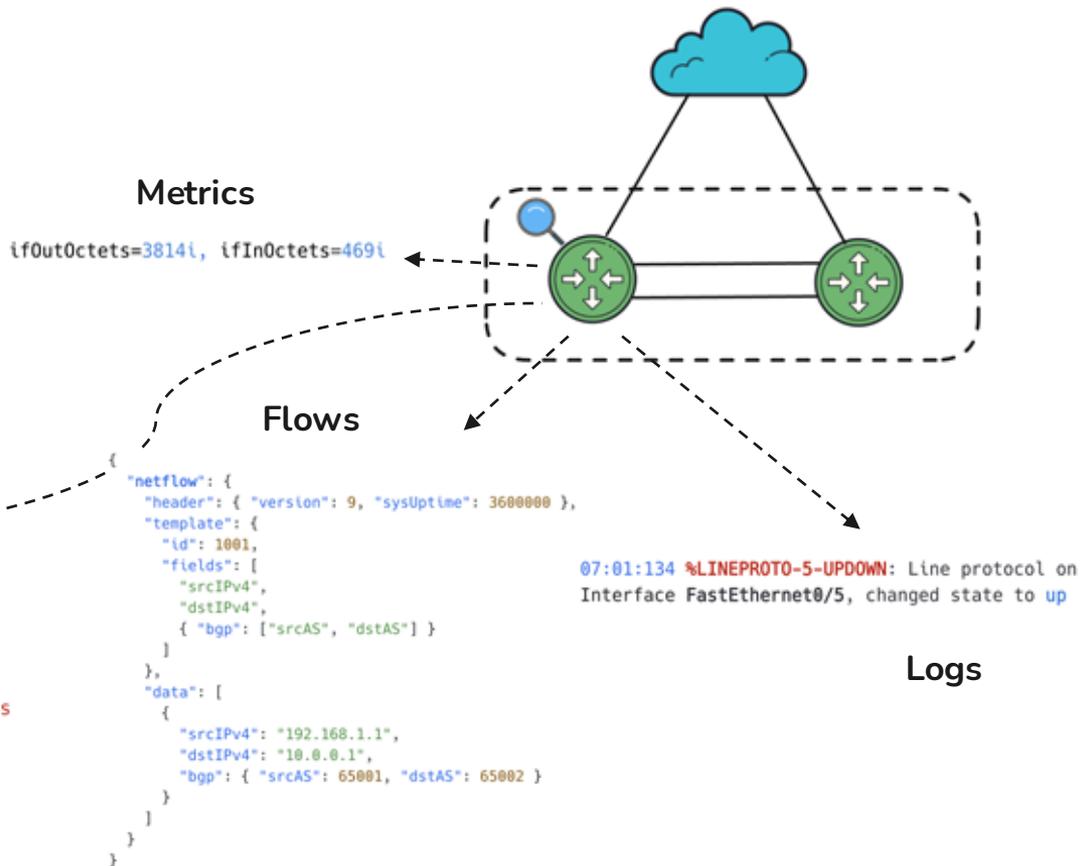
Synthetic Monitoring

```
Router# ping 192.0.2.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/8 ms
```

Flows

```
{
  "netflow": {
    "header": { "version": 9, "sysUptime": 3600000 },
    "template": {
      "id": 1001,
      "fields": [
        "srcIPv4",
        "dstIPv4",
        { "bgp": ["srcAS", "dstAS"] }
      ]
    },
    "data": [
      {
        "srcIPv4": "192.168.1.1",
        "dstIPv4": "10.0.0.1",
        "bgp": { "srcAS": 65001, "dstAS": 65002 }
      }
    ]
  }
}
```

Logs

```
07:01:134 %LINEPROTO-5-UPDOWN: Line protocol on
Interface FastEthernet0/5, changed state to up
```

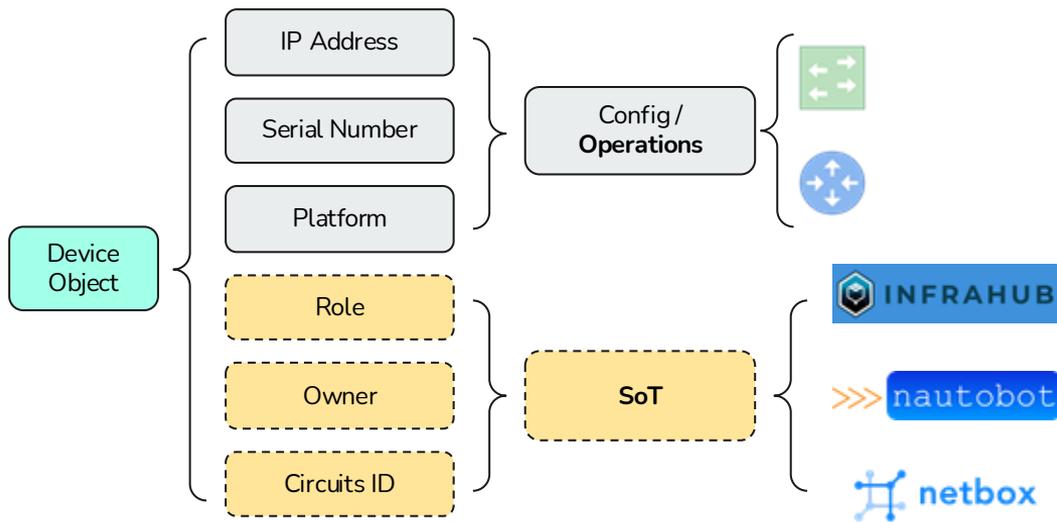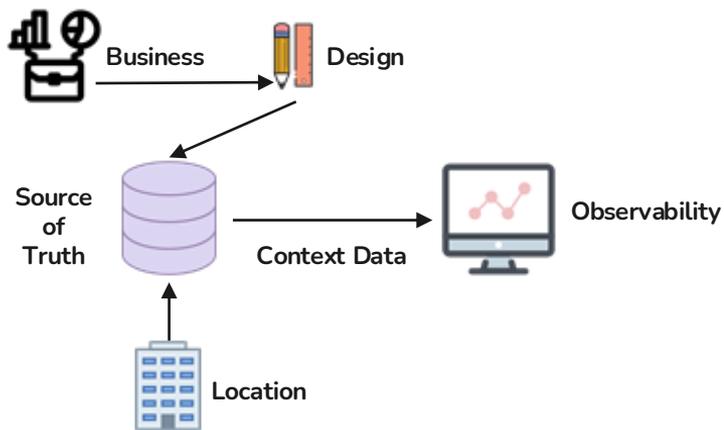# The Problem with Current Tools

- **Vendor Lock-In**: Limited scalability and flexibility.

- **Static Dashboards**: Outdated visualizations that don't adapt.

- **Inflexible Data Collection** and Processing: Data comes in many many many forms…
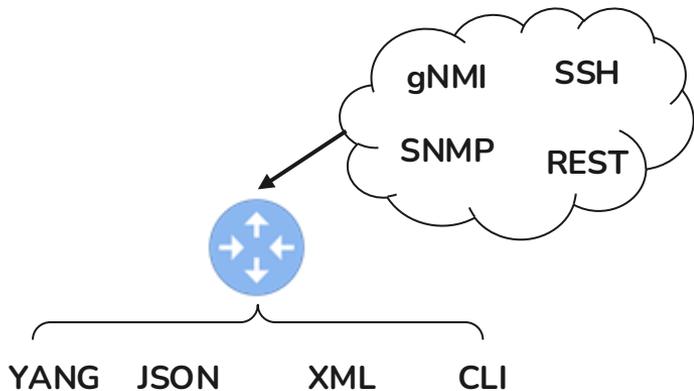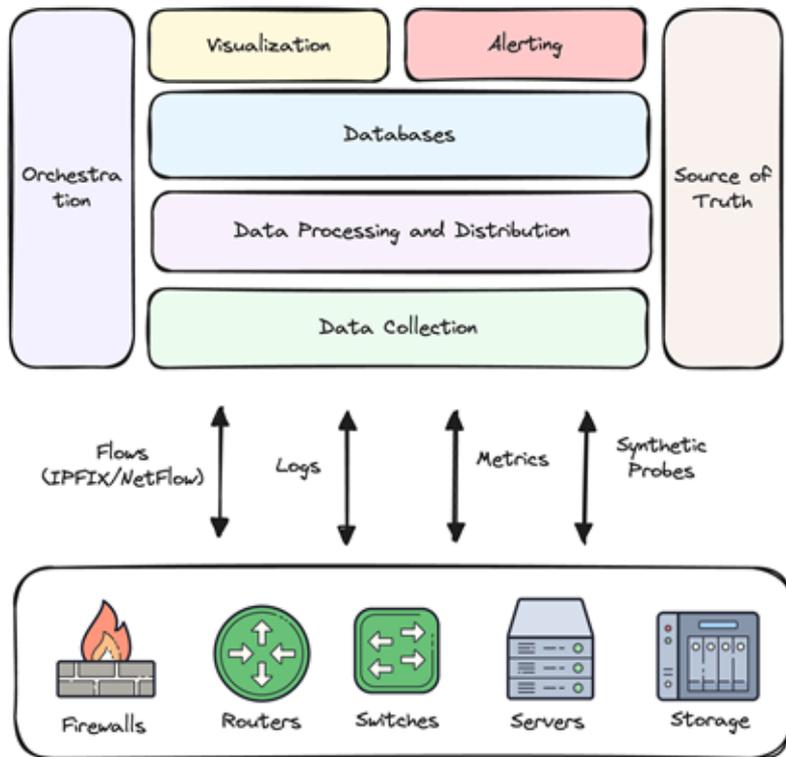
The Problem with Current Tools

- **Missing Context**: Lack of contextual (**enrichment**) data for smarter alerts.

The Observability Architecture

Modular architecture and specialized components helps addressing the challenges.
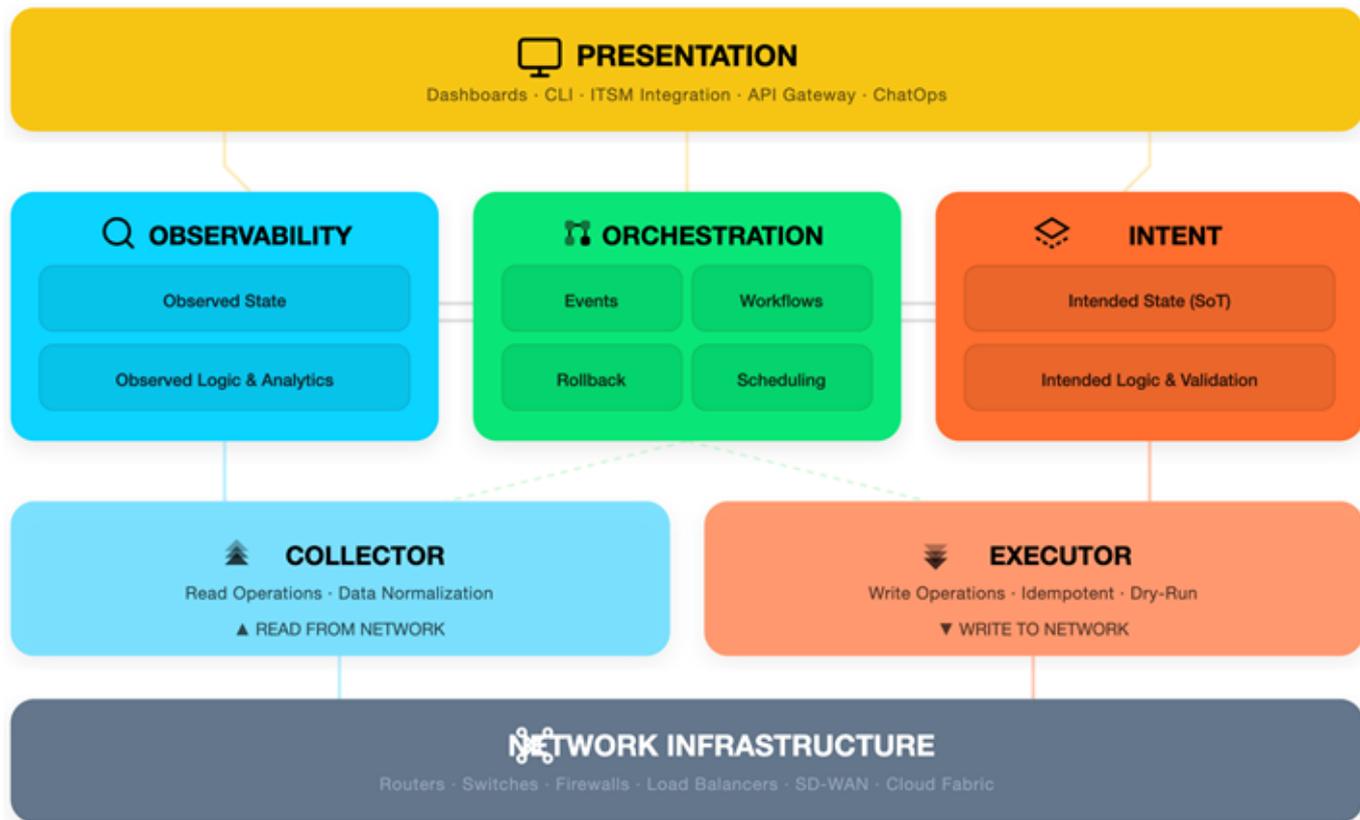
- **Collection**: Gather data from SNMP, streaming telemetry, logs.
- **Processing & Distribution**: Clean and unify diverse data.
- **Storage**: Leverage specialized databases (time-series, log storage).
- **Visualization & Alerts**: Turn data into actionable insights with tools like Grafana.

Where does Observability sit now?

https://reference.networkautomation.forum/Framework/Framework/

PRESENTATION
Dashboards · CLI · ITSM Integration · API Gateway · ChatOps

OBSERVABILITY
Observed State
Observed Logic & Analytics

ORCHESTRATION
Events
Workflows
Rollback
Scheduling

INTENT
Intended State (SoT)
Intended Logic & Validation

COLLECTOR
Read Operations · Data Normalization
▲ READ FROM NETWORK

EXECUTOR
Write Operations · Idempotent · Dry-Run
▼ WRITE TO NETWORK

NETWORK INFRASTRUCTURE
Routers · Switches · Firewalls · Load Balancers · SD-WAN · Cloud Fabric

Tooling Landscape:

- **Collection** and **Execution** layers HAVE TO BE FLEXIBLE in tooling choices
- **Databases for Observability** dictate
  - Querying languages APIs
  - Data ingestion methods
- **Intent stores** need to be able to **model your infrastructure** to provide the right context
- Scale changes architecture and tooling choices
- Do not fall in love with the tools… You never know
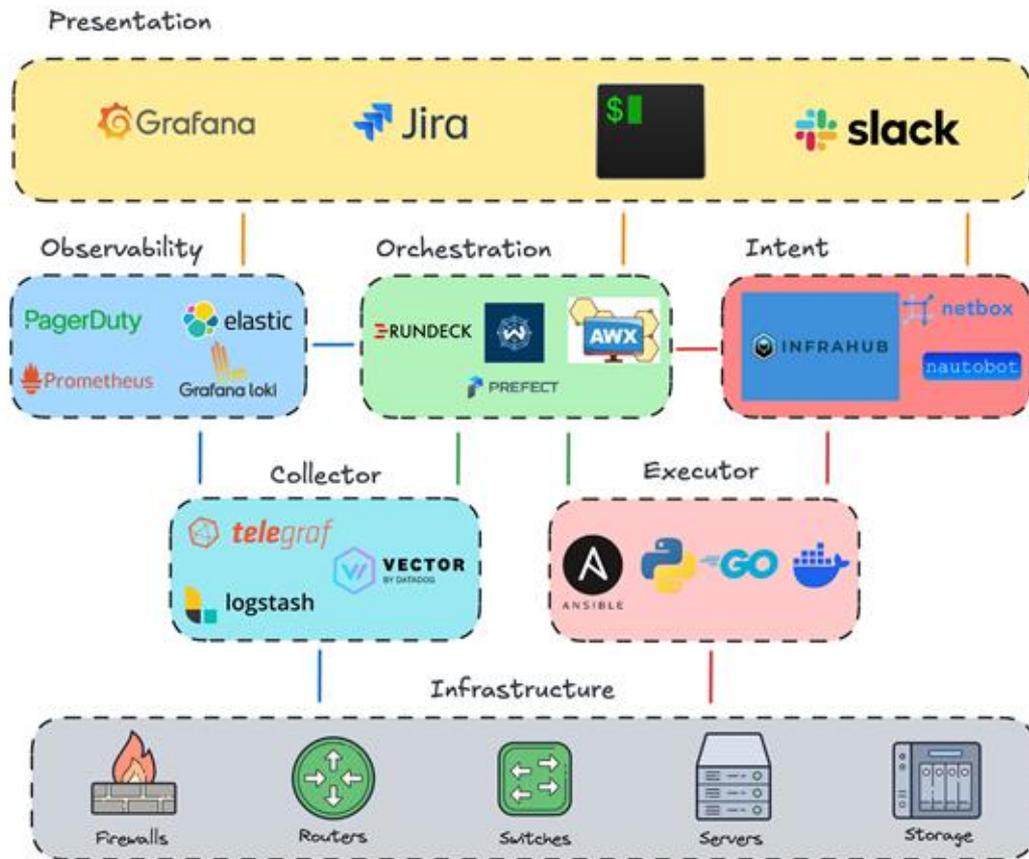
Observability and Framework Working Together

Example Stack:

- Tooling used throughout the book and available here:

https://github.com/network-observability/network-observability-lab

- Key takeaways:
  - Data collection - SNMP, gNMI and Python tooling
  - PromQL and LogQL for accessing data
  - SoT used for data and visualization enrichment
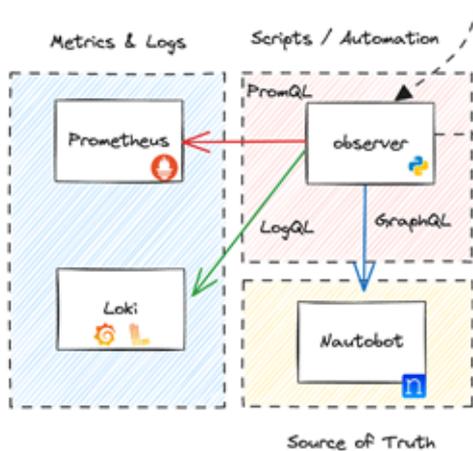 - Programmatic access to observed data
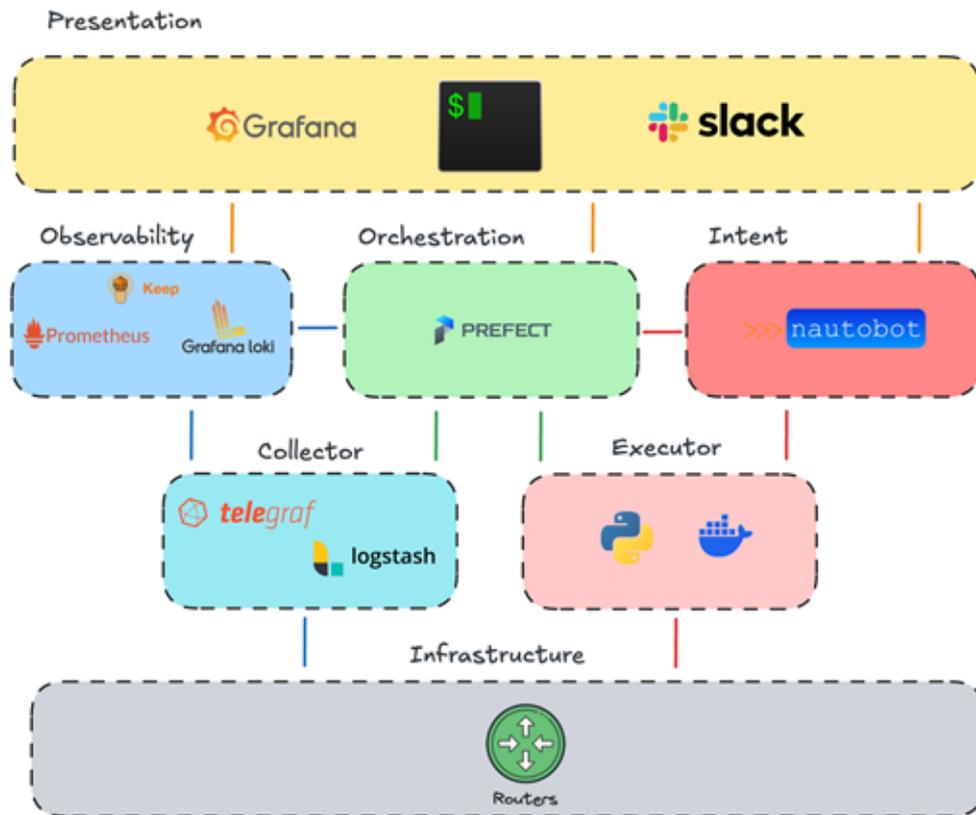
Transforming Operations with Observability

- **Correlation**: Ability to correlate multiple data sources for the same entity.
- **Logs to Metrics**: Set up recording rules that convert logs into metrics.
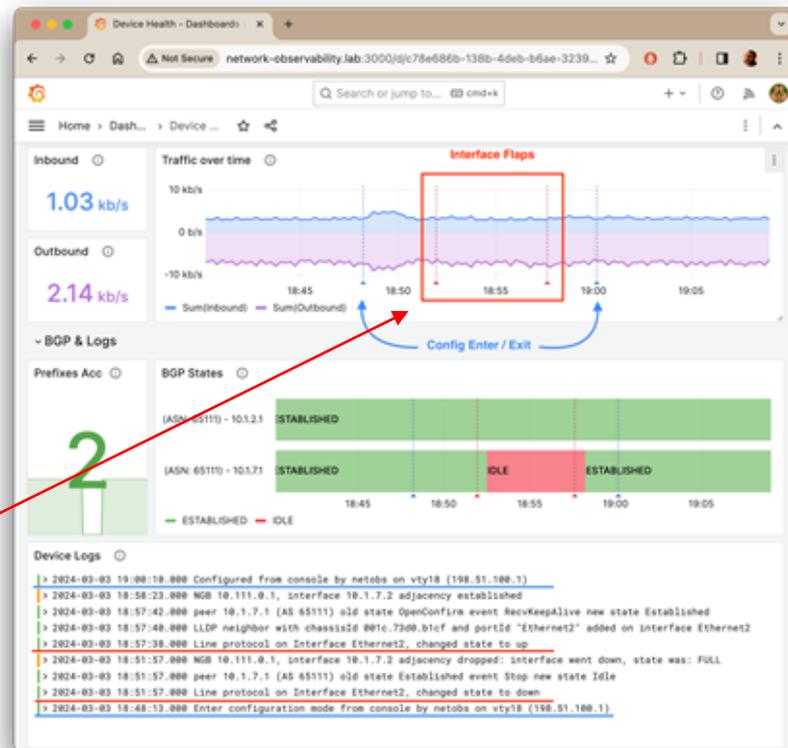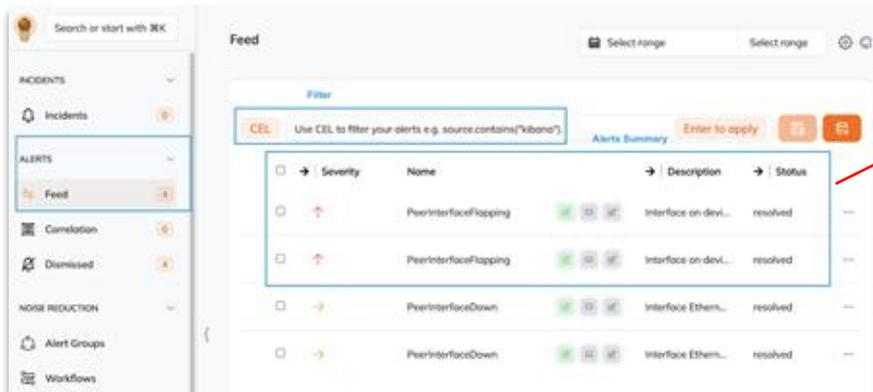
Transforming Operations with Observability

- **Data Access**: API-based access to metrics, logs and alerts.
- **Scripting**: Create scripts or CLI/Web tools interacting with the collected data.
- **Event Driven Automation**: Perform automated workflows to tackle issues like ticketing for interface flapping events.

Metrics & Logs

Scripts / Automation

SSH

PromQL

Prometheus

observer

LogQL

GraphQL

Loki

Nautobot

Source of Truth

CLI App Observer

AI-Enhanced Observability

- **AI-Driven Insights**: Actively query and aggregate network data.
- **Structured Data Empowers AI**: Data must be queryable and unified.

```
{datacenter="DC1", role="internet-border"} |= "BGP session down"
```


1 - An event is generated

2 - API Endpoint /v1/api/rca-webhook

3 - Retrieve more data (logs)

Logs DB

4 - Craft a powerful prompt

> You are an expert...

5 - Ask an LLM bot

LLMs

6 - Output
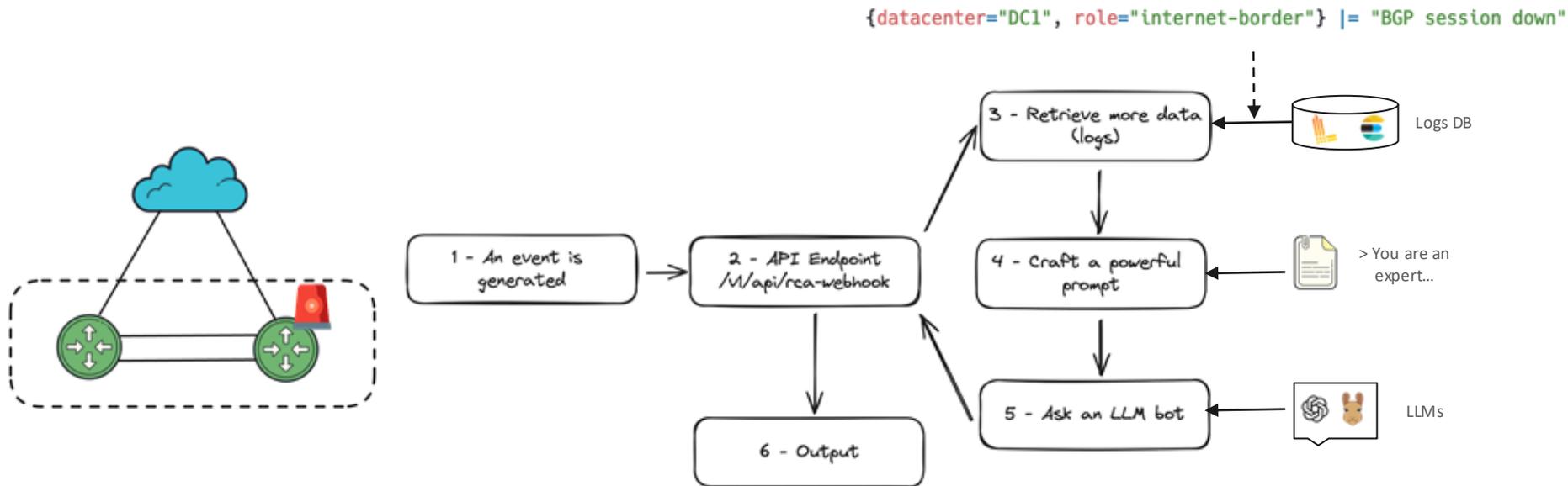
Why Modular Observability Is Key

- **Build a tailored stack**—don't reinvent the wheel.
- **Flexibility**: Choose the **right tools** for **the right job**.
- Reduced complexity and controlled scalability.

SRE / DevOps / Platform

elastic

Grafana Labs

We Are Here

NetOps / NRE / NetDev

Collection & Processing

Agent

Agent

Leverage Teams /
Outsource

AIOps / Alerts & Incidents

PagerDuty

Keep
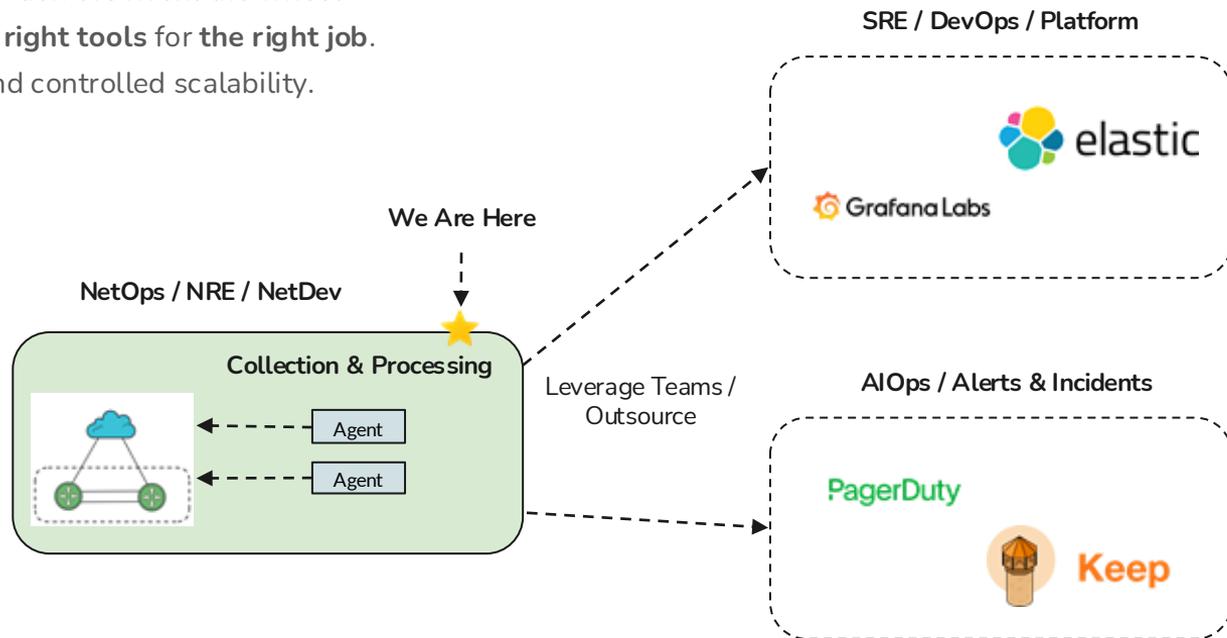
Do you want to test it?

- **Setup a PoC**

https://github.com/network-observability/network-observability-lab

Thank you!