openmetrics-exporter

Do more with your metrics

Last9

Piyush Verma

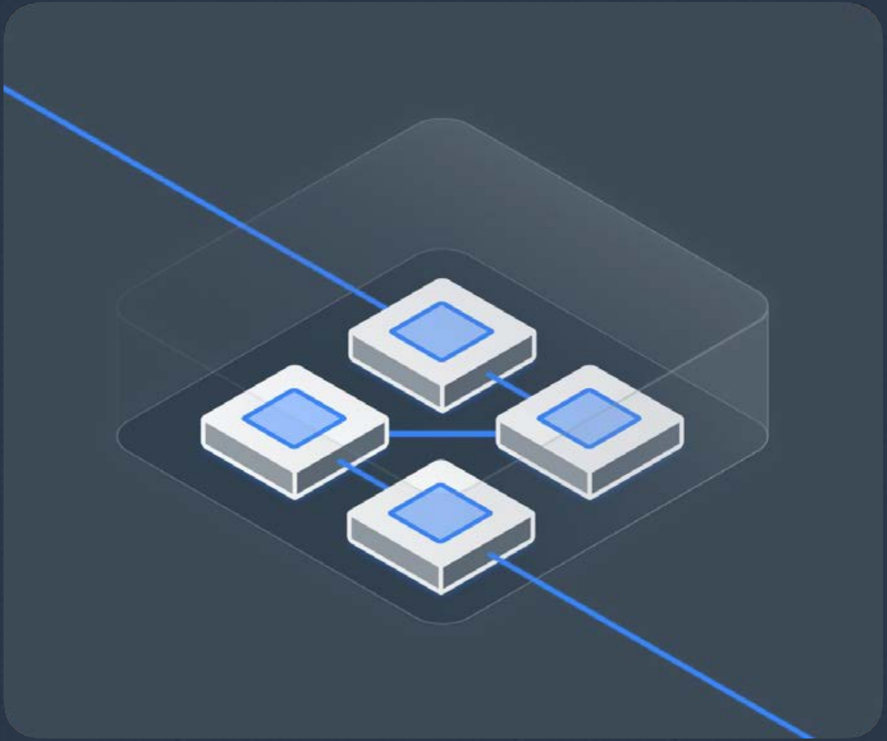
CTO, Co-Founder

Last9.io

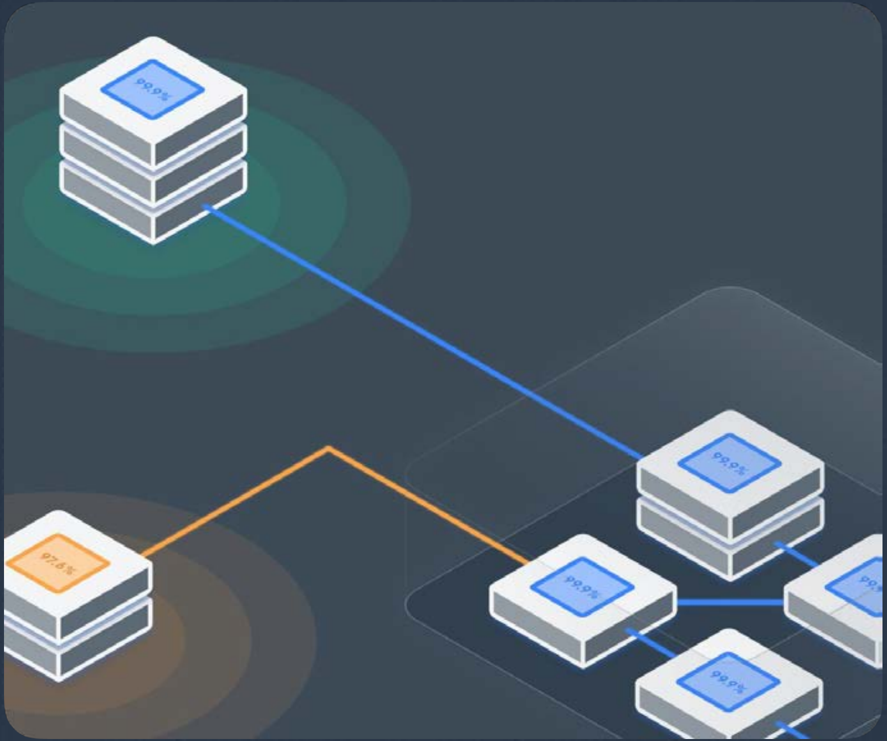
Yet another exporter?

Modern cloud components are built on complex layers.

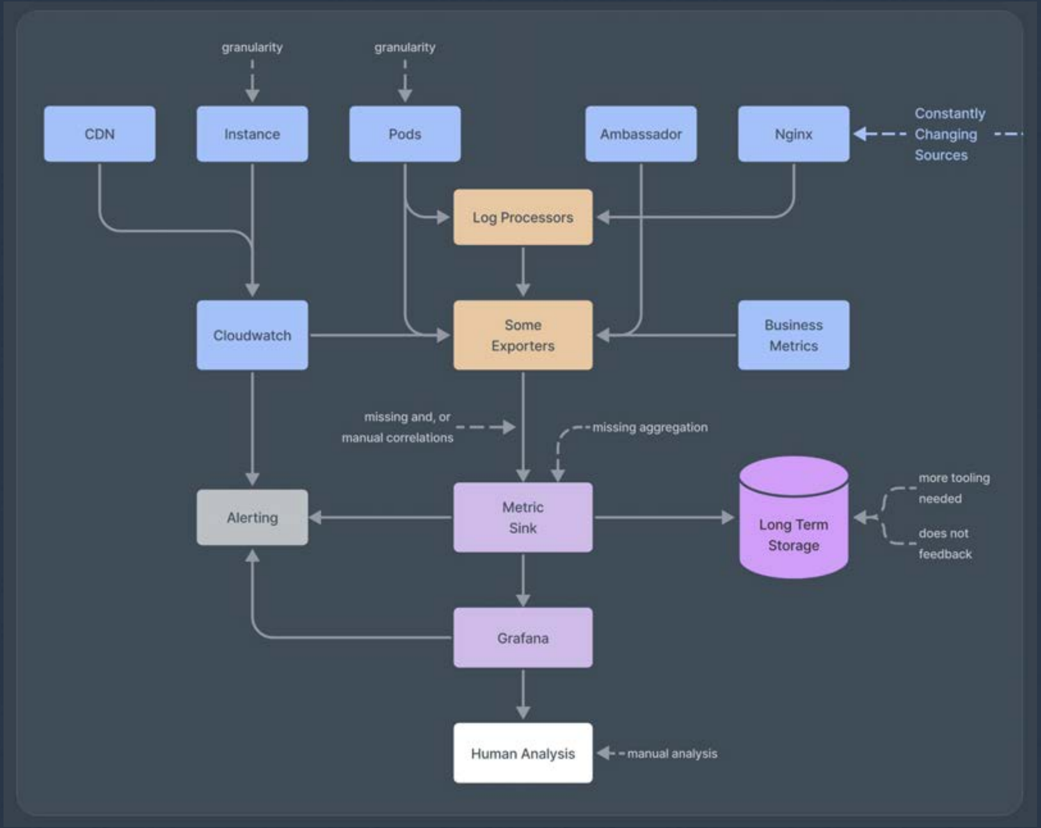
And they keep talking to each other.



And they break, all the time.



How do we observe such dynamicity?



Observability Landscape, as of today

Last9

Openmetrics-Exporter

go_gc_duration_seconds

Have you ever alerted on this?

40% of your metrics

might not be accessed, ever.

How do you observe an EKS Cluster?

kube_state_metrics or cloudwatch-eks, or both?

Do you know if your exporter is

lagging, crashing, or burning your Cloudwatch $$$

Openmetrics-Exporter

🤔

Key challenges

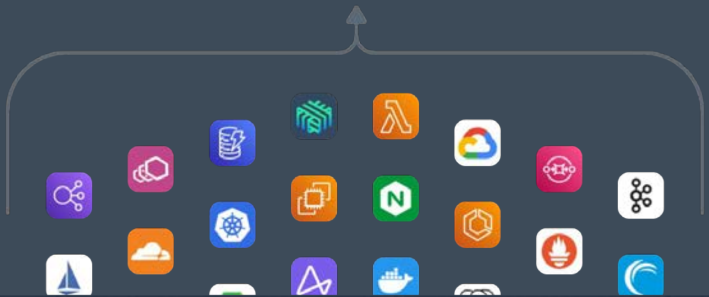
- sources are changing
- no correlations
- metrics explosion
- new source == new exporter

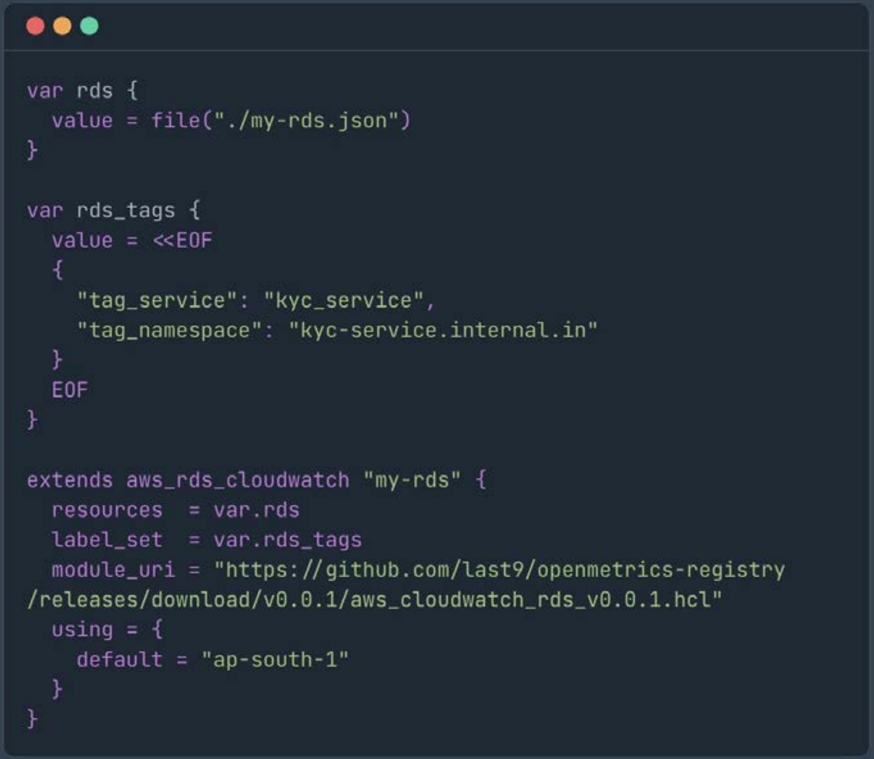
Introducing:
Observability-as-code

`Openmetrics-Exporter`

Step1 : Declare

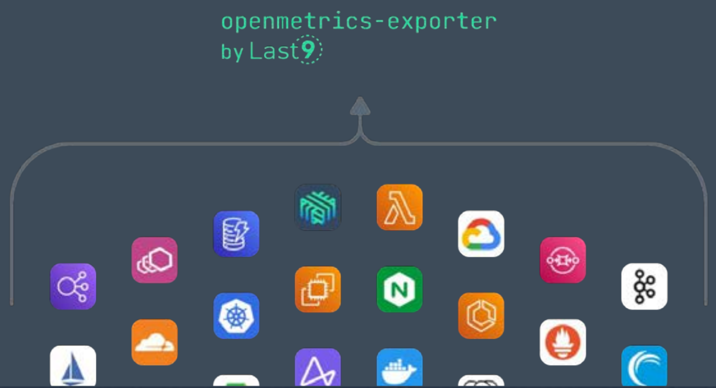
openmetrics-exporter
by Last9

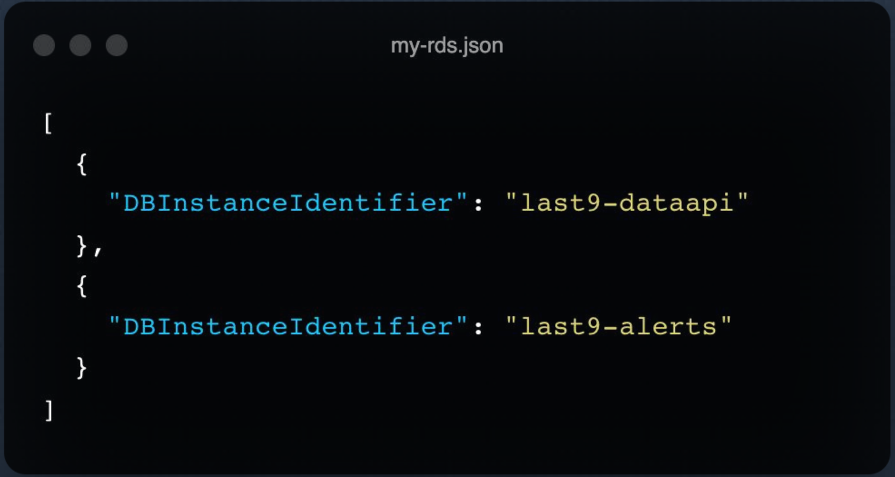




```
var rds {
  value = file("./my-rds.json")
}

var rds_tags {
  value = <<EOF
  {
    "tag_service": "kyc_service",
    "tag_namespace": "kyc-service.internal.in"
  }
  EOF
}

extends aws_rds_cloudwatch "my-rds" {
  resources  = var.rds
  label_set  = var.rds_tags
  module_uri = "https://github.com/last9/openmetrics-registry
/releases/download/v0.0.1/aws_cloudwatch_rds_v0.0.1.hcl"
  using = {
    default = "ap-south-1"
  }
}
```

Step1 : Declare



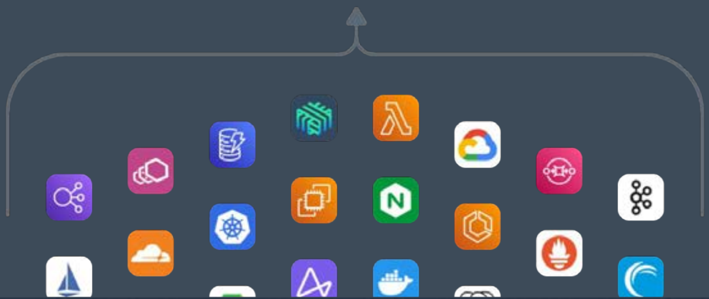


```
[
  {
    "DBInstanceIdentifier": "last9-dataapi"
  },
  {
    "DBInstanceIdentifier": "last9-alerts"
  }
]
```

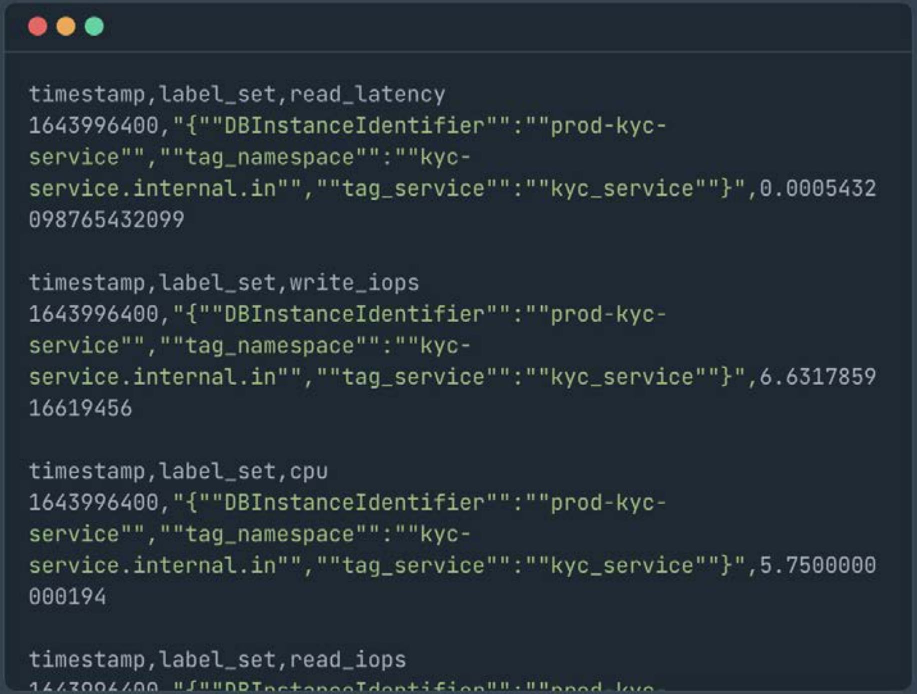
Last9

Openmetrics-Exporter

Step2 : Plan



openmetrics-exporter
by Last9

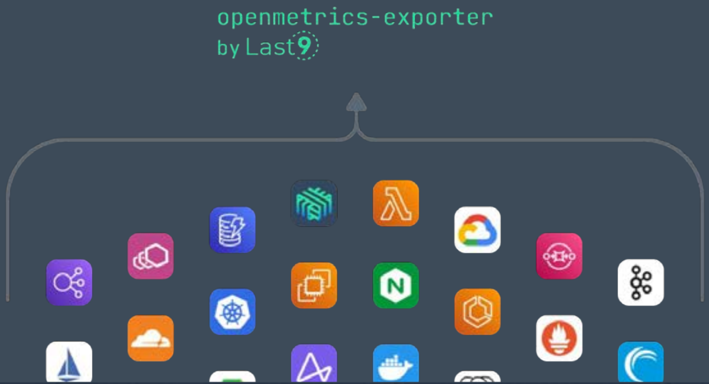


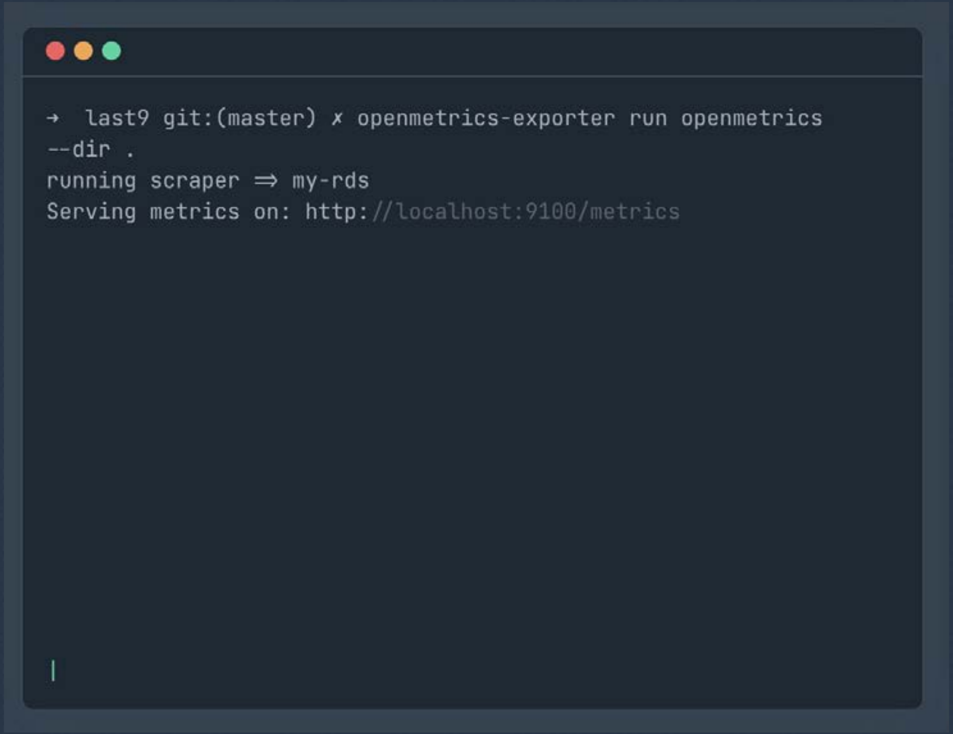
```
timestamp,label_set,read_latency
1643996400,"{""DBInstanceIdentifier"":""prod-kyc-
service"",""tag_namespace"":""kyc-
service.internal.in"",""tag_service"":""kyc_service""}",0.0005432
098765432099

timestamp,label_set,write_iops
1643996400,"{""DBInstanceIdentifier"":""prod-kyc-
service"",""tag_namespace"":""kyc-
service.internal.in"",""tag_service"":""kyc_service""}",6.6317859
16619456

timestamp,label_set,cpu
1643996400,"{""DBInstanceIdentifier"":""prod-kyc-
service"",""tag_namespace"":""kyc-
service.internal.in"",""tag_service"":""kyc_service""}",5.7500000
000194

timestamp,label_set,read_iops
1643996400,"{""DBInstanceIdentifier"":""prod-kyc-
```

Last9

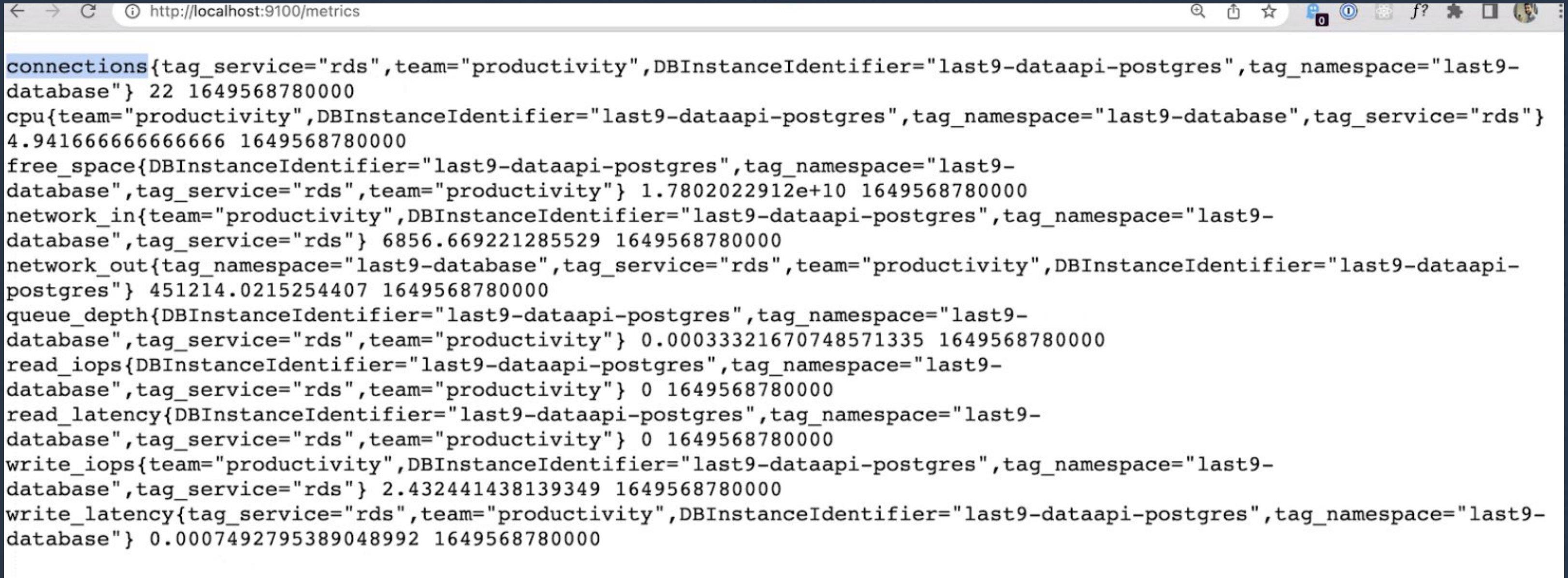
Openmetrics-Exporter

Step3 : Dispatch





```
→  last9 git:(master) ✗ openmetrics-exporter run openmetrics
--dir .
running scraper ⇒ my-rds
Serving metrics on: http://localhost:9100/metrics
```

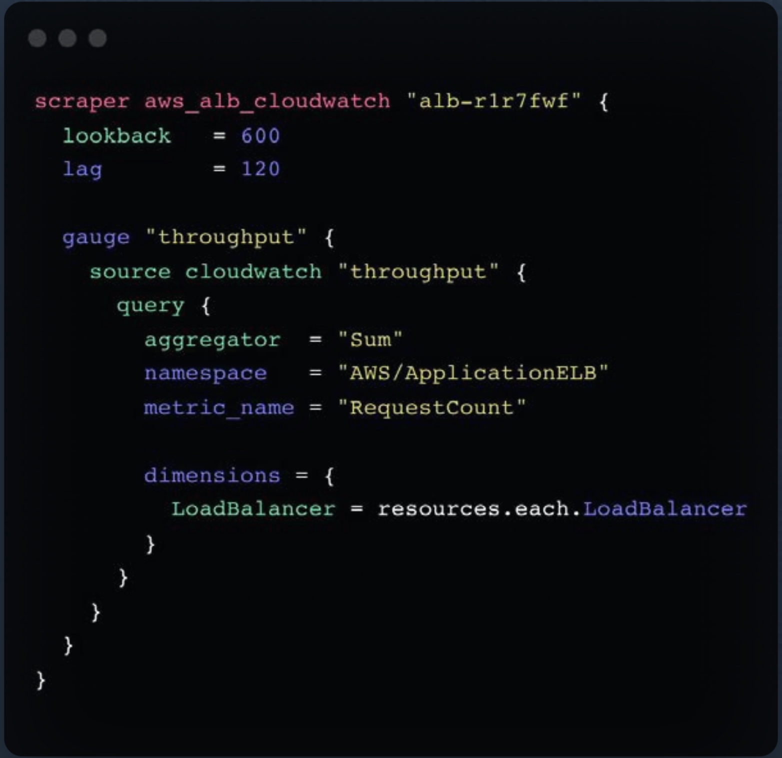


```
connections{tag_service="rds",team="productivity",DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database"} 22 1649568780000
cpu{team="productivity",DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-database",tag_service="rds"}
4.941666666666666 1649568780000
free_space{DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds",team="productivity"} 1.7802022912e+10 1649568780000
network_in{team="productivity",DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds"} 6856.669221285529 1649568780000
network_out{tag_namespace="last9-database",tag_service="rds",team="productivity",DBInstanceIdentifier="last9-dataapi-
postgres"} 451214.0215254407 1649568780000
queue_depth{DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds",team="productivity"} 0.00033321670748571335 1649568780000
read_iops{DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds",team="productivity"} 0 1649568780000
read_latency{DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds",team="productivity"} 0 1649568780000
write_iops{team="productivity",DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database",tag_service="rds"} 2.432441438139349 1649568780000
write_latency{tag_service="rds",team="productivity",DBInstanceIdentifier="last9-dataapi-postgres",tag_namespace="last9-
database"} 0.0007492795389048992 1649568780000
```

Seeing is believing

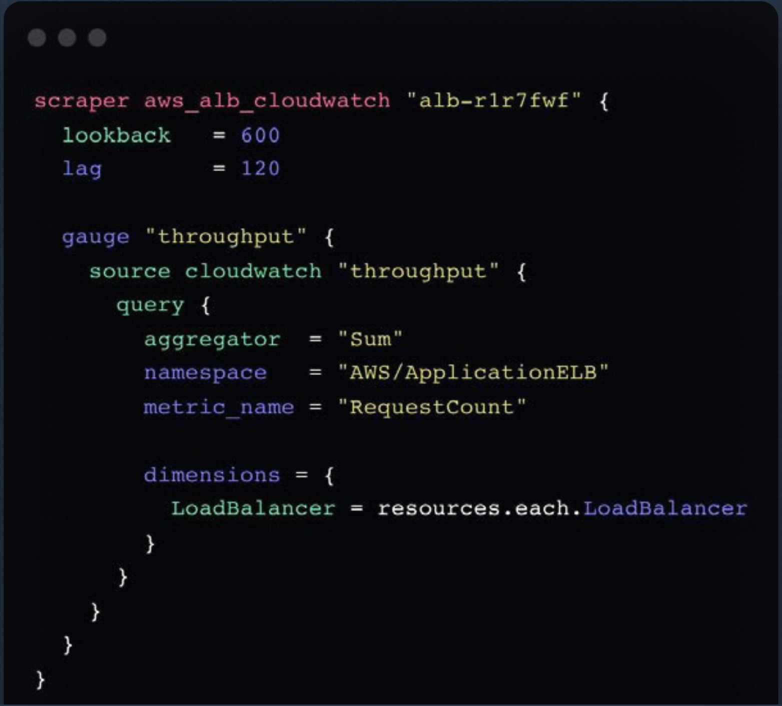
Openmetrics-Exporter

Anatomy of an openmetrics-exporter file

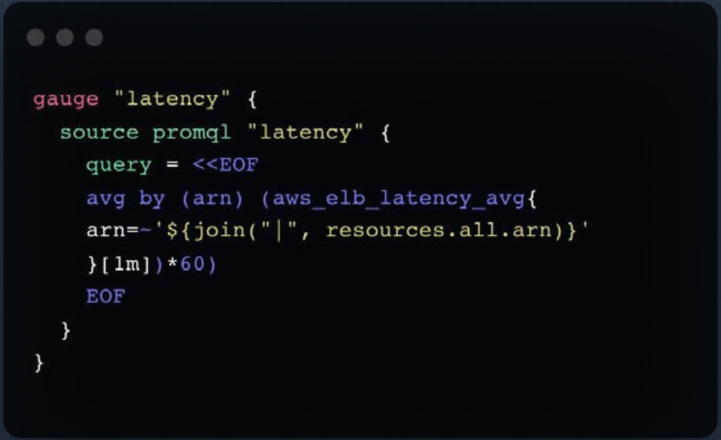


```
scraper aws_alb_cloudwatch "alb-r1r7fwf" {
  lookback    = 600
  lag         = 120

  gauge "throughput" {
    source cloudwatch "throughput" {
      query {
        aggregator  = "Sum"
        namespace   = "AWS/ApplicationELB"
        metric_name = "RequestCount"

        dimensions = {
          LoadBalancer = resources.each.LoadBalancer
        }
      }
    }
  }
}
```

Building correlations



```
scraper aws_alb_cloudwatch "alb-r1r7fwf" {
  lookback    = 600
  lag         = 120

  gauge "throughput" {
    source cloudwatch "throughput" {
      query {
        aggregator  = "Sum"
        namespace   = "AWS/ApplicationELB"
        metric_name = "RequestCount"

        dimensions = {
          LoadBalancer = resources.each.LoadBalancer
        }
      }
    }
  }
}
```

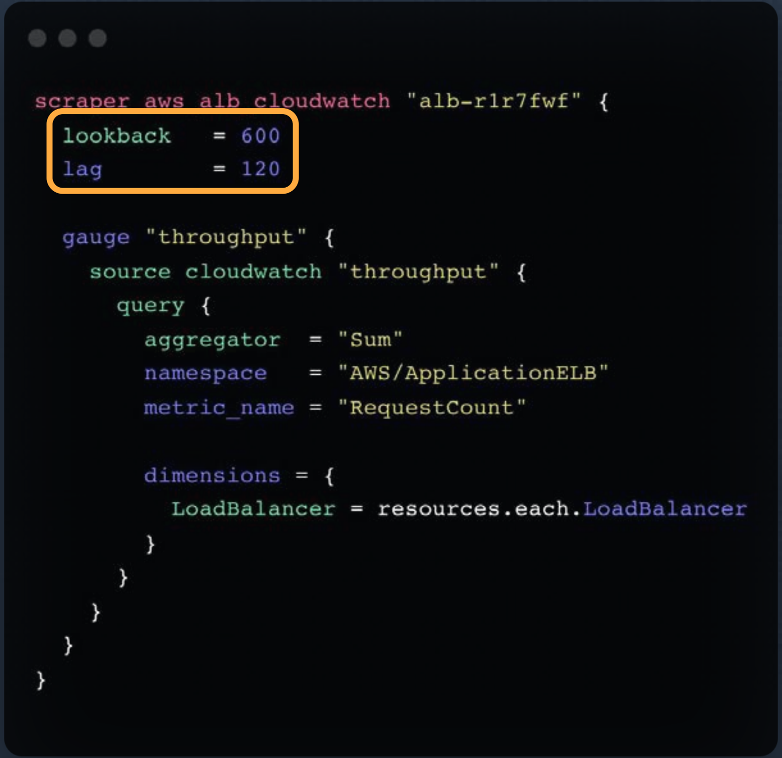


```
gauge "latency" {
  source promql "latency" {
    query = <<EOF
    avg by (arn) (aws_elb_latency_avg{
    arn=~'${join("|", resources.all.arn)}'
    }[1m])*60)
    EOF
  }
}
```

Last9

Openmetrics-Exporter

Keeping up with changing values



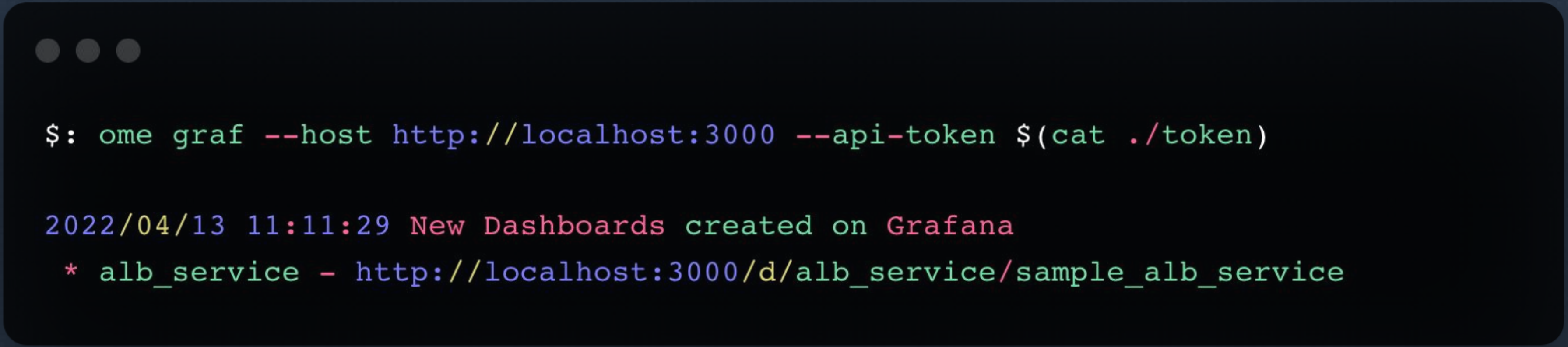
```
scraper aws_alb_cloudwatch "alb-r1r7fwf" {
  lookback    = 600
  lag         = 120

  gauge "throughput" {
    source cloudwatch "throughput" {
      query {
        aggregator  = "Sum"
        namespace   = "AWS/ApplicationELB"
        metric_name = "RequestCount"

        dimensions = {
          LoadBalancer = resources.each.LoadBalancer
        }
      }
    }
  }
}
```

Last9

Openmetrics-Exporter

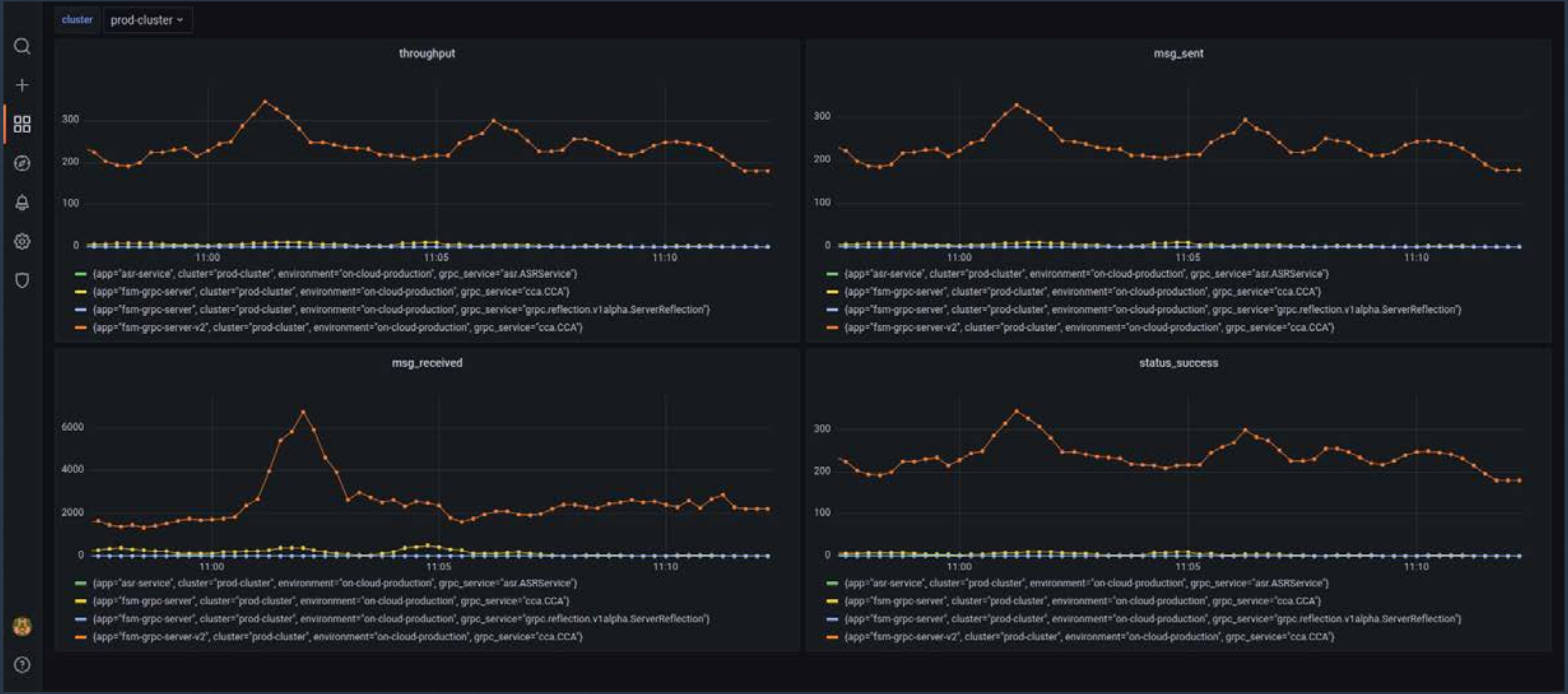
Faster feedback loop

openmetrics-exporter → Grafana



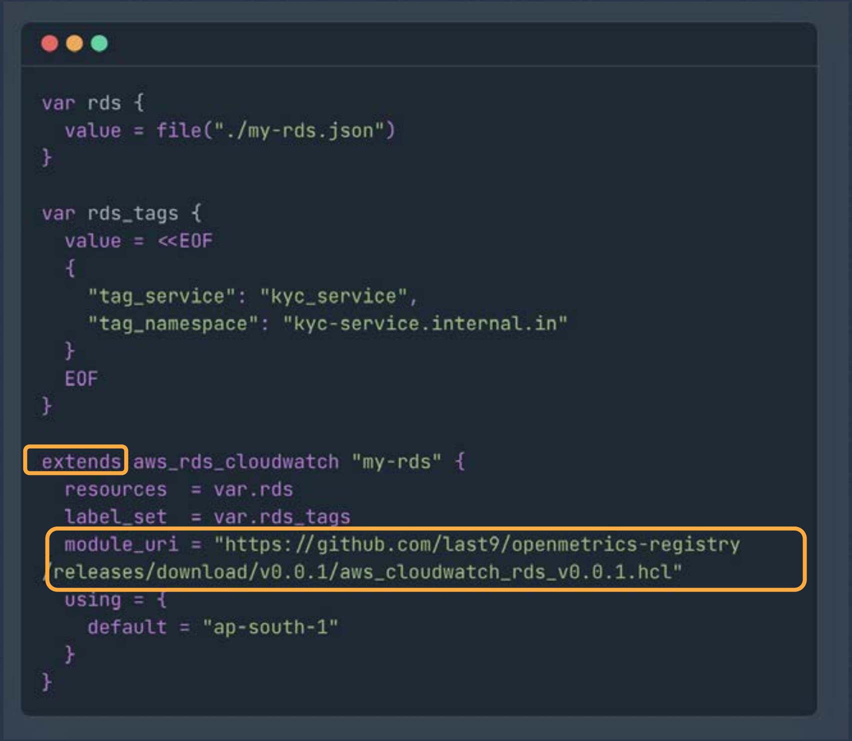
```
$: ome graf --host http://localhost:3000 --api-token $(cat ./token)

2022/04/13 11:11:29 New Dashboards created on Grafana
 * alb_service - http://localhost:3000/d/alb_service/sample_alb_service
```

openmetrics-exporter → Grafana

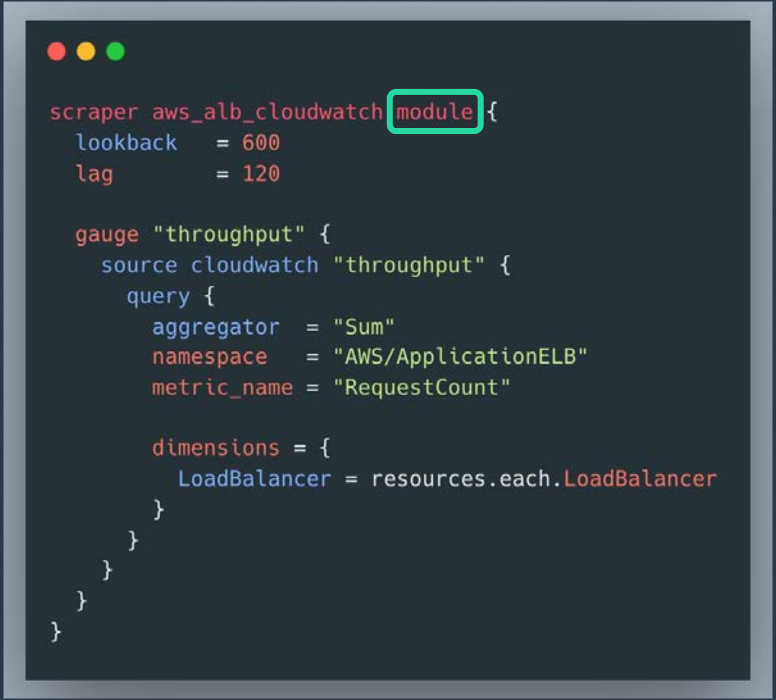

cluster
prod-cluster
throughput
300
200
100
0
11:00
11:05
11:10
{app="asr-service", cluster="prod-cluster", environment="on-cloud-production", grpc_service="asr.ASRService"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="grpc.reflection.v1alpha.ServerReflection"}
{app="fsm-grpc-server-v2", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
msg_sent
300
200
100
0
11:00
11:05
11:10
{app="asr-service", cluster="prod-cluster", environment="on-cloud-production", grpc_service="asr.ASRService"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="grpc.reflection.v1alpha.ServerReflection"}
{app="fsm-grpc-server-v2", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
msg_received
6000
4000
2000
0
11:00
11:05
11:10
{app="asr-service", cluster="prod-cluster", environment="on-cloud-production", grpc_service="asr.ASRService"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="grpc.reflection.v1alpha.ServerReflection"}
{app="fsm-grpc-server-v2", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
status_success
300
200
100
0
11:00
11:05
11:10
{app="asr-service", cluster="prod-cluster", environment="on-cloud-production", grpc_service="asr.ASRService"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}
{app="fsm-grpc-server", cluster="prod-cluster", environment="on-cloud-production", grpc_service="grpc.reflection.v1alpha.ServerReflection"}
{app="fsm-grpc-server-v2", cluster="prod-cluster", environment="on-cloud-production", grpc_service="cca.CCA"}

Reusability

Built on principles of software engineering



```
var rds {
  value = file("./my-rds.json")
}

var rds_tags {
  value = <<EOF
  {
    "tag_service": "kyc_service",
    "tag_namespace": "kyc-service.internal.in"
  }
  EOF
}

extends aws_rds_cloudwatch "my-rds" {
  resources  = var.rds
  label_set  = var.rds_tags
  module_uri = "https://github.com/last9/openmetrics-registry
/releases/download/v0.0.1/aws_cloudwatch_rds_v0.0.1.hcl"
  using = {
    default = "ap-south-1"
  }
}
```



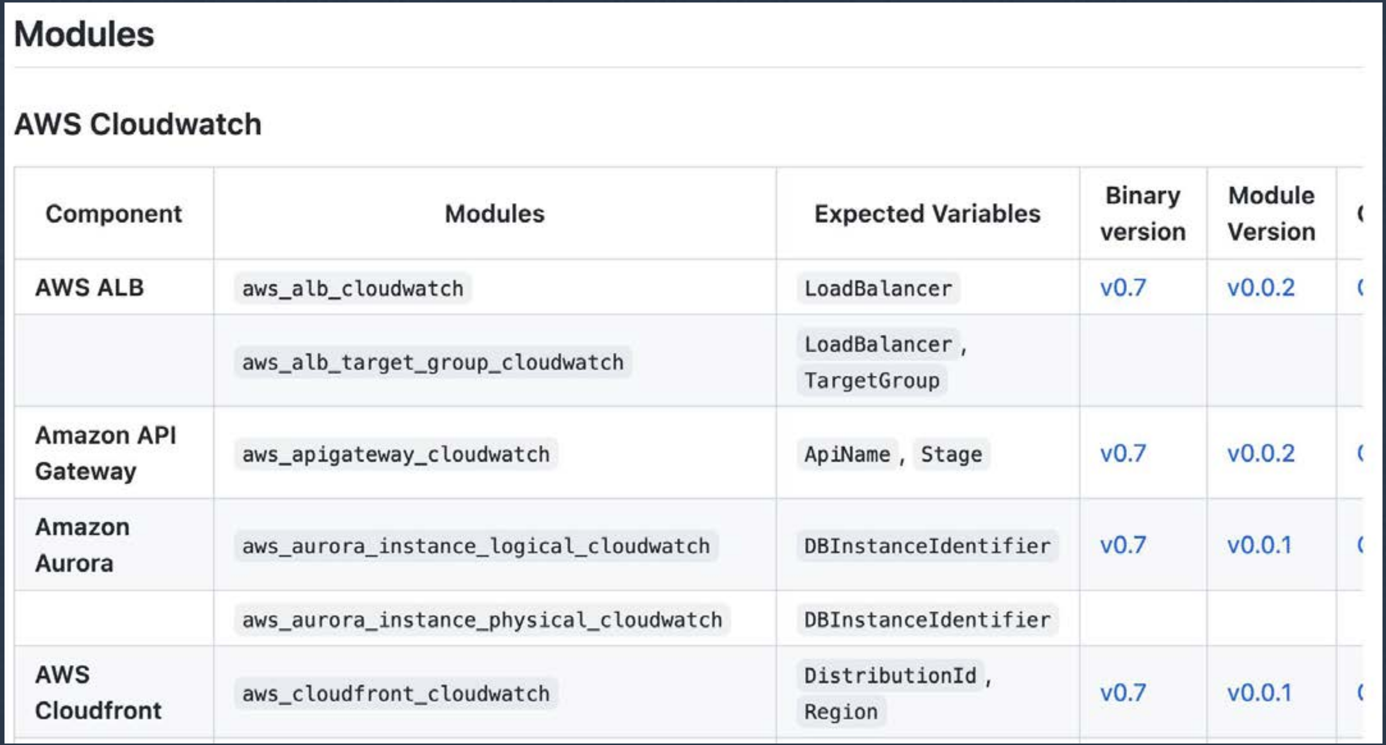
```
scraper aws_alb_cloudwatch module {
  lookback    = 600
  lag         = 120

  gauge "throughput" {
    source cloudwatch "throughput" {
      query {
        aggregator  = "Sum"
        namespace   = "AWS/ApplicationELB"
        metric_name = "RequestCount"

        dimensions = {
          LoadBalancer = resources.each.LoadBalancer
        }
      }
    }
  }
}
```

modules

Openmetrics-Exporter

Catalog

openmetrics-exporter modules



Modules

AWS Cloudwatch

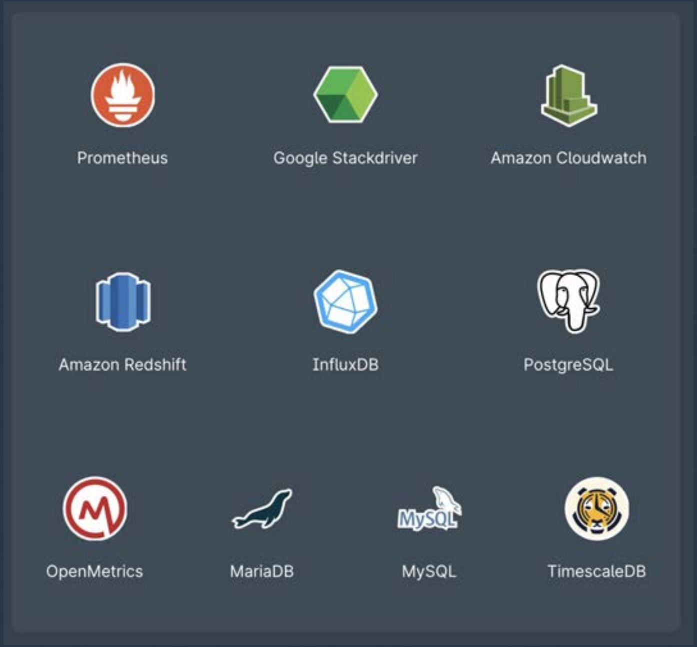
Component	Modules	Expected Variables	Binary version	Module Version	(
AWS ALB	`aws_alb_cloudwatch`	`LoadBalancer`	v0.7	v0.0.2	(
	`aws_alb_target_group_cloudwatch`	`LoadBalancer` , `TargetGroup`			
Amazon API Gateway	`aws_apigateway_cloudwatch`	`ApiName` , `Stage`	v0.7	v0.0.2	(
Amazon Aurora	`aws_aurora_instance_logical_cloudwatch`	`DBInstanceIdentifier`	v0.7	v0.0.1	(
	`aws_aurora_instance_physical_cloudwatch`	`DBInstanceIdentifier`			
AWS Cloudfront	`aws_cloudfront_cloudwatch`	`DistributionId` , `Region`	v0.7	v0.0.1	(

Last9

Head to Registry

Openmetrics-Exporter

Data sources



Prometheus

Google Stackdriver

Amazon Cloudwatch

Amazon Redshift

InfluxDB

PostgreSQL

OpenMetrics

MariaDB

MySQL

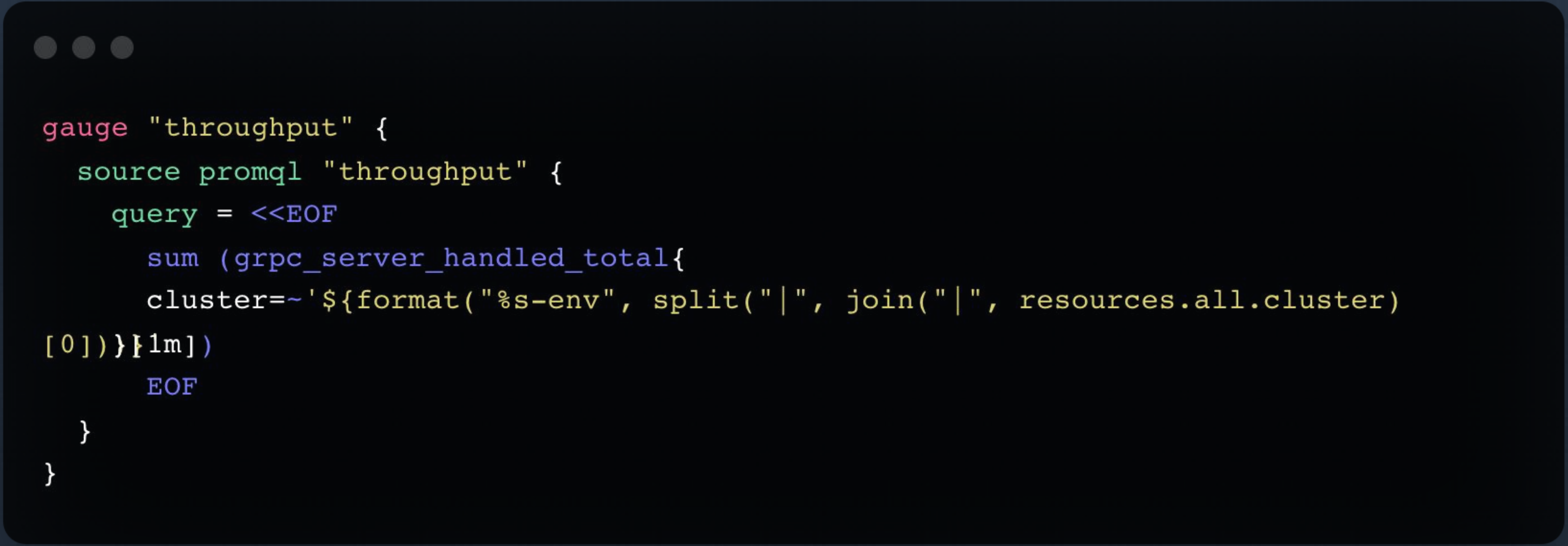
TimescaleDB

Openmetrics-Exporter

Why HCL?
Why not YAML?

YAML provides you Job-Security; HCL let’s you ‘concentrate’ 🤪

Why not YAML?



```
gauge "throughput" {
  source promql "throughput" {
    query = <<EOF
      sum (grpc_server_handled_total{
      cluster=~'${format("%s-env", split("|", join("|", resources.all.cluster)
[0])}[1m])
      EOF
  }
}
```

Why HCL?

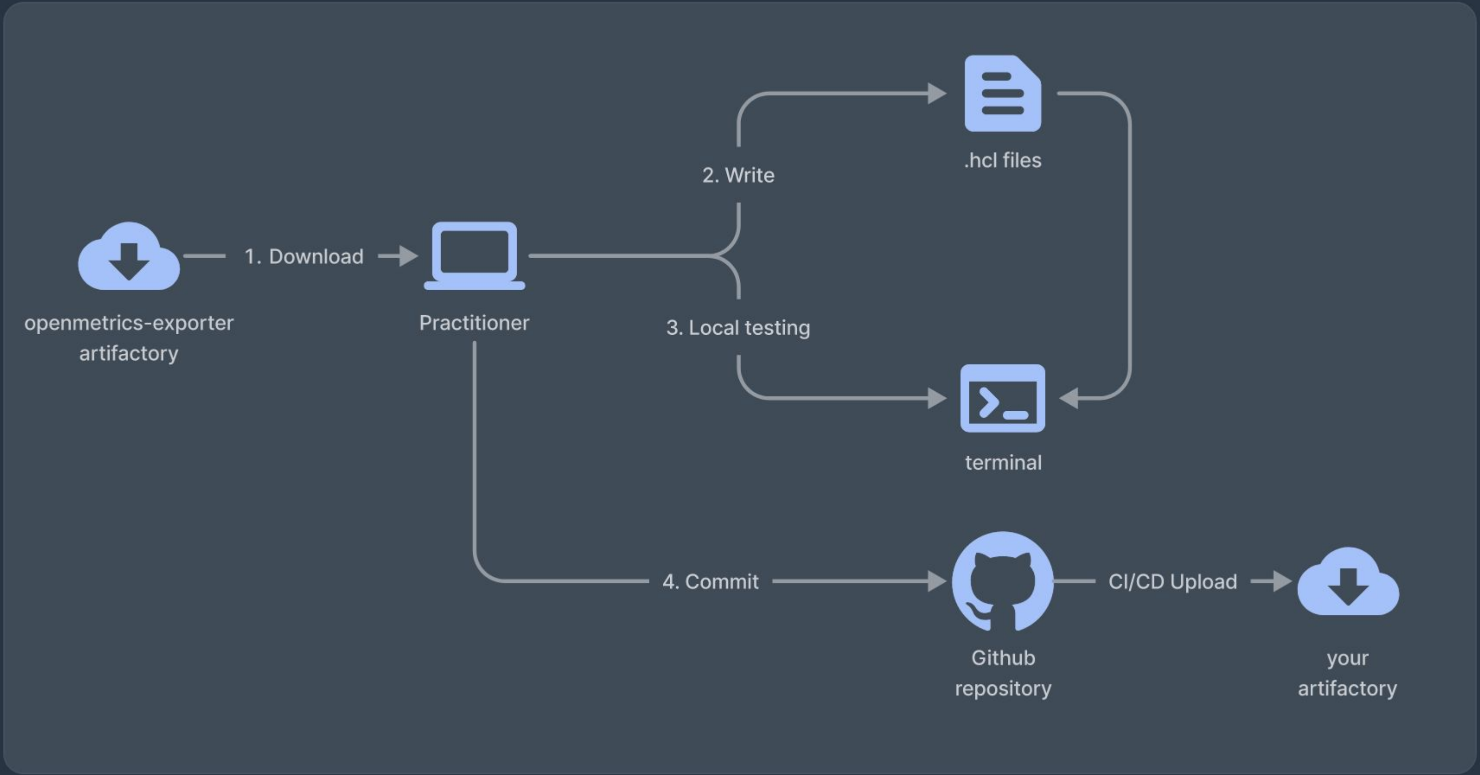
- Same as Terraform, Consul, or Nomad.
- Fits your existing editors.
- No learning curve.
- Existing GitOps.
- First class expressions and logic evaluation.

How was this different from prometheus-exporters?

How was it different from existing approach?

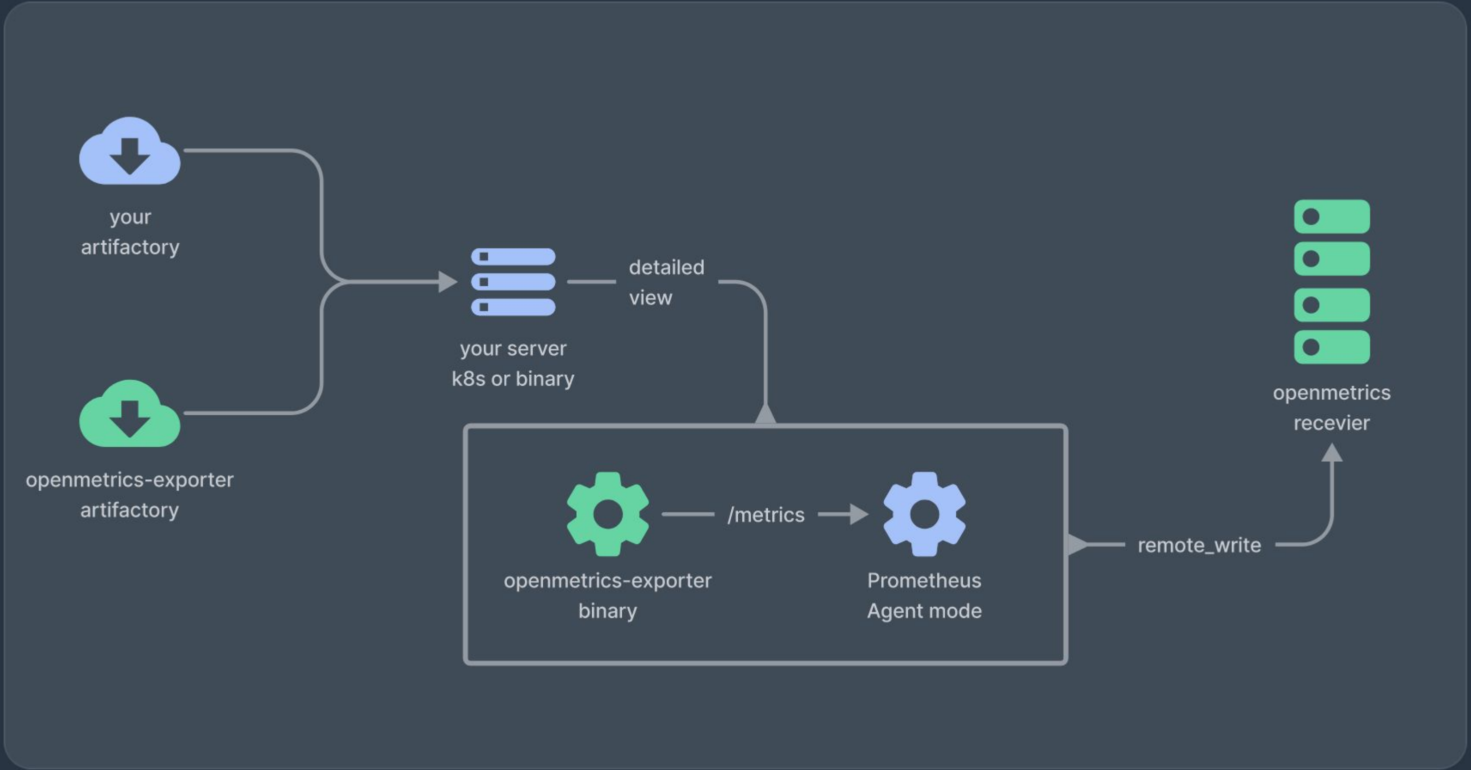
Scenario	Existing exporters	openmetrics-exporter
New Source of Data	❌ New Binary	✅ Just another Scraper
What Metrics?	❌ No way too choose.	✅ Pick and Choose.
Building Correlations	❌ Post-Processing	✅ Native support.
Logic Manipulation	❌ Not supported	✅ Native expressions

Putting it all together



Step 1: Build and Ship

Openmetrics-Exporter



Step 2: Dispatch and Run

Last9

Openmetrics-Exporter

Born at Last9

Enabling you to ship reliable software.

Last9

Openmetrics-Exporter

Visit

https://last9.io/openmetrics-exporter