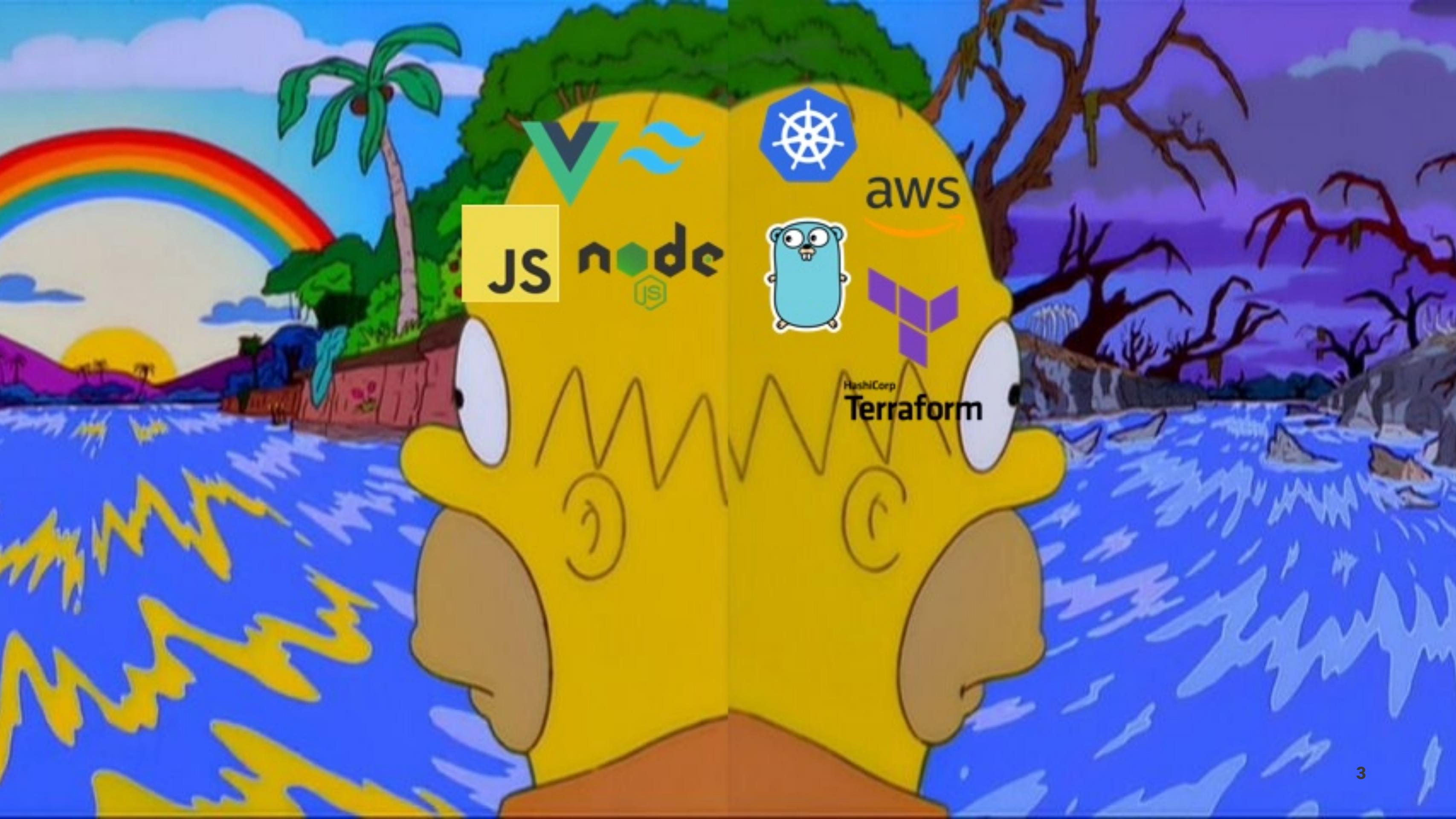


Collect, Ingest, Observe!

A New Era for (free) JS Monitoring







**EL TIPO PUEDE CAMBIAR
DE TODO. DE CARA DE CASA
DE FAMILIA DE NOVIA DE RELIGION
DE DIOS. PERO HAY UNA
COSA QUE NO PUEDE
CAMBIAR.
NO PUEDE CAMBIAR
DE PASION**





Nacho Anaya



Cloud Engineer @AltinityDB



@ianaya89



...



Turner Novak
@TurnerNovak

Coinbase (?) had a \$65 million Datadog bill per its Q1 earnings call. Wild.

h/t [@ChairliftCap](#)



Mark Ronald Murphy
JPMorgan Chase & Co, Research Division

David, looking at the math on this large upfront bill that did not recur, it seems to be about \$65 million, if I'm running that correctly. Can you possibly shed a little more light? For instance, will you recapture that or some of that in Q2? And what type of customer and customer dynamic is operating at that level? And then I have a quick follow-up.

Copyright © 2023 S&P Global Market Intelligence, a division of S&P Global Inc. All Rights reserved.
[spglobal.com/marketintelligence](#)

10

DATADOG, INC. FQ1 2023 EARNINGS CALL MAY 04, 2023

David M. Obstler
Chief Financial Officer

Yes. We -- that is a customer of ours. What we said was the billing frequency changed and the size. So that customer's bill will, one, be spread out more over time. That company -- that was a crypto company and continues to be a customer of ours, but that was an early optimizer. We had always talked that some of the industries that were most affected optimized. And that is -- so we will get that -- we will get that bill at a smaller size than was billed last year in a more of a chunked up billing way.

10:02 PM · May 5, 2023 · 605.2K Views

99 Retweets 134 Quotes 1,034 Likes 243 Bookmarks





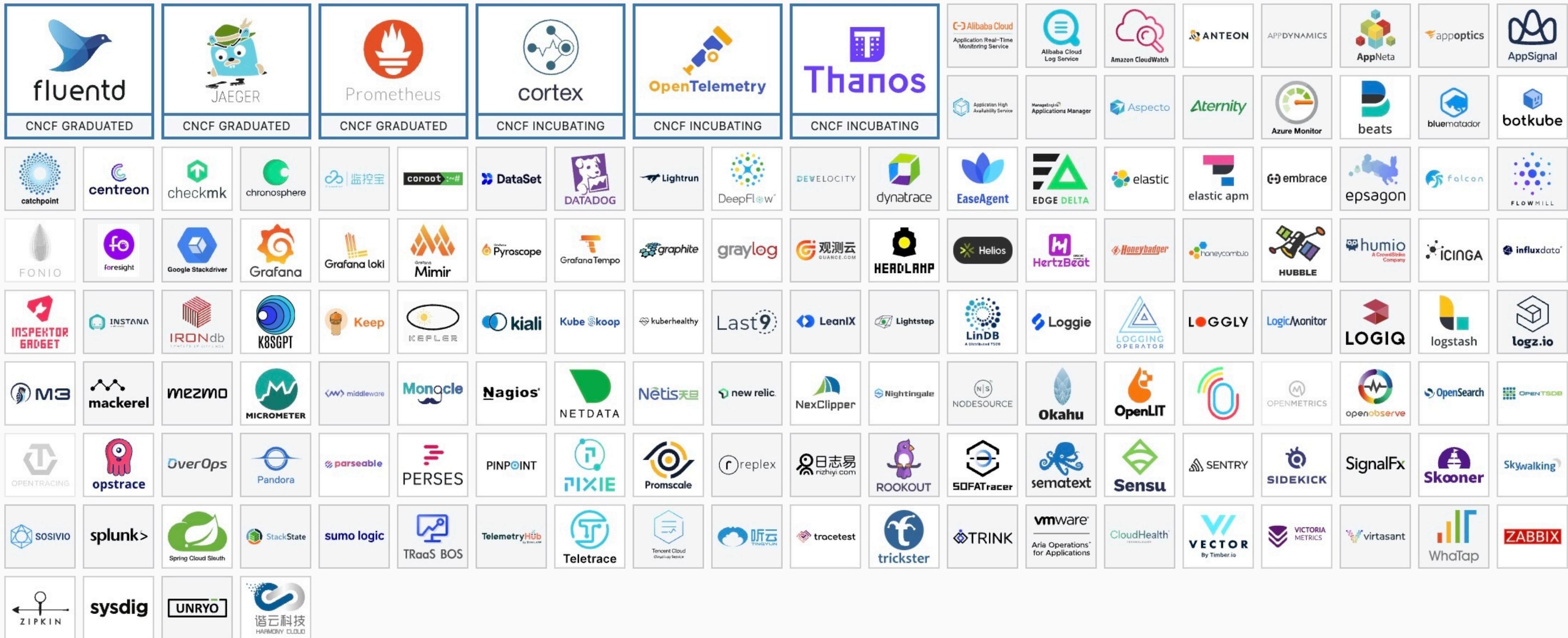


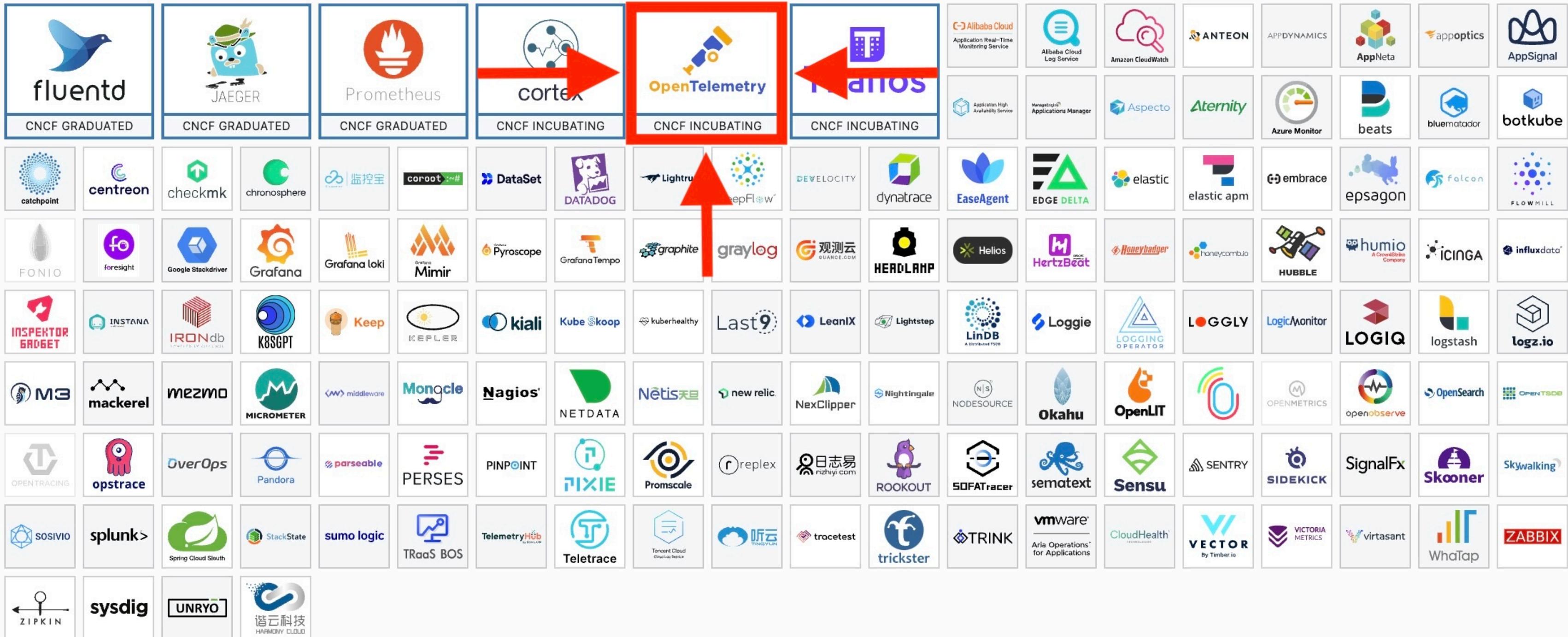












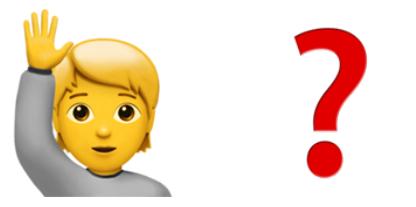
Observability

o11y





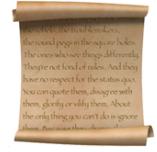






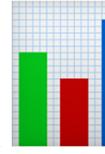
Señales





Logs

```
{  
    "timestamp": "2024-08-04T12:34:56.789Z",  
    "level": "INFO",  
    "service": "user-authentication",  
    "environment": "production",  
    "message": "User login successful",  
    "context": {  
        "userId": "12345",  
        "username": "johndoe",  
        "ipAddress": "192.168.1.1",  
        "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64)..."  
    },  
    "transactionId": "abcd-efgh-ijkl-mnop",  
    "duration": 200,  
    "request": {  
        "method": "POST",  
        "url": "/api/v1/login",  
        "headers": {  
            "Content-Type": "application/json",  
            "Accept": "application/json"  
        }  
    },  
    "response": {  
        "statusCode": 200,  
        "body": {  
            "success": true,  
            "token": "jwt-token-here"  
        }  
    }  
}
```

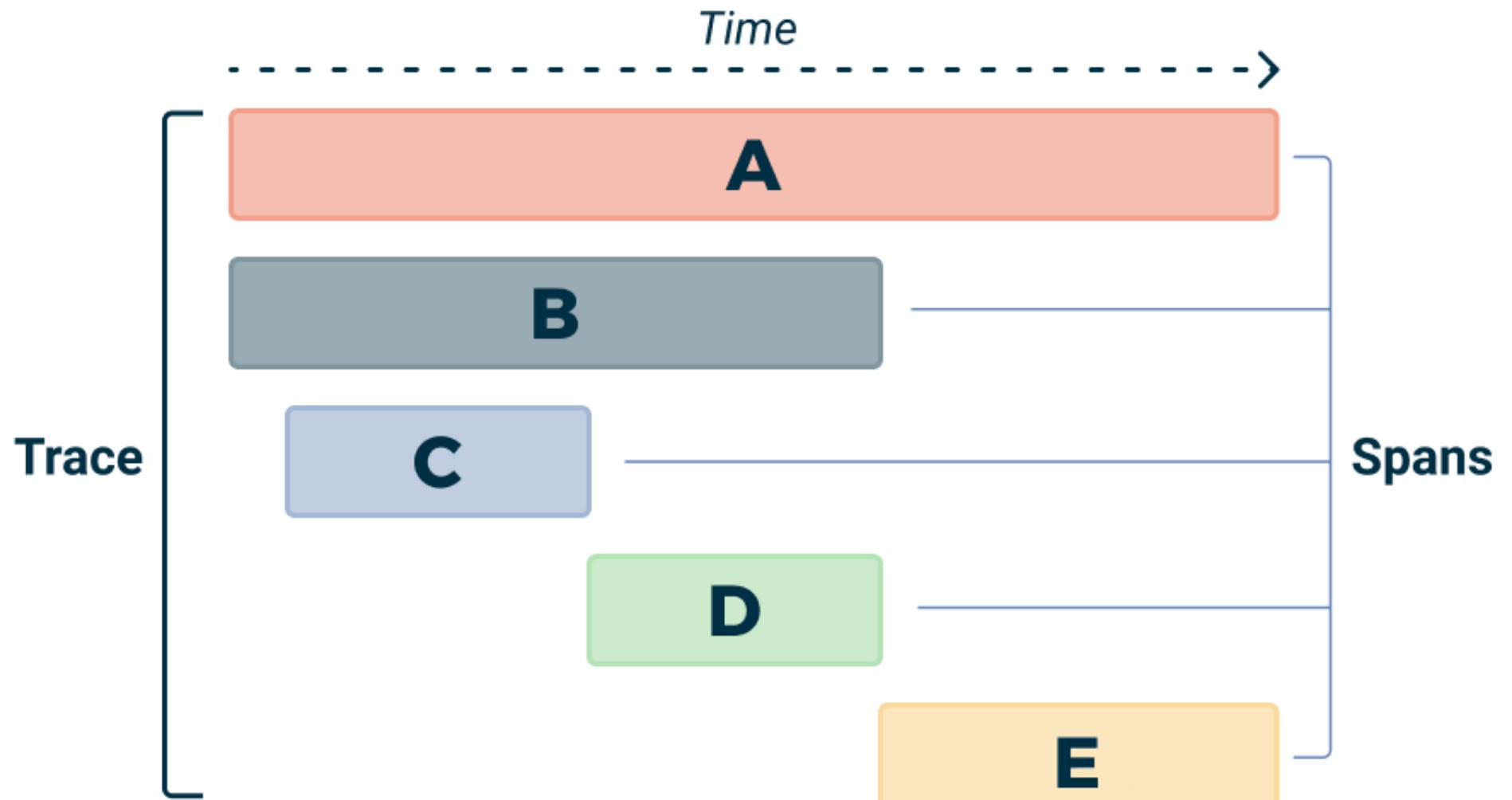
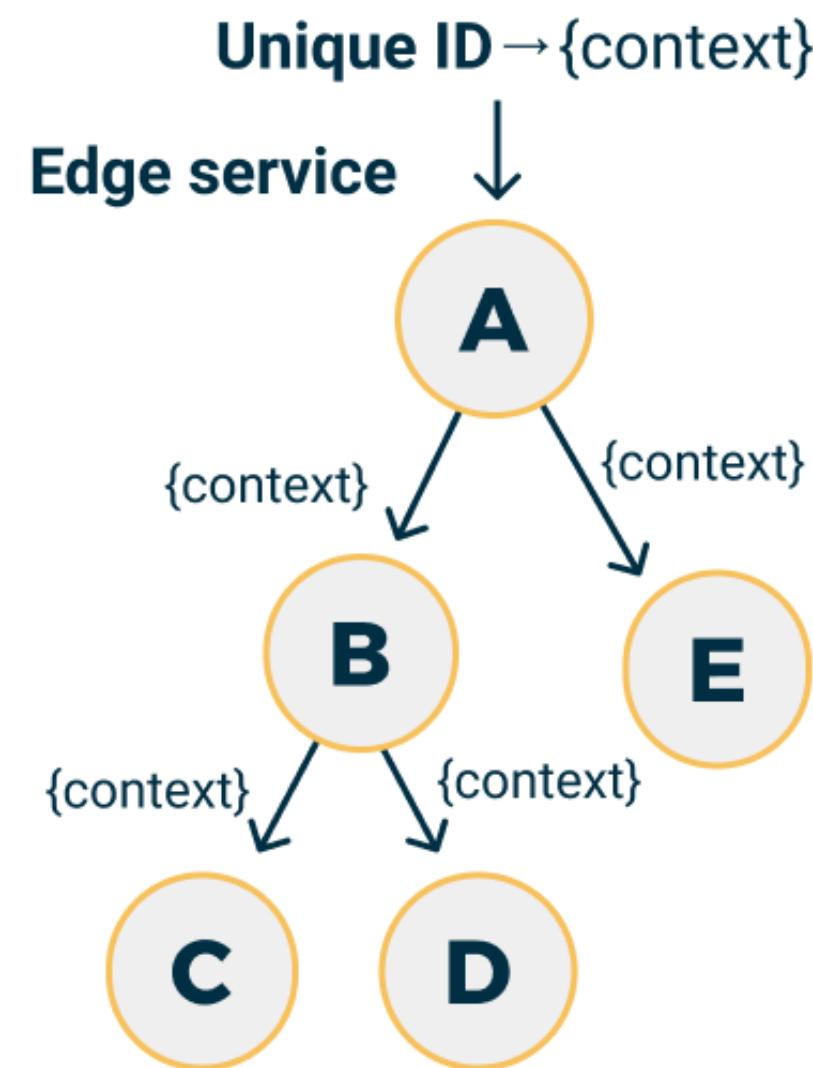


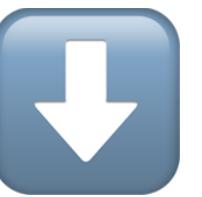
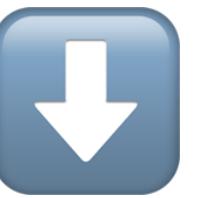
Metrics

```
{  
    "ScopeName": "@opentelemetry/instrumentation-pg",  
    "ScopeVersion": "0.47.0",  
    "ScopeDroppedAttrCount": 0,  
    "ScopeSchemaUrl": "",  
    "MetricName": "http.client.duration",  
    "MetricDescription": "Measures HTTP requests.",  
    "MetricUnit": "ms",  
    "Attributes": {  
        "http.method": "POST",  
        "http.status_code": "200"  
    },  
    "StartTimeUnix": "2024-10-31 23:24:18.185000",  
    "TimeUnix": "2024-11-01 14:11:39.321000",  
    "Count": 455,  
    "Sum": 2119.28,  
    "Flags": 0,  
    "Min": 1.262666,  
    "Max": 38.179917  
}
```

 Traces

```
{  
    "name": "hello",  
    "context": {  
        "trace_id": "5b8aa5a2d2c872e8321cf37308d69df2",  
        "span_id": "051581bf3cb55c13"  
    },  
    "parent_id": null,  
    "start_time": "2022-04-29T18:52:58.114201Z",  
    "end_time": "2022-04-29T18:52:58.114687Z",  
    "attributes": {  
        "http.route": "some_route1"  
    },  
    "events": [  
        {  
            "name": "Guten Tag!",  
            "timestamp": "2022-04-29T18:52:58.114561Z",  
            "attributes": {  
                "event_attributes": 1  
            }  
        }  
    ]  
}
```



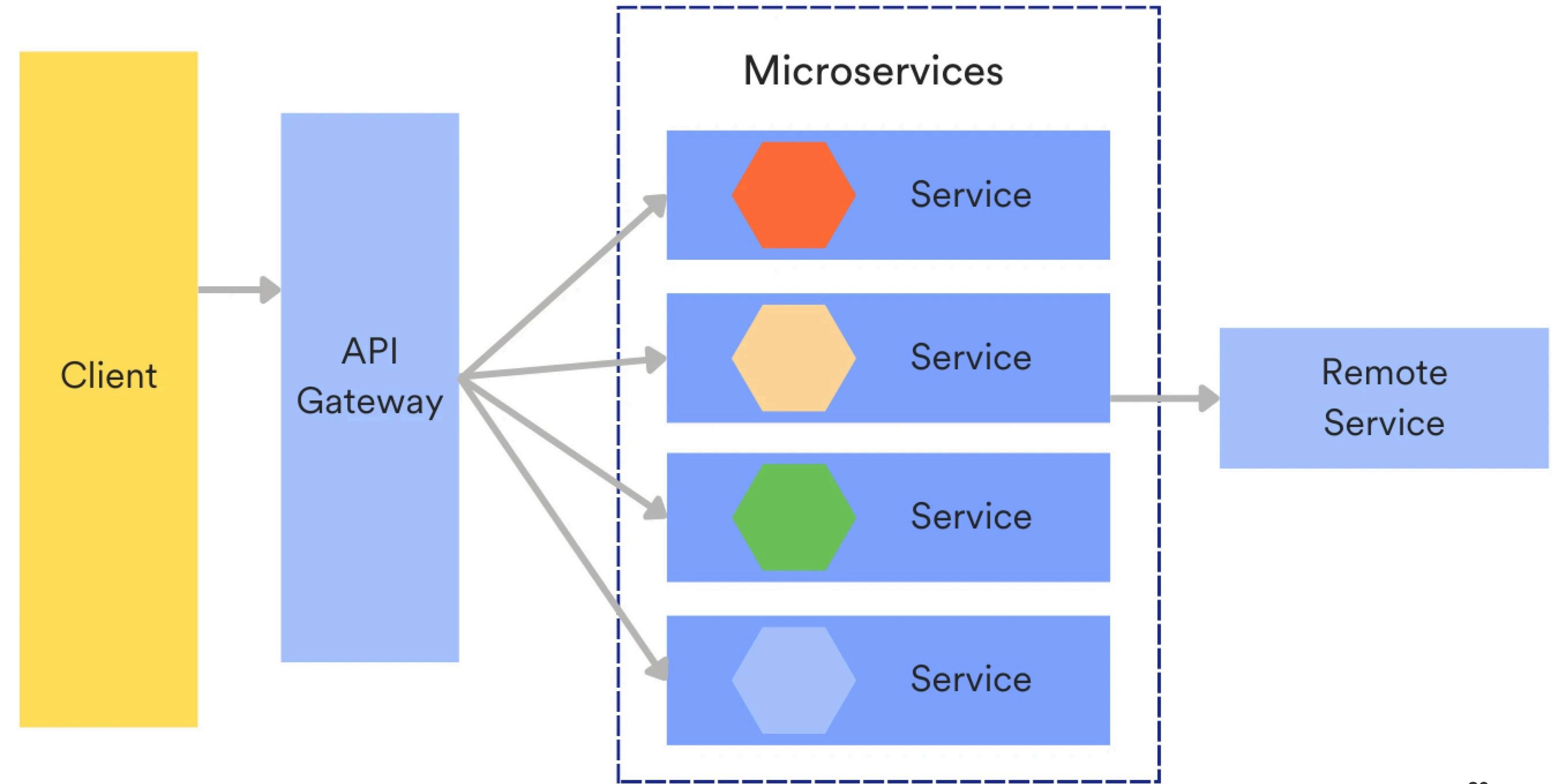


```
$ vmstat -n 2 10
```

procs		memory				swap		io		system		cpu				
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
0	0	343296	21690808	2290104	6897160	0	0	3	187	0	2	4	2	94	0	0
0	0	343296	21690800	2290104	6897160	0	0	0	60	2989	7688	2	1	97	0	0
0	0	343296	21690140	2290104	6897164	0	0	0	72	4704	13677	3	2	95	0	0
0	0	343296	21689888	2290104	6897164	0	0	0	14	3132	9364	2	1	97	0	0
0	0	343296	21690220	2290104	6897168	0	0	0	86	3014	7995	1	1	97	0	0
0	0	343296	21690448	2290104	6897176	0	0	0	20	2660	7297	1	1	98	0	0
0	0	343296	21690268	2290104	6897176	0	0	0	12	2695	7222	1	1	98	0	0
1	0	343296	21690196	2290104	6897180	0	0	0	80	3641	10419	2	1	97	0	0
0	0	343296	21689696	2290104	6897180	0	0	0	14	4108	12605	3	2	95	0	0
0	0	343296	21689900	2290104	6897184	0	0	0	60	2688	7270	2	1	97	0	0















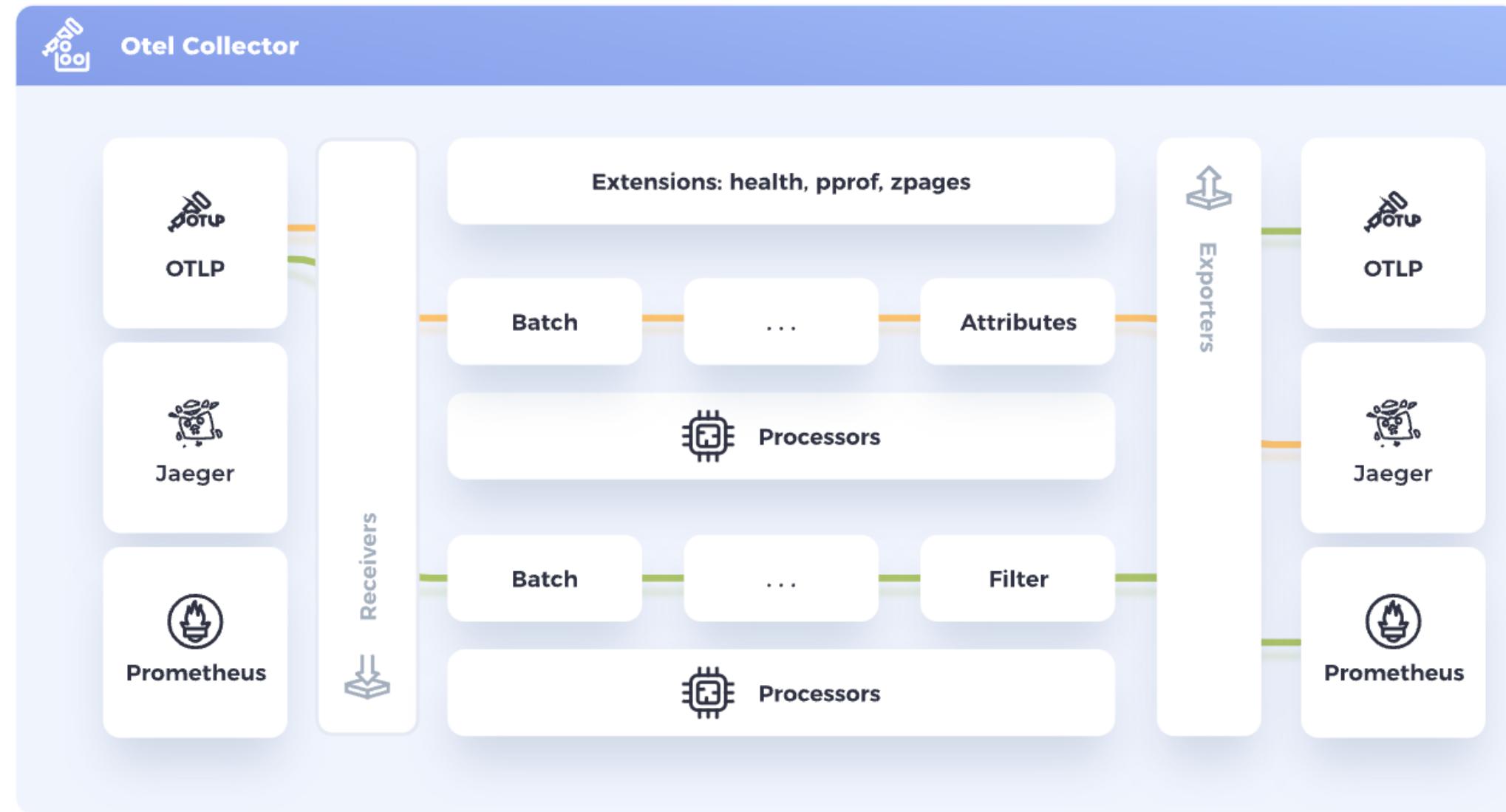
- Estandard
- Protocolo (OTLP)
- Context Propagation
- API & SDK & Tools
- Code Instrumentation ➡️
- Zero-Code Instrumentation

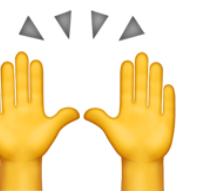


1. SDK Setup
2. Provider (traces, logs, metrics)
3. Exporter (traces, logs, metrics)
4. Generar Data (libs)

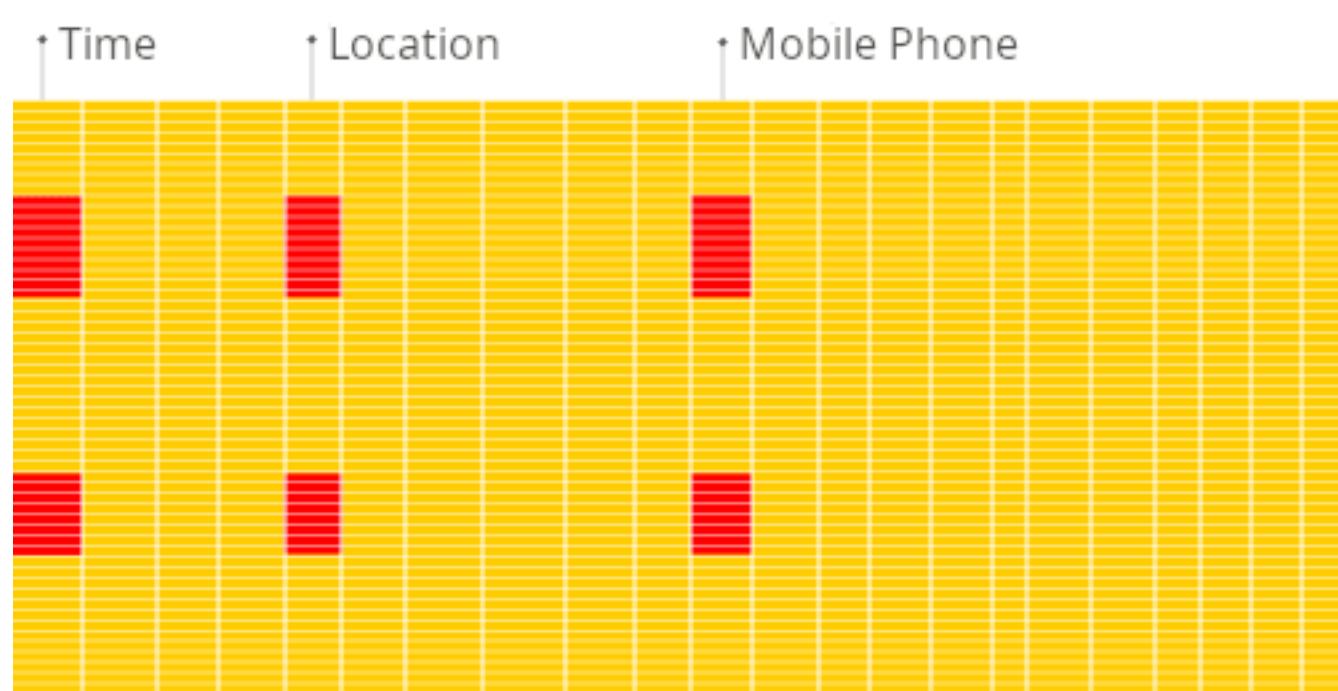
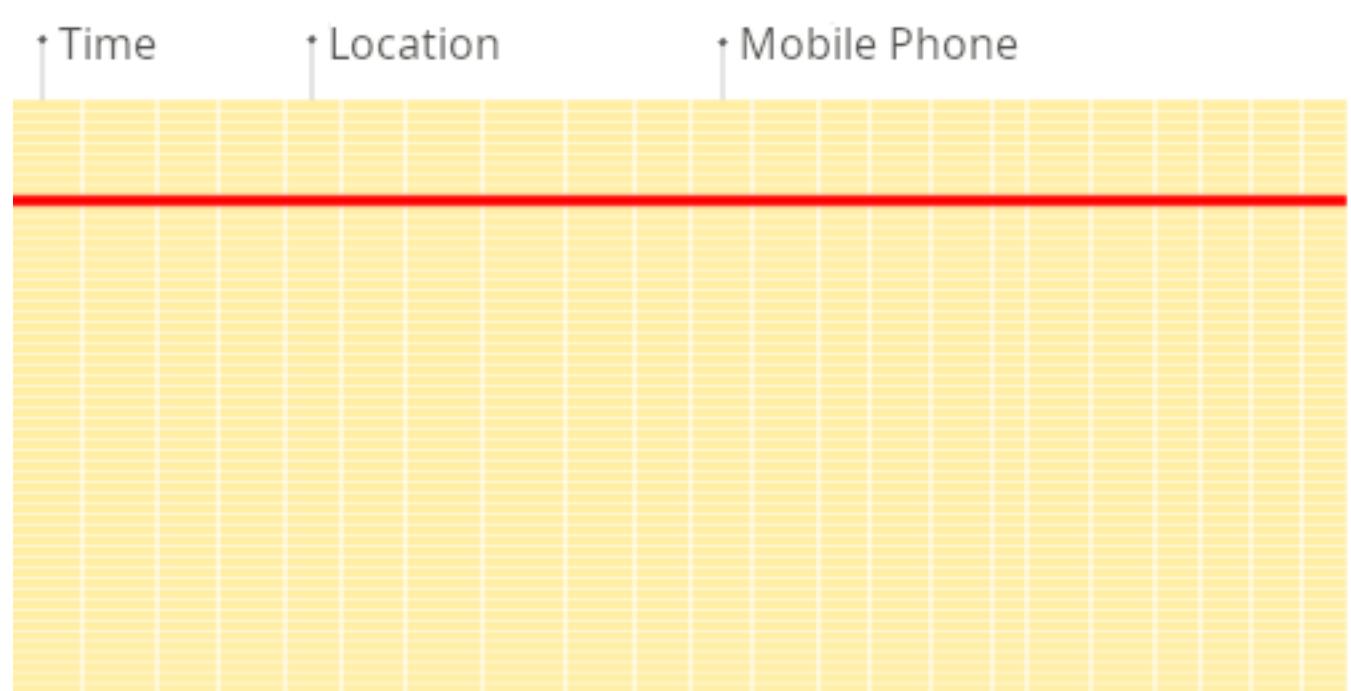


OTel Collector









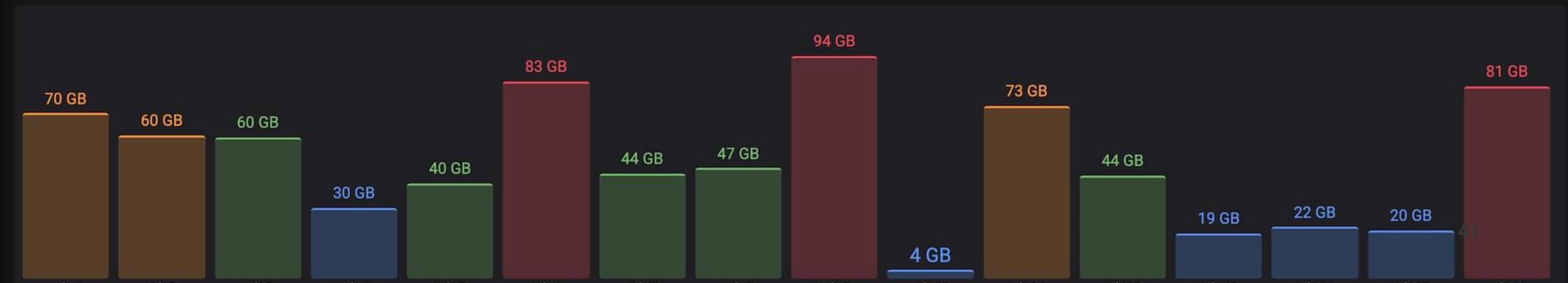
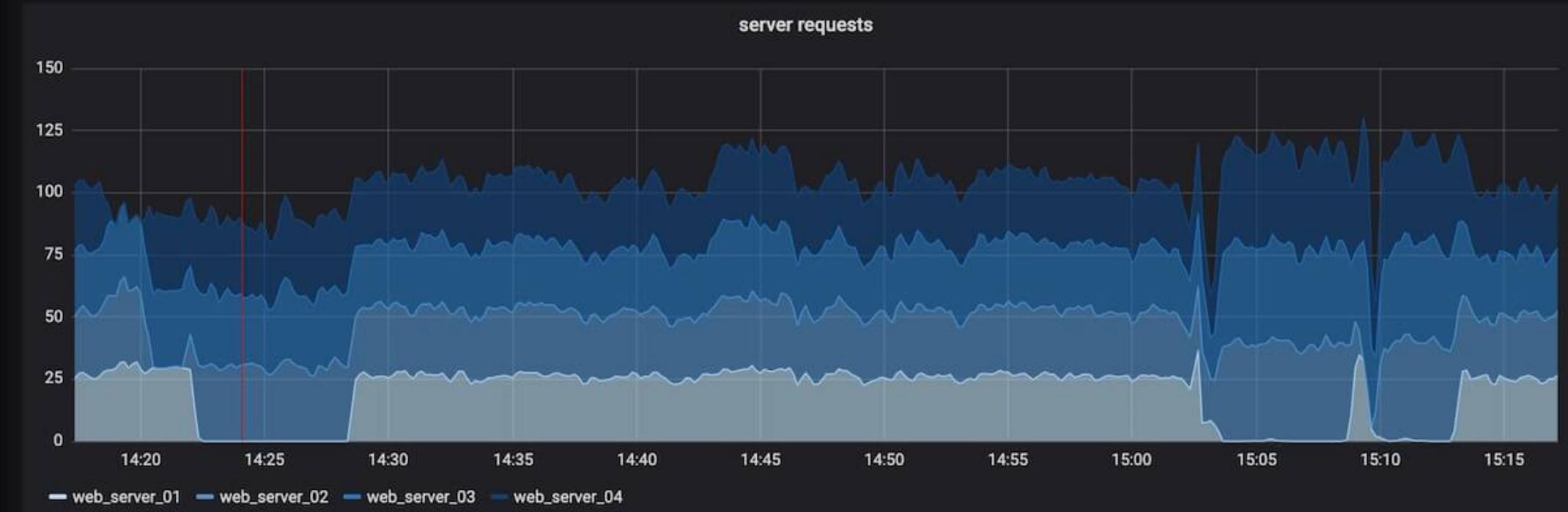
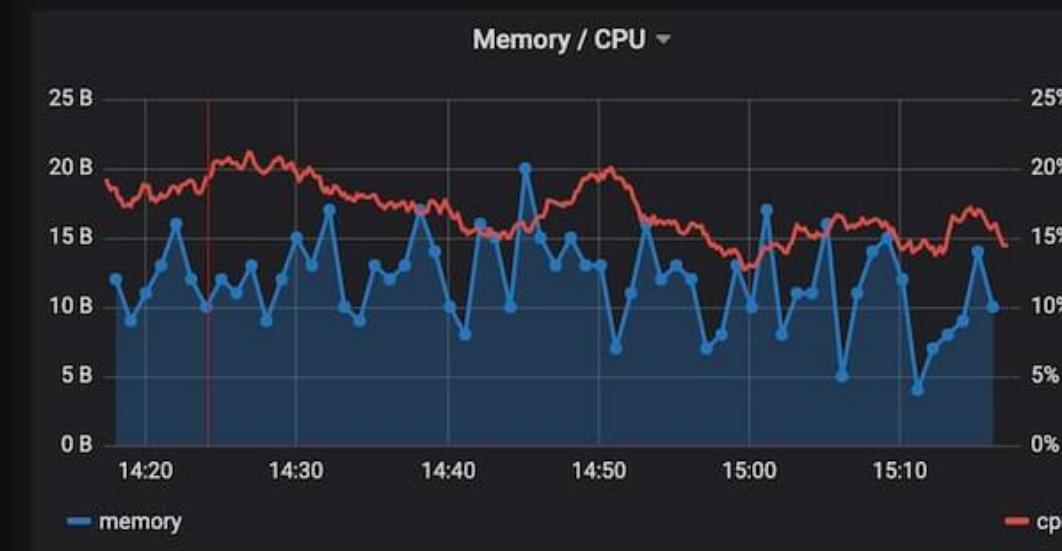
```
CREATE TABLE helloworld.my_first_table
(
    user_id UInt32,
    message String,
    timestamp DateTime,
    metric Float32
)
ENGINE = MergeTree()
PRIMARY KEY (user_id, timestamp)

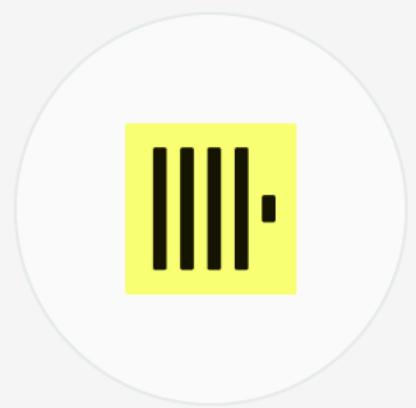
INSERT INTO helloworld.my_first_table (user_id, message, timestamp, metric) VALUES
(101, 'Hello, ClickHouse!', now(), -1.0 ),
(102, 'Insert a lot of rows per batch', yesterday(), 1.41421 ),
(102, 'Sort your data based on your commonly-used queries', today(), 2.718 ),
(101, 'Granules are the smallest chunks of data read', now() + 5, 3.14159 )

SELECT * FROM helloworld.my_first_table
```



Grafana





ClickHouse

[Overview](#)
[Installation](#)
[Change log](#)
[Related content](#)
[Create account](#)
[Version 4.5.0](#)


Dependencies

Grafana >=9.5.0

Developer

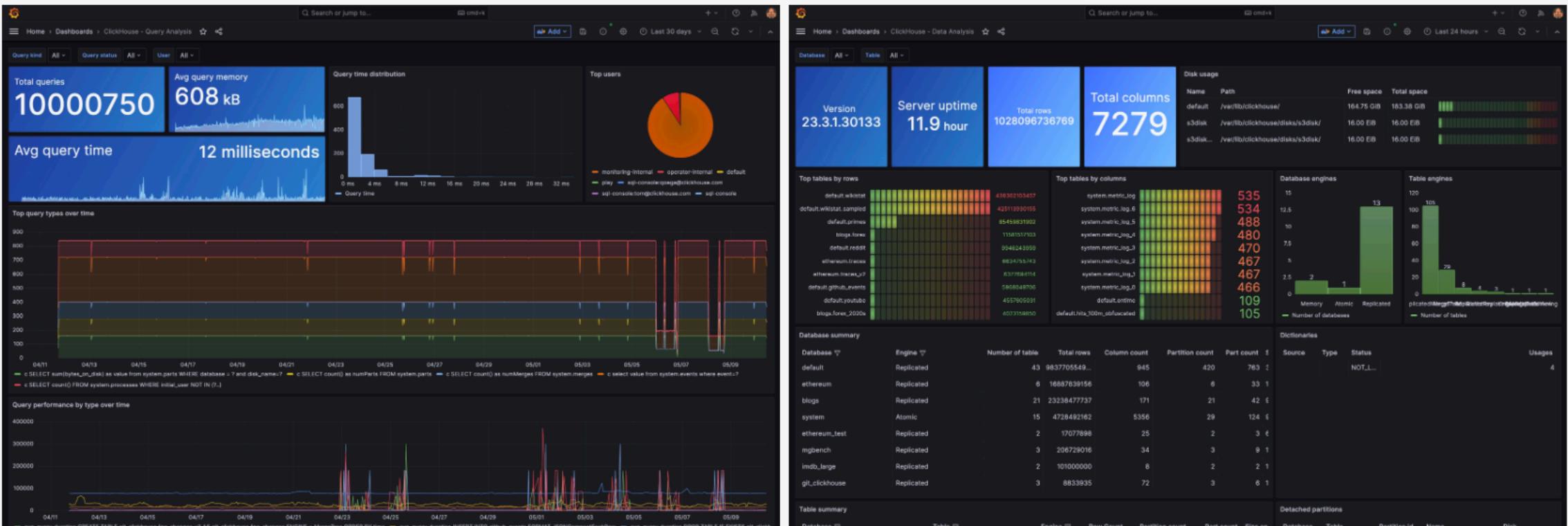
Grafana Labs

Resources

[Website](#)
[License](#)

Official ClickHouse data source for Grafana

The ClickHouse data source plugin allows you to query and visualize ClickHouse data in Grafana.



Version compatibility

Users on Grafana v9.x and higher of Grafana can use v4. Users on Grafana v8.x are encouraged to continue using v2.2.0 of the plugin.

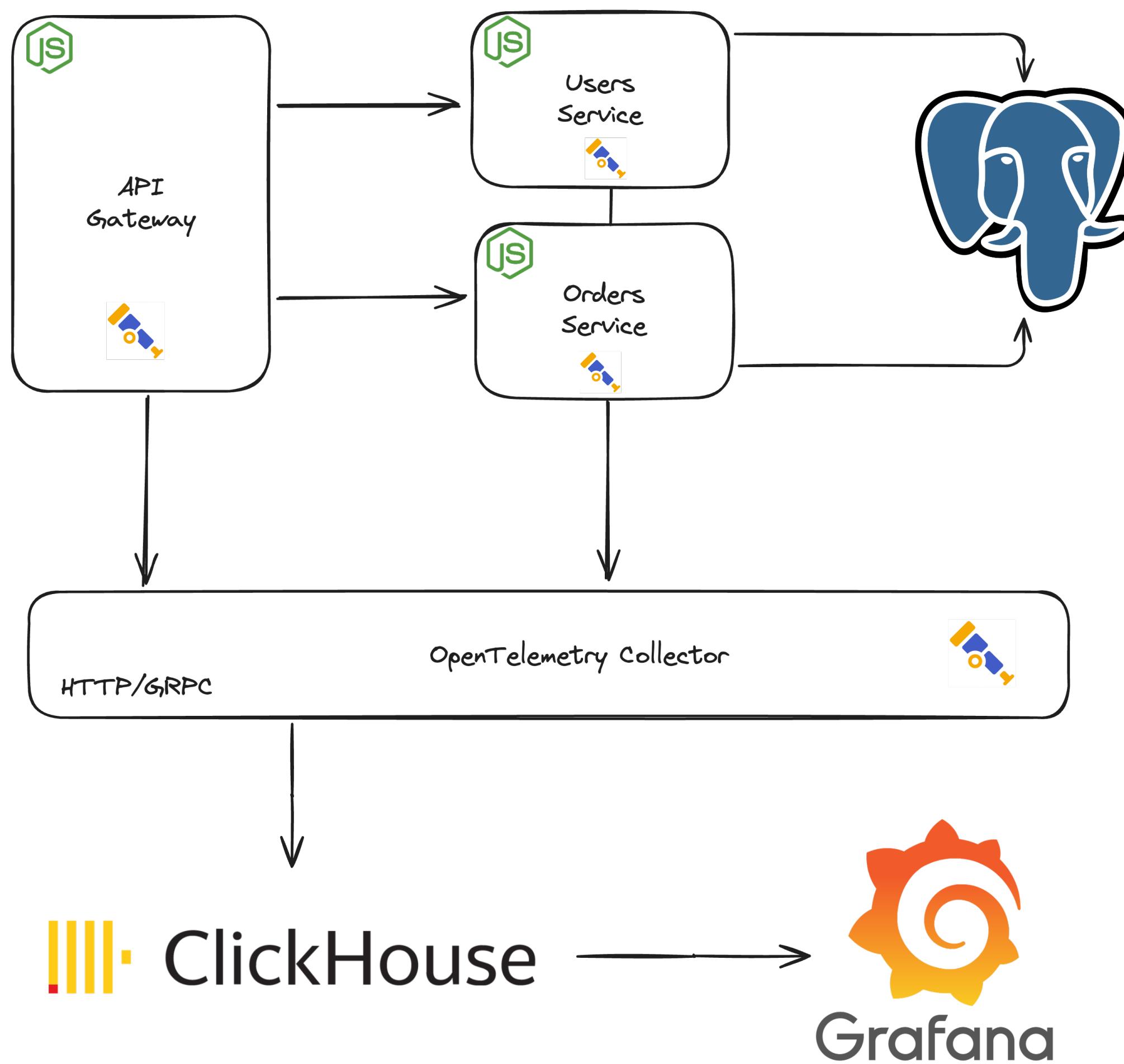


OpenTelemetry

ClickHouse

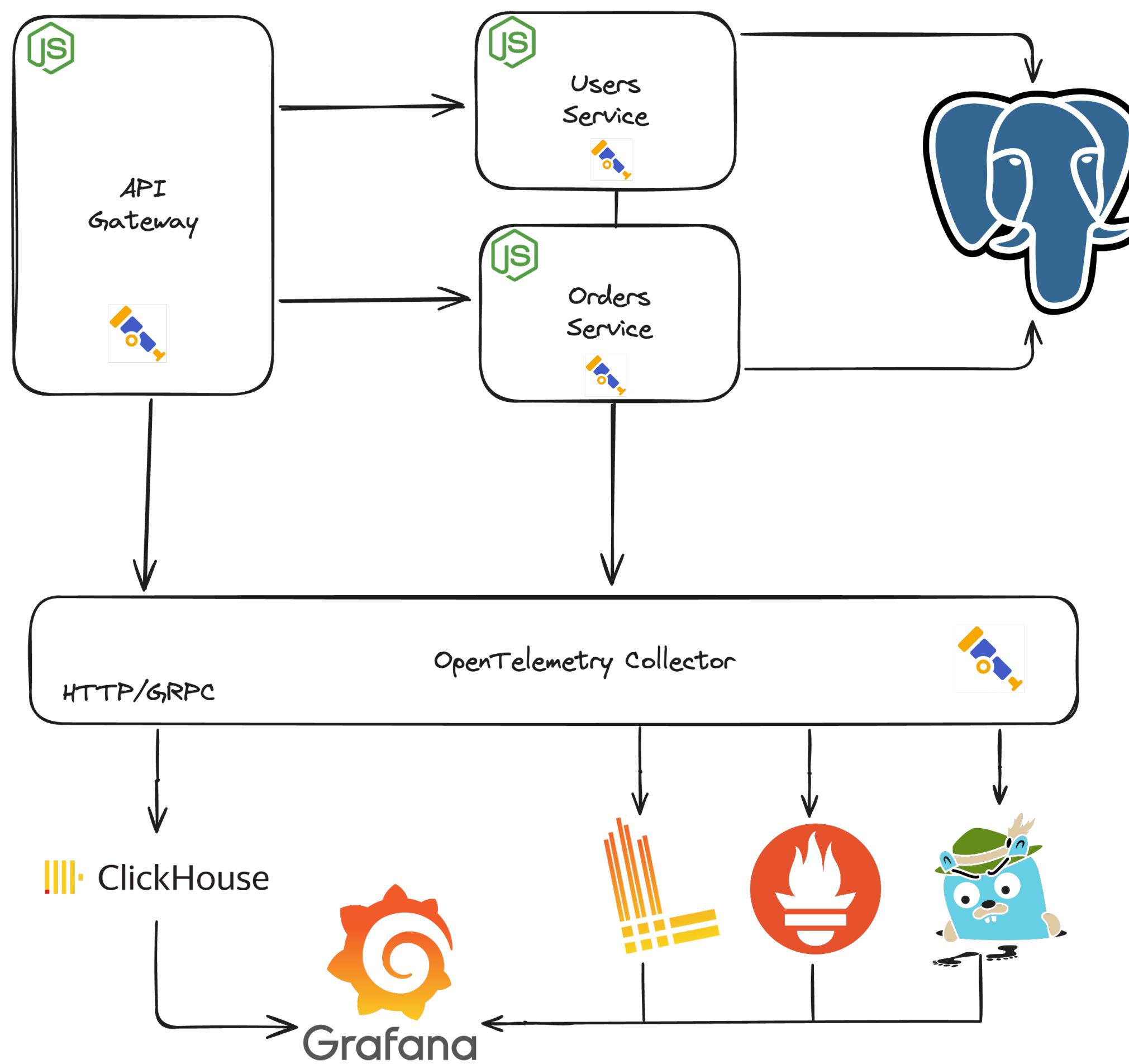


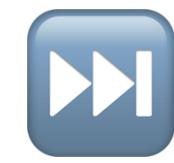
Grafana











opentelemetry-demo

O'REILLY®

Learning OpenTelemetry

Setting Up and Operating
a Modern Observability System



Ted Young & Austin Parker

O'REILLY®

Observability Engineering

Achieving Production Excellence



Charity Majors,
Liz Fong-Jones
& George Miranda⁴⁹



- Building a logging platform with Clickhouse and saving millions over DataDog
- Building an Observability Solution with ClickHouse
- How we used ClickHouse to store OpenTelemetry Traces and up our Observability Game





Gracias!

bit.ly/ianaya89-otel



@ianaya89