CLOUDFLARE®

Flows and BGP, a multi-Terabyte story

Louis Poinsignon - Cloudflare

Why monitoring

- Billing
    - Reducing costs

- Traffic engineering
    - Where should we peer?
    - Where should we set-up a new PoP?
    - Optimizing our network

- Anomaly detection
    - Troubleshooting
    - Proactive monitoring and predictions

Ways to monitor

- SNMP
- Flows
- BGP

What this presentation will be about

Flows

The previous pipeline

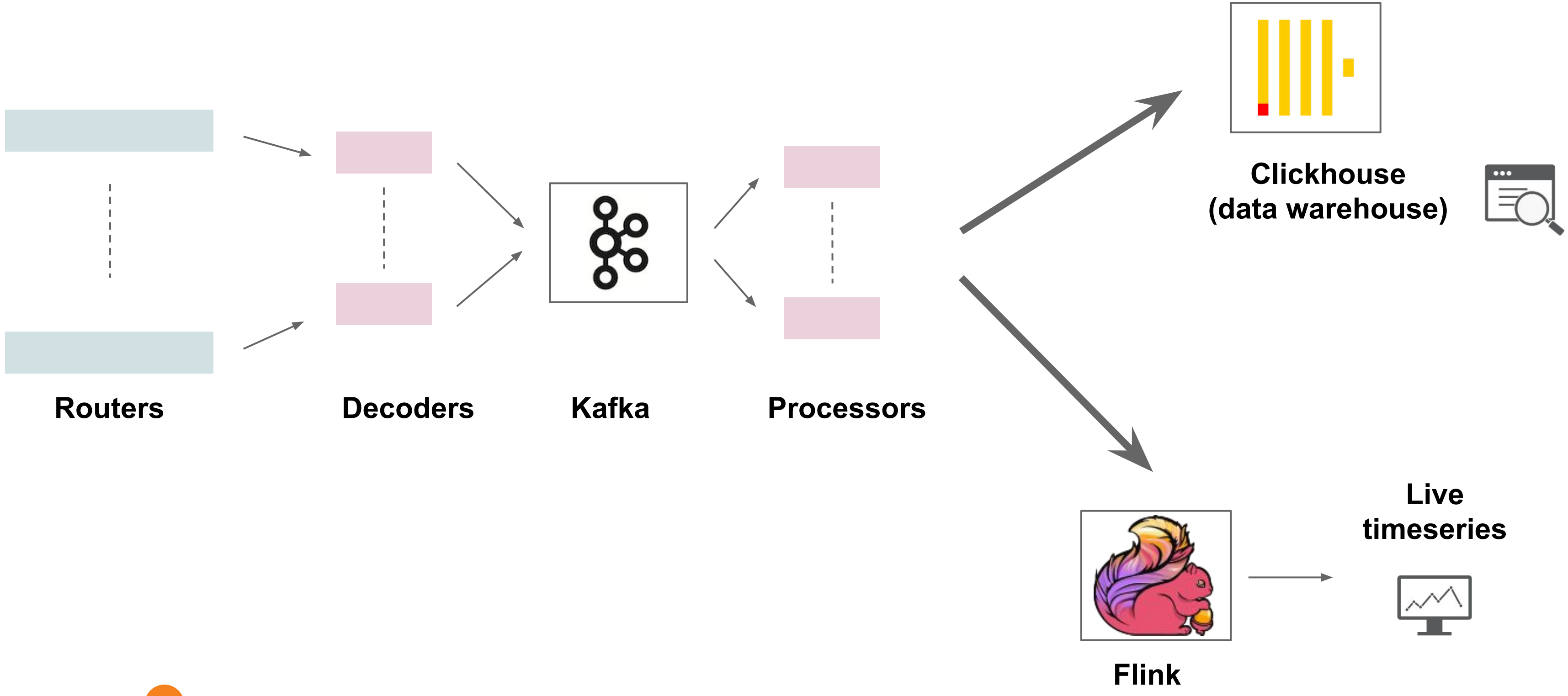
- Single machine
    - High CPU usage
    - Not easily accessible
- Dropping packets (around 10%)
- Not monitored
- sFlow cannot be aggregated with NetFlow
- Data not accessible to other teams / hard to develop on

- nfacctd + opentsdb

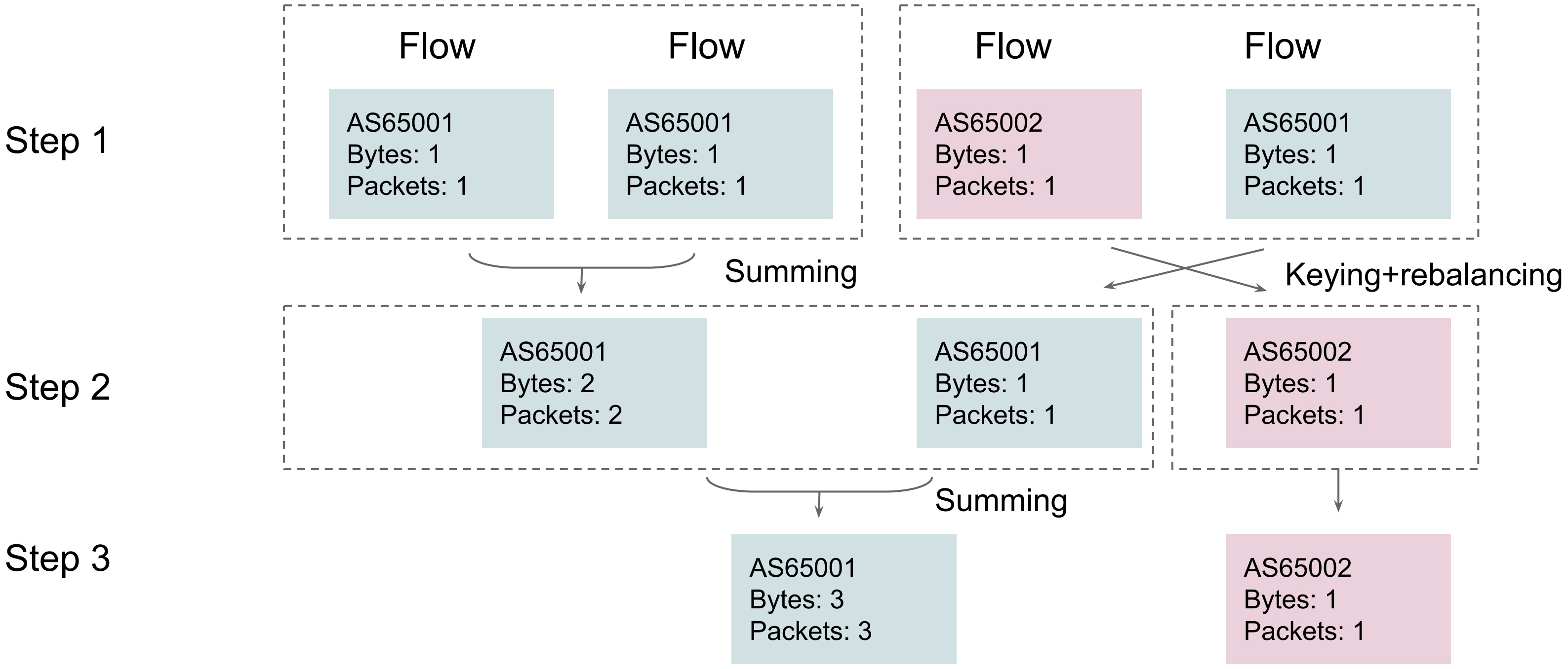
Time for an upgrade

- Let’s use the tools provided by the platform team
  - Load-balancers
  - Mesos
  - Containers
  - Databases
  - Messages brokers
  - Big data processing clusters

What we built



Flink - MapReduce



Step 1

Flow

AS65001
Bytes: 1
Packets: 1

Flow

AS65001
Bytes: 1
Packets: 1

Flow

AS65002
Bytes: 1
Packets: 1

Flow

AS65001
Bytes: 1
Packets: 1

Summing

Keying+rebalancing

Step 2

AS65001
Bytes: 2
Packets: 2

AS65001
Bytes: 1
Packets: 1

AS65002
Bytes: 1
Packets: 1

Summing

Step 3

AS65001
Bytes: 3
Packets: 3

AS65002
Bytes: 1
Packets: 1

Flink - Sample program

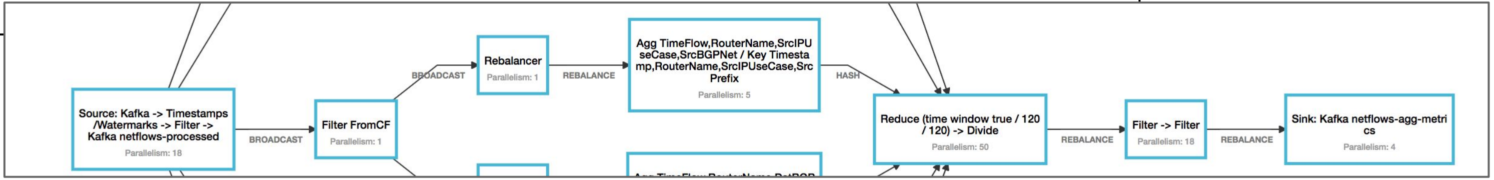
```
DataStream<FlowMessage> inData =
new FlinkKafkaConsumer09<FlowMessage>(
                "netflows-processed",
                new FlowMessageDeserializer(),
                propertiesConsumer);

DataStream<FlowMessage> inDataEyeball =
                inData.filter(new FlowFilter.EyeballFilter()).
                setParallelism(1).broadcast();

DataStream<FlowAggMessage> inDataAgg =
                inDataEyeball.map(new FlowUtils.Mapper("DstAS,colo"));

inDataAgg.reduce(new FlowTransformations.FlowAggReduceKey());
```

Source (Kafka)

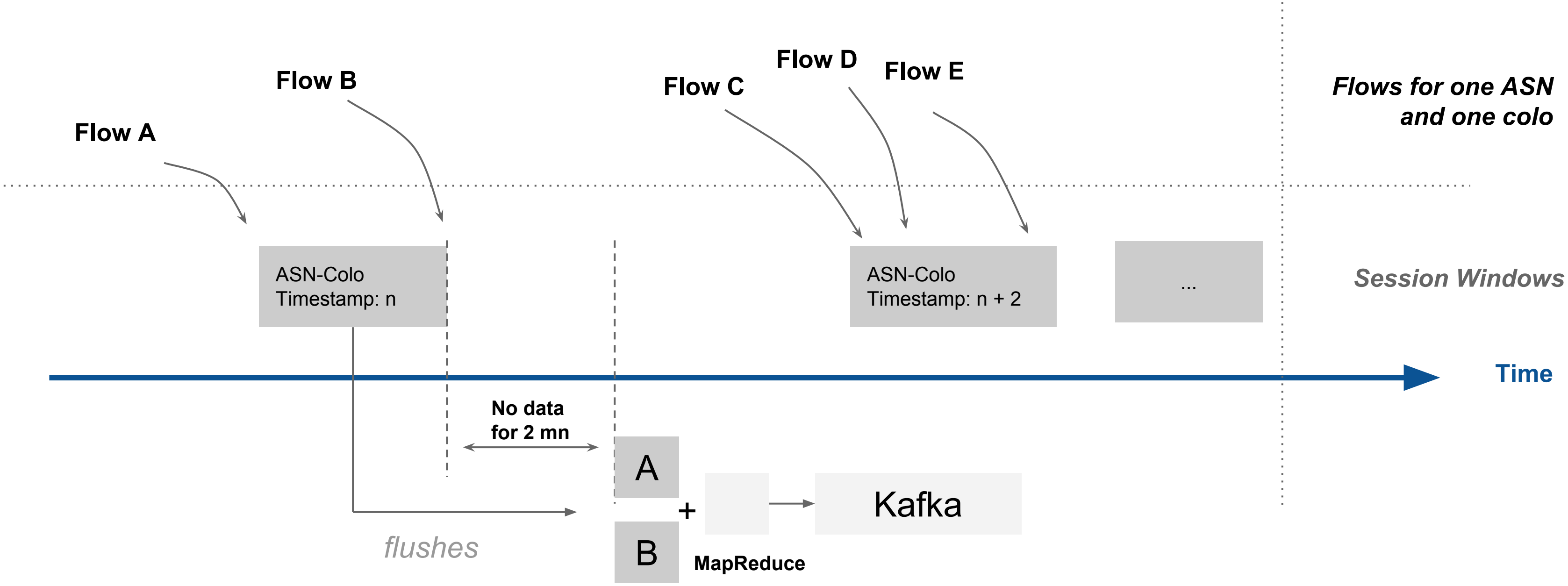
Filter

Mapping

Reduce

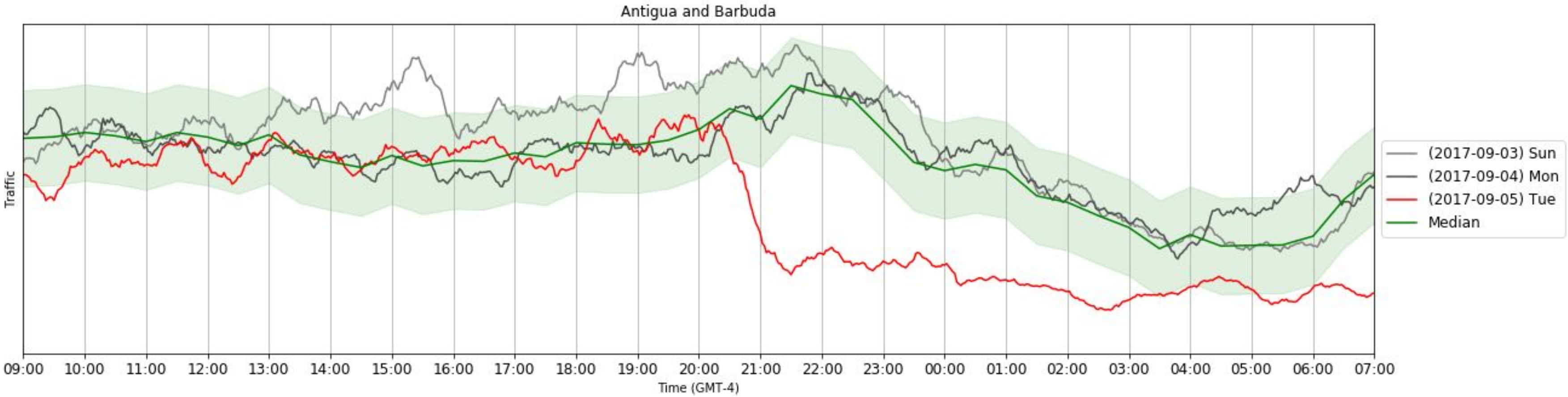

Source: Kafka -> Timestamps /Watermarks -> Filter -> Kafka netflows-processed
Parallelism: 18
BROADCAST
Filter FromCF
Parallelism: 1
BROADCAST
Rebalancer
Parallelism: 1
REBALANCE
Agg TimeFlow,RouterName,SrcIPUseCase,SrcBGPNet / Key Timestamp,RouterName,SrcIPUseCase,SrcPrefix
Parallelism: 5
HASH
Reduce (time window true / 120 / 120) -> Divide
Parallelism: 50
REBALANCE
Filter -> Filter
Parallelism: 18
REBALANCE
Sink: Kafka netflows-agg-metrics
Parallelism: 4

Flink - Windowing



Results - Aggregations

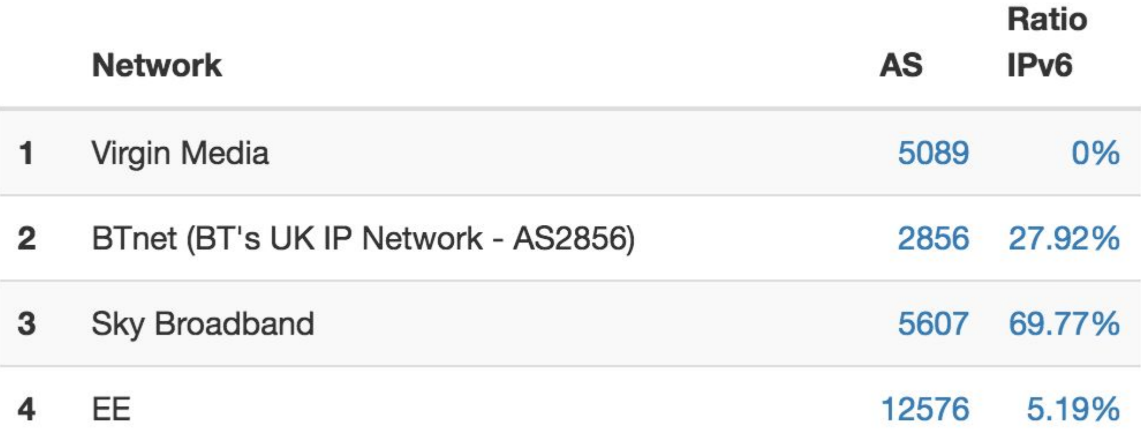
Traffic generated by all the datacenters, by country, by network, etc.



https://blog.cloudflare.com/the-story-of-two-outages/

Results - Statistics

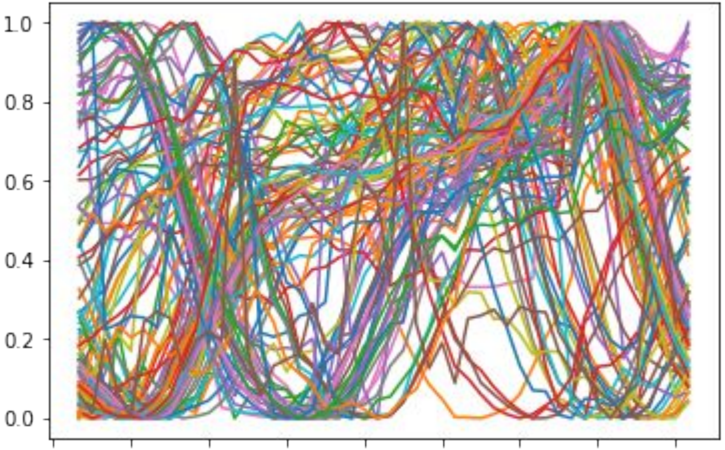
Top providers per country, IPv6 penetration, etc.



			Ratio
	Network	**AS**	**IPv6**
1	Virgin Media	5089	0%
2	BTnet (BT's UK IP Network - AS2856)	2856	27.92%
3	Sky Broadband	5607	69.77%
4	EE	12576	5.19%

Results - Example: maintenance

Building a list with the best hours for maintenance



Normalized traffic of our PoP over a day

Algorithms

Derivation

Correlation

  Pearson coefficient

    Quantify the difference

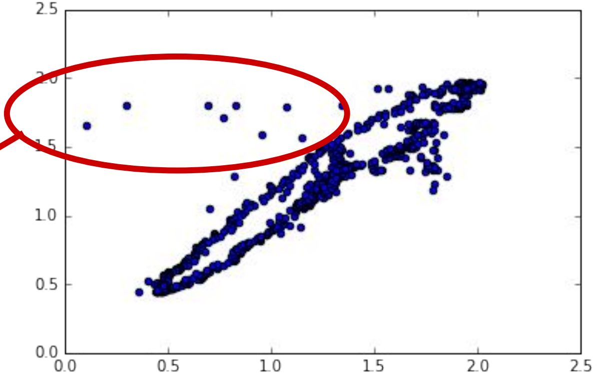
Median

  Remove small variations

Variance

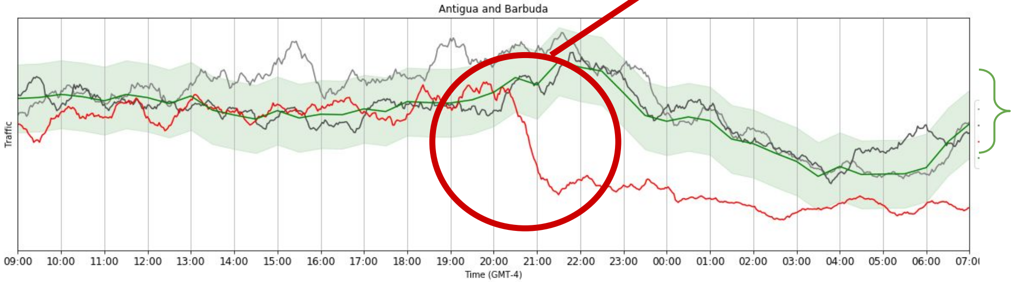
  Variation intensity

Correlation coefficient



Outliers

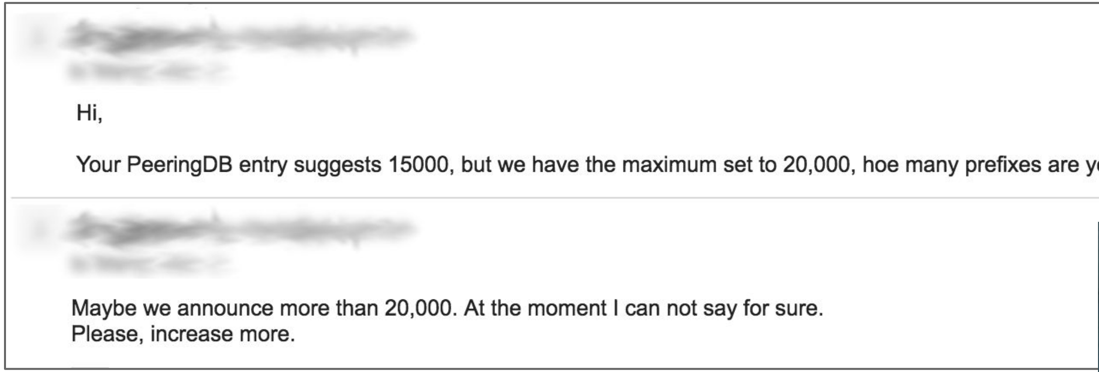
Antigua and Barbuda



Local variance following median

BGP

BGP flaps



[redacted]

Hi,

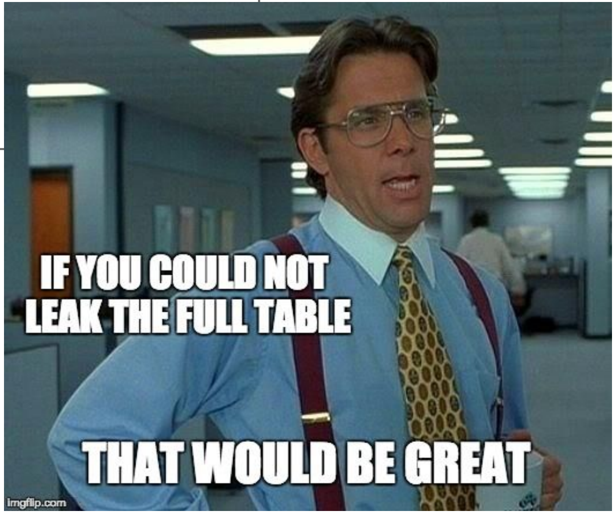
Your PeeringDB entry suggests 15000, but we have the maximum set to 20,000, hoe many prefixes are you sending?

[redacted]

Maybe we announce more than 20,000. At the moment I can not say for sure.
Please, increase more.

I can not say for sure.

Please, increase more.



BGP

100+ routers with full tables

Want to be aware of route-leaks or other anomalies

Need to do data analysis and periodic reports

Provide improved data for the flow pipeline

- Mappings prefixes to ASN
- Next-hop to ASN
- Peering/transit information

BGP Pipeline

Completely custom BGP implementation to live on a cloud with elastic IPs.

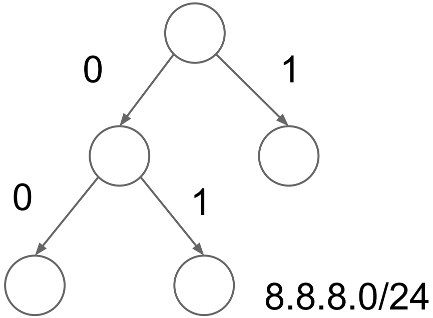
- More flexible than GoBGP

Horizontal scaling and hashing

- A single machine contains a subset of the data.
- Messages from x will always reach y

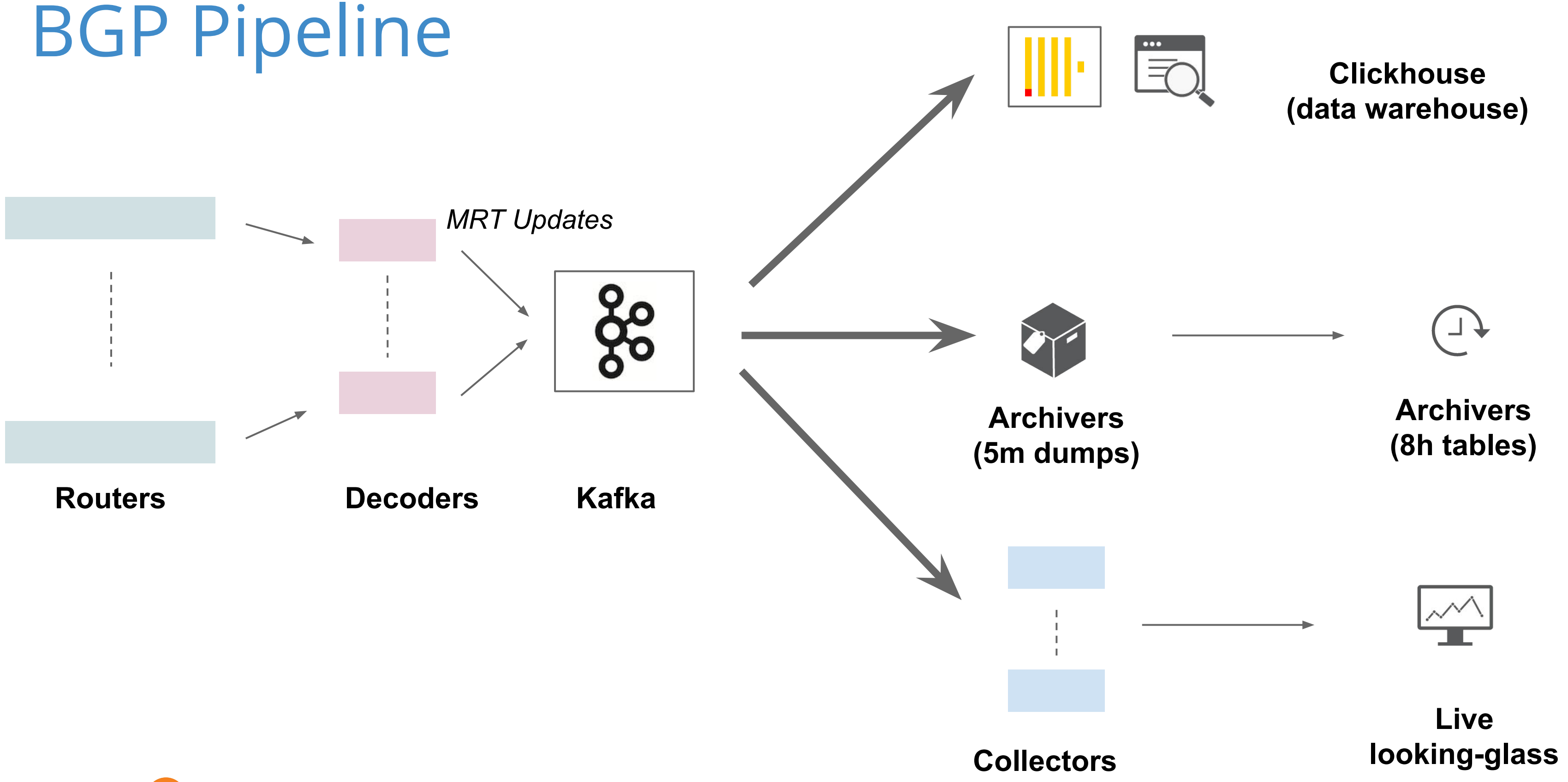
Develop custom trie implementation

- Stores IP prefixes in an optimized way



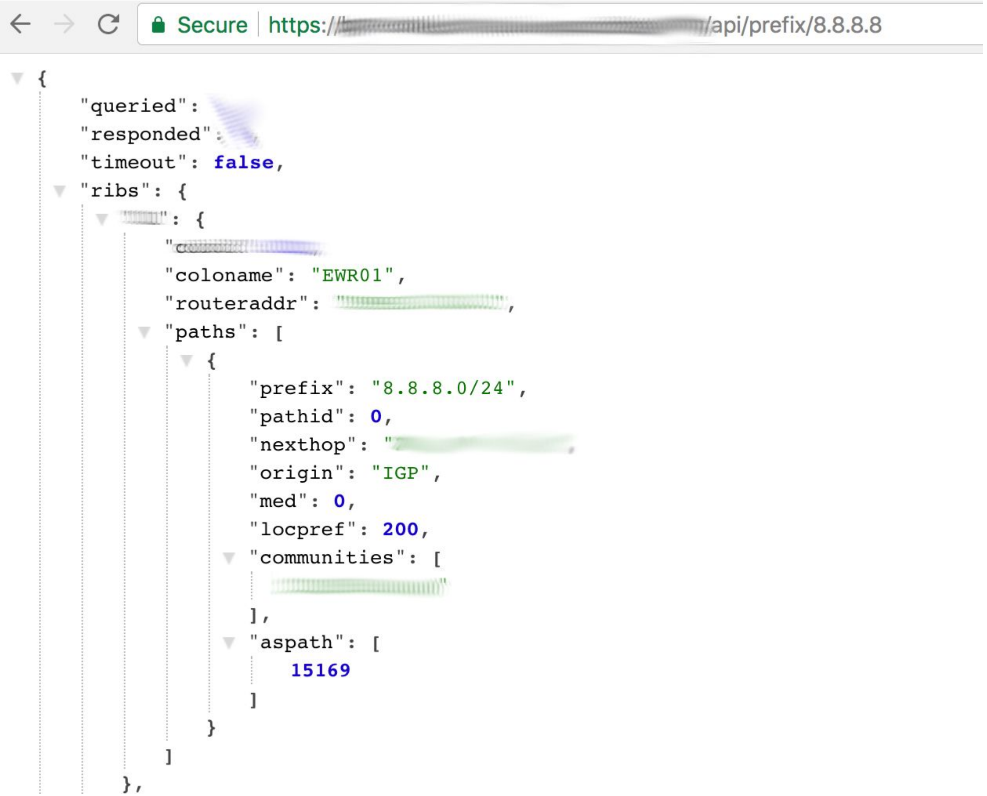
CLOUDFLARE

BGP Pipeline



CLOUDFLARE

BGP API



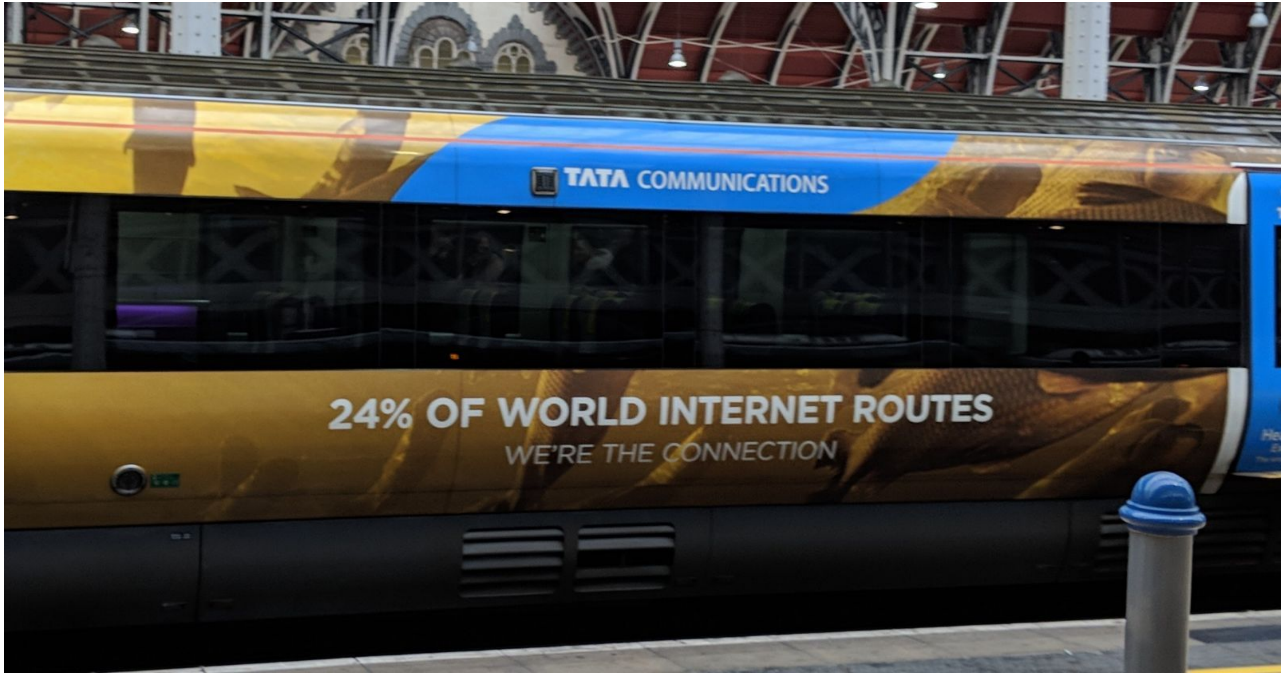
Secure | https://[redacted]/api/prefix/8.8.8.8

```
{
    "queried": [redacted]
    "responded": [redacted]
    "timeout": false,
    "ribs": {
        "[redacted]": {
            "[redacted]
            "coloname": "EWR01",
            "routeraddr": [redacted],
            "paths": [
                {
                    "prefix": "8.8.8.0/24",
                    "pathid": 0,
                    "nexthop": "[redacted],
                    "origin": "IGP",
                    "med": 0,
                    "locpref": 200,
                    "communities": [
                        [redacted]"
                    ],
                    "aspath": [
                        15169
                    ]
                }
            ]
        },
```

Discoveries

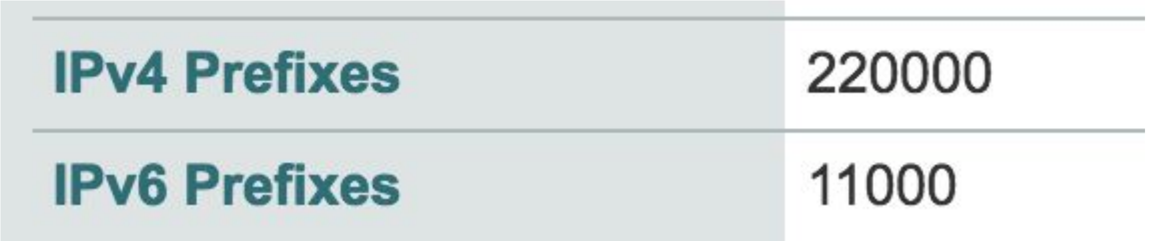
- People sending us IX LAN prefixes
- Receiving smaller than /48 IPv6 and smaller than /24 IPv4
- Longest AS-Path
    - 2402:8100:3980::/42 → 37 ASNs

A train



“We have approximately 170 000 routes”

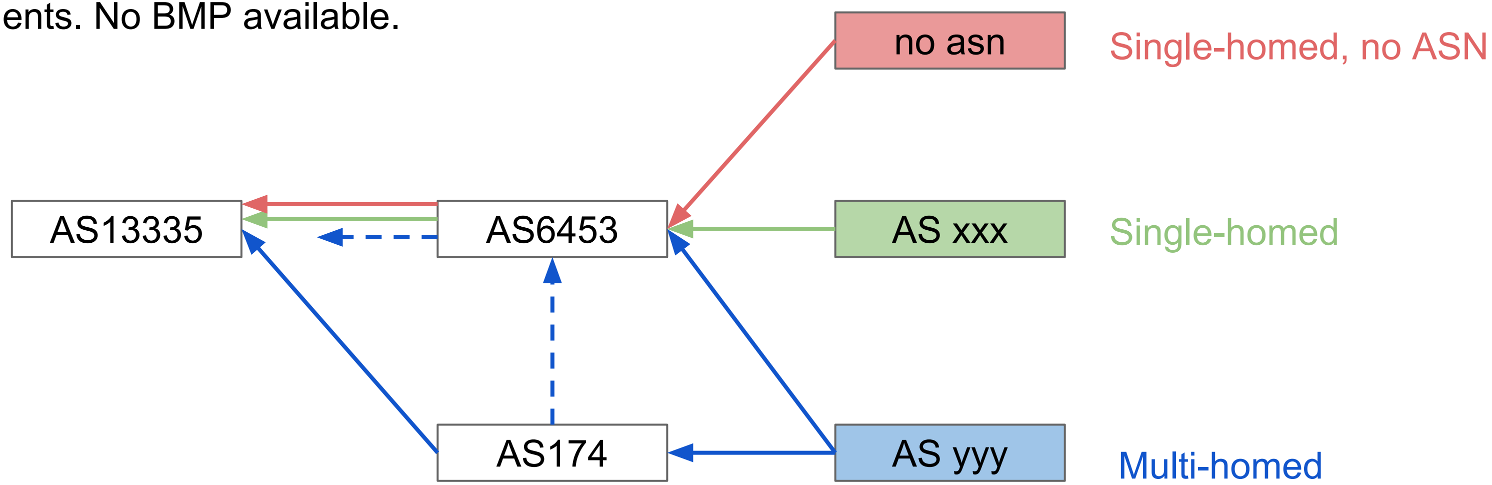
TATA’s Peeringdb



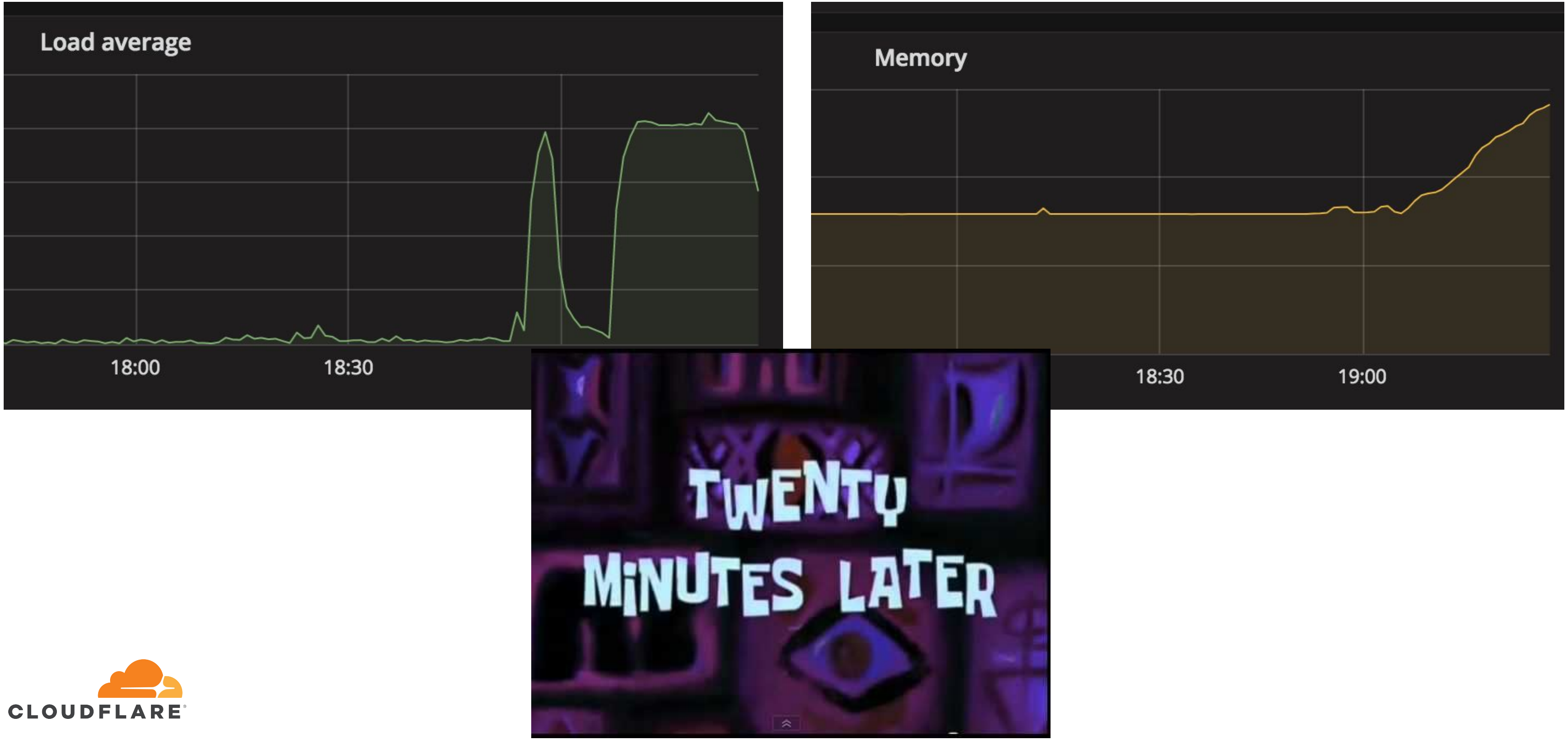
IPv4 Prefixes	220000
IPv6 Prefixes	11000

Transit

3 types of clients. No BMP available.



Let’s process



Results

315997 routes (approx 42%)

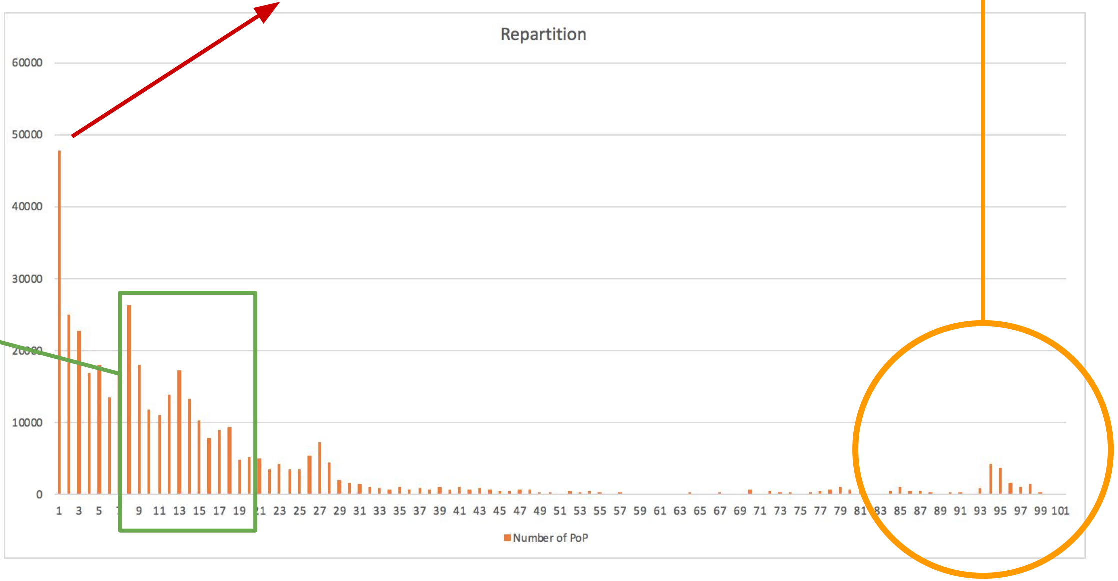
- with 6453 as the last ASN with no transit listed behind

205 routes

- With 6453 as the last ASN

Around 100k routes per PoP

- With 6453 as ASN

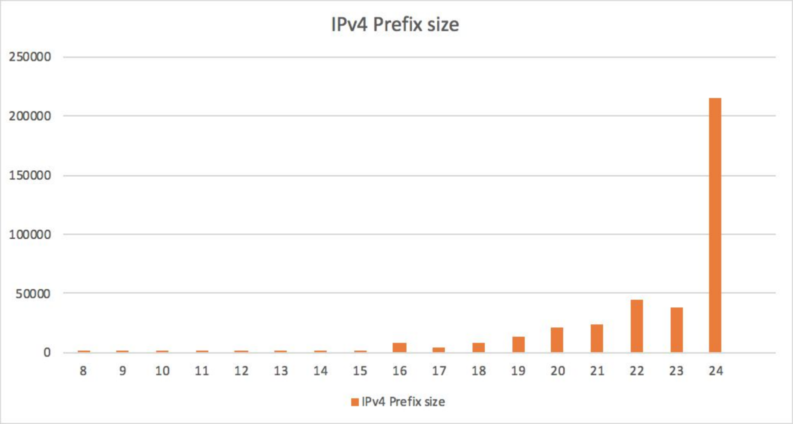


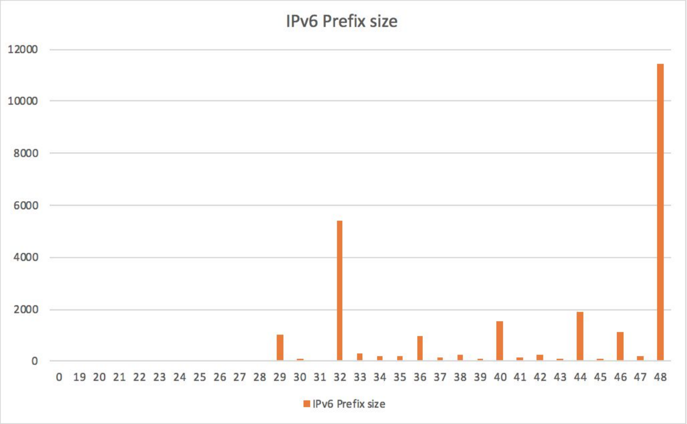
Only as last resort

Single-homed

Around 24%

IP prefix sizes





Size, costs

Size

Flows

- Raw feed: 10-50 mbps
- Unique storage: 1-5 Terabytes
- Aggregated storage: 10-100 GigaBytes

BGP

- Raw feed: few kbps (100s updates per second)
- Around 35 GB of RAM used for storing the tables
- Around 100 MB per full table stored in MRT format

Costs

The price of running it on a cloud (Amazon/Google/Azure) will depend on your size.

Rule of thumb:

traffic	10Gb/s	100Gb/s	1Tb/s
routers	10	50	100
Flows	$200/mo	$800/mo	$1,500/mo
BGP	$200/mo	$600/mo	$800/mo

Open-source

Open-source (soon)

Flows:

- Decoders
- ~~Processors~~ → too specific to Cloudflare
- Flink example

BGP:

- Library to manage a session
- ~~Server, APIs~~ → too specific to Cloudflare

Tools:

- Maxmind DB encoder

Thank you!
Questions?