



elastic

Elastic Stack Data Administration

An Elastic Training Course

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

training.elastic.co

Course: Elastic Stack Data Administration

Version 6.0.1

© 2015-2017 Elasticsearch BV. All rights reserved. Decompiling, copying, publishing and/or distribution without written consent of Elasticsearch BV is strictly prohibited.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Lab Prep Checklist

- Download the PDF from the links provided in email
- There is a .zip file also in the email links, but this file will not be used during the class
 - However, it can be used to practice on your own machine after the class
- In this course, you will use a pre-built environment with all the datasets and software ready to go

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Agenda and Introductions

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Course Agenda

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Services Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Gabriel Montecinos
20-Nov-2017 - Kapsch TrafficCom

Introductions

- Name
- Company
- What do you do?
- What are you using Elasticsearch for?
- What do you hope to get out of this training?

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logistics

- Facilities
- Emergency Exits
- Restrooms
- Breaks/Lunch

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 1

Elastic Stack Data Administration Concepts

Gabriel Montoya - 20-Nov-2017 - KapsusTrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- The Story of Elastic Stack
- Elastic Stack Components
- Understanding Event Data
- Getting Started with Beats
- Heartbeat
- Kibana

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

The Story of Elastic Stack

Gabriel Montoya - 20-Nov-2017 - KapschTronicCom



First version of **Elasticsearch** (0.4) released

2010

First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

2010 2012

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

Kibana and Logstash OS projects join

2010 2012 2013

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

Kibana and **Logstash** OS projects join

Elasticsearch 1.0 GA

2010 2012 2013 2014

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

Kibana and **Logstash** OS projects join

Elasticsearch 1.0 GA

Elastic Cloud acquired

Beats OS joins

Elasticsearch 2.0 GA

2010 2012 2013 2014 2015

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

Kibana and **Logstash** OS projects join

Elasticsearch 1.0 GA

Elastic Cloud acquired

Beats OS joins

Elasticsearch 2.0 GA

Elastic Stack

Machine Learning acquired

5.0 GA

2010

2012

2013

2014

2015

2016

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



First version of **Elasticsearch** (0.4) released

Elasticsearch becomes a company

Kibana and **Logstash** OS projects join

Elasticsearch 1.0 GA

Elastic Cloud acquired

Beats OS joins

Elasticsearch 2.0 GA

Elastic Stack

Machine Learning acquired

5.0 GA

APM acquired

6.0 GA

2010

2012

2013

2014

2015

2016

2017

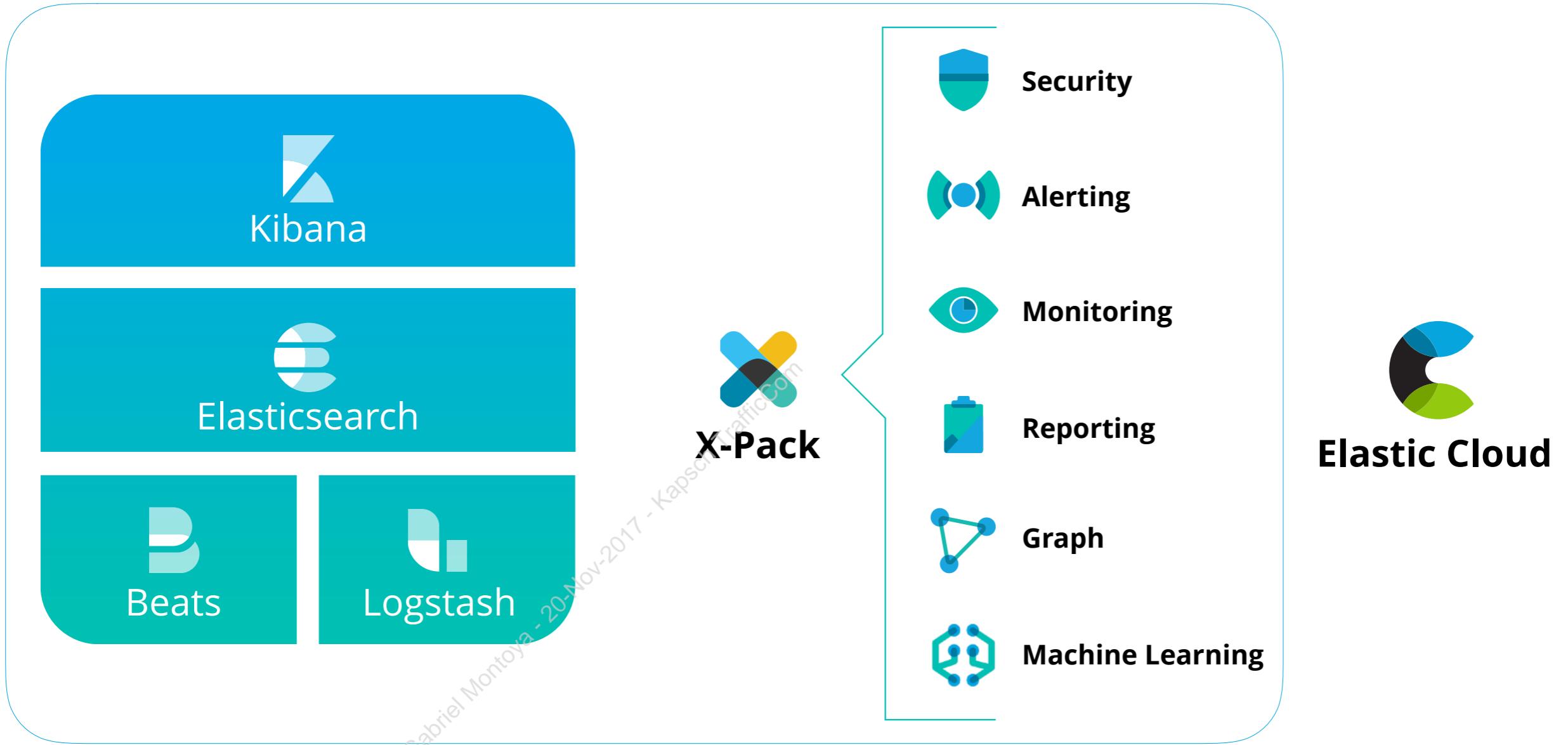
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Elastic Stack Components

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



The Elastic Stack



Elastic Stack



Beats



Logstash



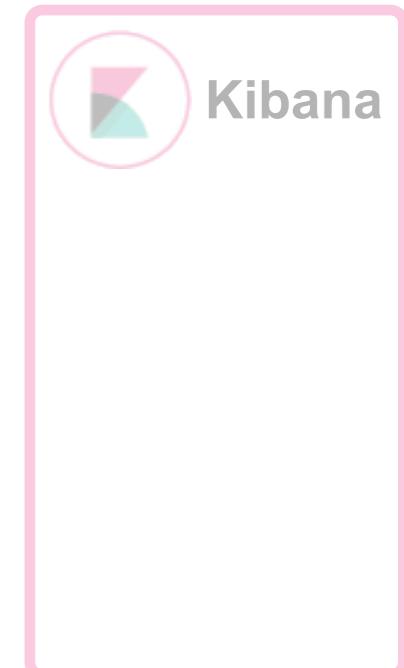
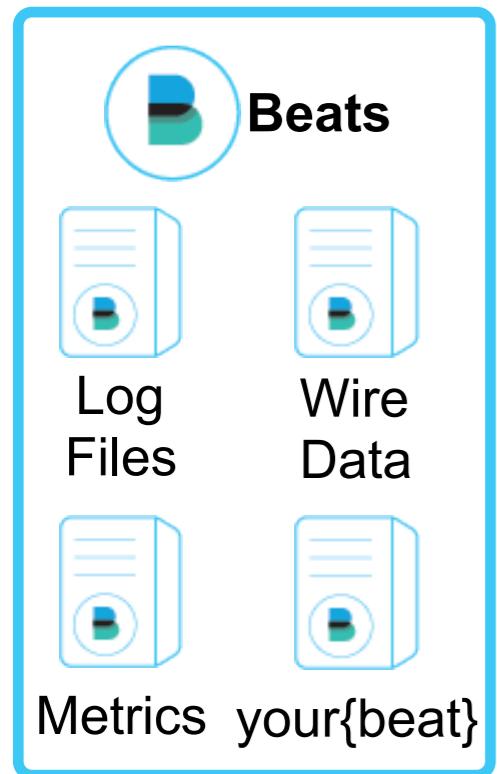
Elasticsearch



Kibana

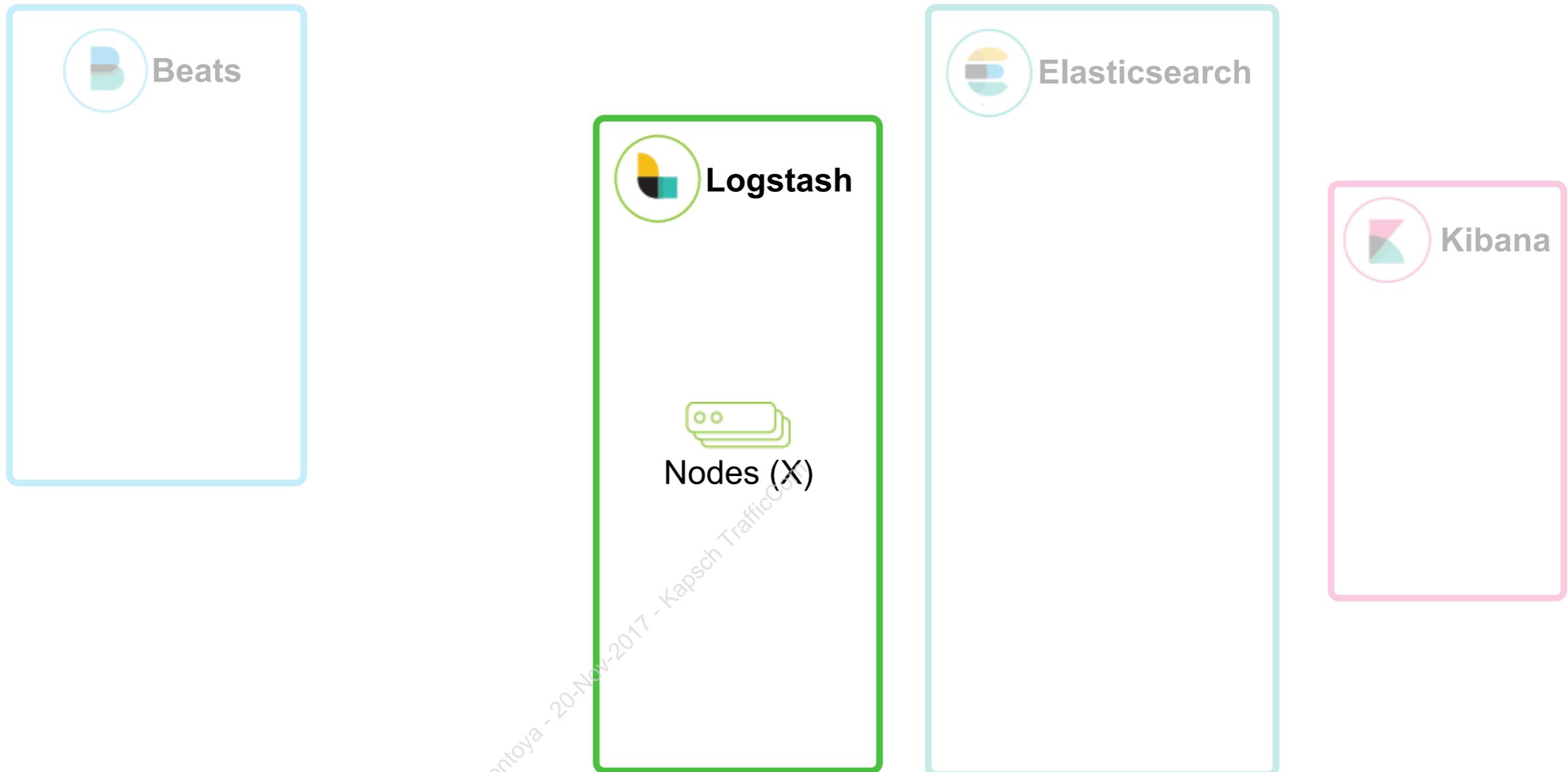
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats



- ▶ Data Shipper
- ▶ Light Weight
- ▶ Application Side Component

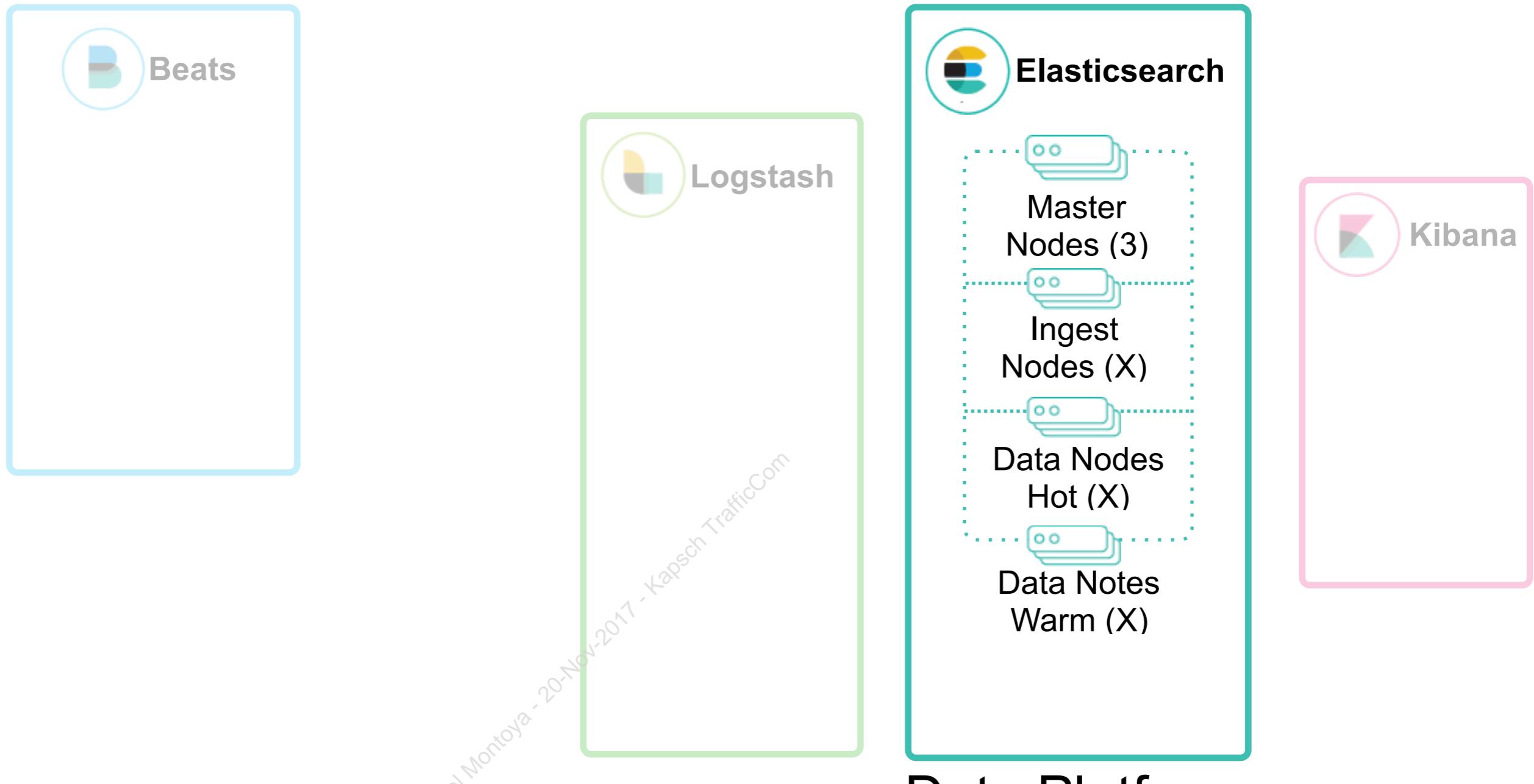
Logstash



- ▶ Data Collector/Processor
- ▶ Heavy Weight
- ▶ Server Side Component

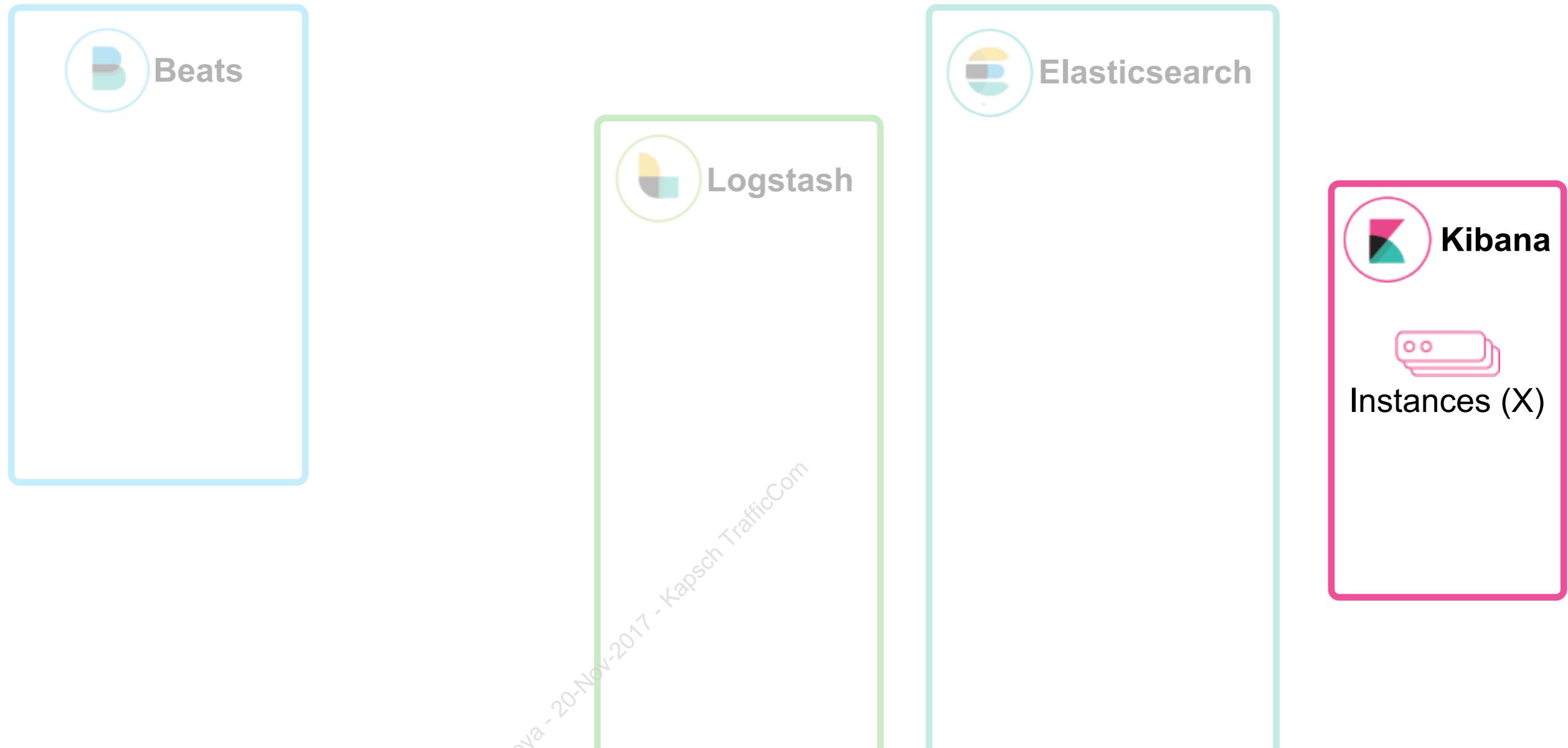


Elasticsearch



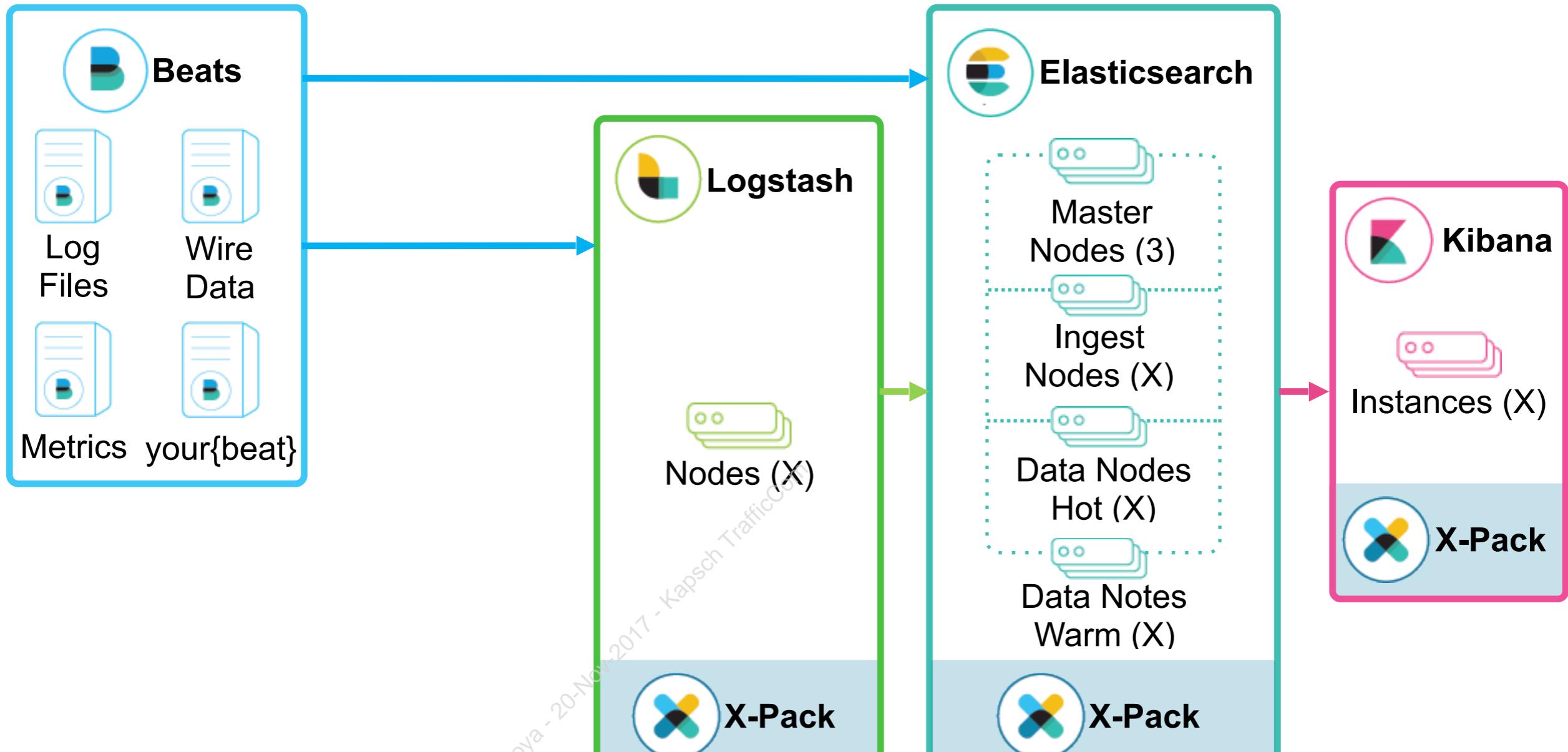
- ▶ Data Platform
- ▶ Really Fast
- ▶ HTTP + JSON

Kibana

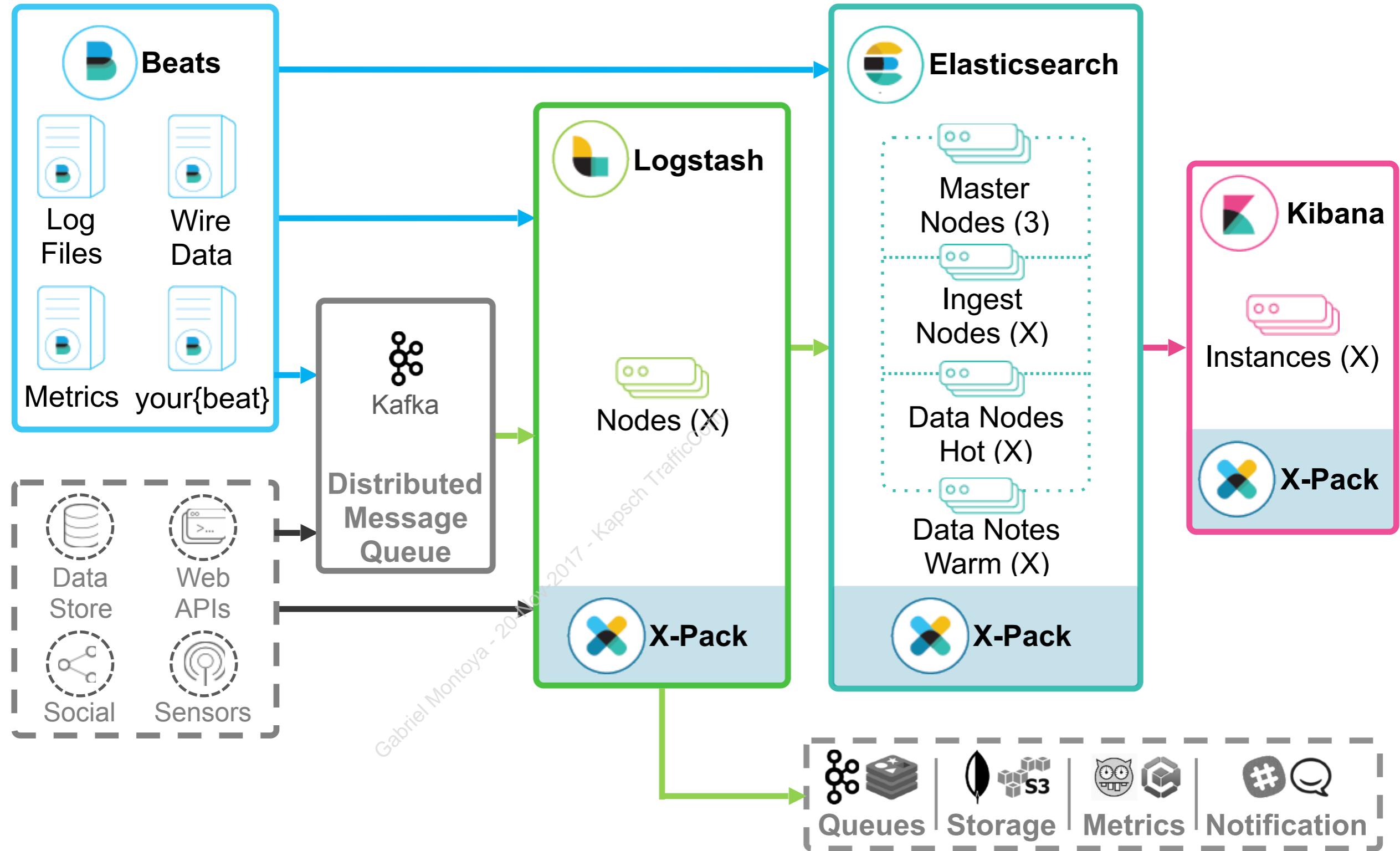


- ▶ Data Visualization
- ▶ Stack Configuration
- ▶ Elastic Stack UI

Elastic Stack



The Elastic Journey of an Event



Understanding Event Data

Gabriel Montoya - 20-Nov-2017 - Kapsch TrainCom

Understanding Event Data

- Data can be divided into two main categories:
 - **fixed-size data**
 - predictable (small or large)
 - easier
 - **event-based data**
 - unpredictable
 - harder
- Even though you can use the Elastic Stack to analyze fixed-size data, this course focuses on event-based data

Gabriel Montoya / 2017-Nov-2017 - Kapsch TrafficCom



Event

- "*An action or occurrence recognized by software, often originating asynchronously from the external environment, that may be handled by the software.*" (Wikipedia)
- Any piece of information that has a **timestamp** and **data**:
 - log line
 - tweet
 - CPU
 - temperature
 - ...

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Event Lifecycle

1. **Genesis:** an event occurs
2. **Record:** device or application creates a record of that event
3. **Transport & Store:** move logs to alternate storage (archival, indexing, etc)
4. **Search & Analyze:** search for events, analyze data for trends and anomalies
5. **Archive:** long-term storage of events (often for regulatory reasons)
6. **Purge:** delete old data which is no longer required

Gabriel Montoya - 2017 - Kapsch TrafficCom

Event-Based Data

- Events are represented as ***documents***
 - Typically events represent a real-world event that occurred
 - Cannot be repeated (at least not the exact same event)
 - Once occurred, it cannot be modified (it happened in the past and we don't support time travel.... yet)
- Each event (document) is associated with a ***timestamp***
- Documents keep on "flowing" in
 - Not only do we not support time travel, we cannot even stop time from ticking

Gabriel Montoya - 20-Nov-2010 Kapsch TrafficCom



Event-Based Data

- The nature of time-based data
 - Scales with time - always increasing
 - Impossible to predict the scale of future events
- Dictates the nature of how we use time-based data
 - We cannot keep all events for all time (not enough room in the world)
 - We mostly care about recent events compared to old events
 - Old needs to be defined as it is relative to the problem you're facing

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

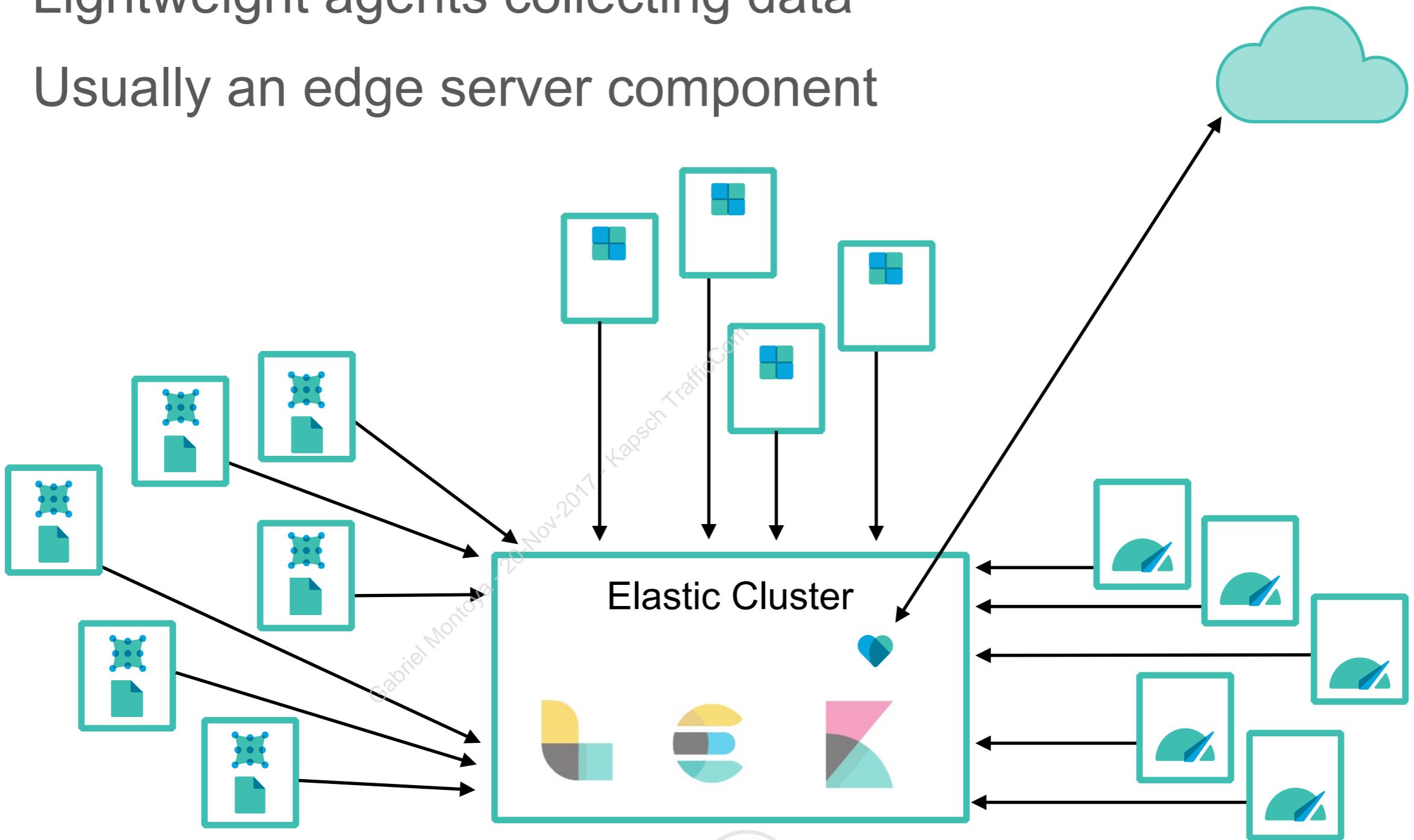


Getting Started with Beats

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

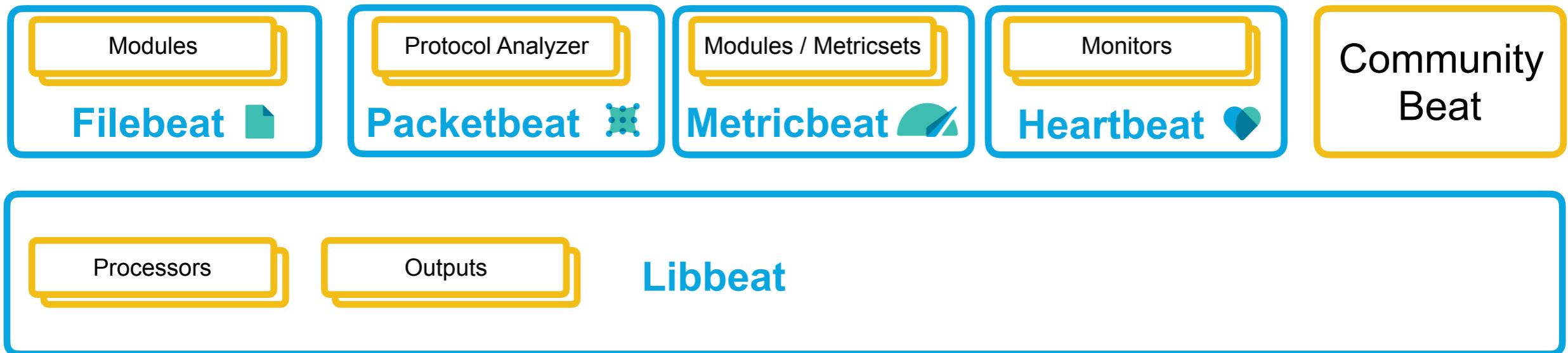
Beats

- Open source data shippers
- Lightweight agents collecting data
- Usually an edge server component



Beats

- Multiple flavors



- Every Beat has two main components:
 - data collector (flavor: file, packet, metric, heart, ...)
 - data processor and publisher (**Libbeat**)

Getting Started with Beats

- Distribution
 - binaries for each environment

Downloads: [DEB 32-BIT sha1](#) [DEB 64-BIT sha1](#) [RPM 32-BIT sha1](#)
[RPM 64-BIT sha1](#) [LINUX 32-BIT sha1](#) [LINUX 64-BIT sha1](#)
[MAC sha1](#) [WINDOWS 32-BIT sha1](#) [WINDOWS 64-BIT sha1](#)

APT and YUM repositories
are also available

- Install

- windows

```
unzip metricbeat-6.6.0-windows-x86_64.zip  
PS C:\Program Files\Metricbeat> .\install-service-metricbeat.ps1
```

- mac

```
tar -zxf metricbeat-6.0.0-amd64.tar.gz
```

- linux

```
tar -zxf metricbeat-6.0.0-amd64.tar.gz
```

- deb

```
sudo dpkg -i metricbeat-6.0.0-amd64.deb
```

- rpm

```
sudo rpm -vi metricbeat-6.0.0-x86_64.rpm
```



Getting Started with Beats

- Start

- windows

```
PS C:\Program Files\Metricbeat> Start-Service metricbeat
```

- mac

```
sudo chown root metricbeat.yml  
sudo chown root modules.d/system.yml  
sudo ./metricbeat
```

other beats might start differently in mac

- linux

```
./metricbeat
```

- deb

```
sudo service metricbeat start
```

- rpm

```
sudo service metricbeat start
```

- Stop

- windows

```
PS C:\Program Files\Metricbeat> Stop-Service metricbeat
```

- terminal

```
Ctrl+C # or kill <pid>
```

- rpm/deb

```
sudo service metricbeat stop
```



Getting Started with Beats

- Config File (<beats_name>.yml)

- extra fields (name, tags, fields)

```
tags: ["front-end", "web-tier"]
```

- output (elasticsearch, logstash, kafka, redis)

```
output.elasticsearch:  
  hosts: ["localhost:9200"]
```

- Elasticsearch templates

```
setup.template.settings:  
  index.number_of_shards: 1  
  index.codec: best_compression
```

- logging

```
# log levels: critical, error, warning, *info*, debug  
logging.level: warning
```



Getting Started with Beats

- Config File (<beats_name>.yml)

- Kibana server

```
# Starting with version 6.0.0, dashboards are loaded via
# the Kibana API. "localhost:5601" is the default config.
setup.kibana:
  host: "173.45.63.27:8080"
```

- Kibana pre-built dashboards

```
setup.dashboards.enabled: true
```

OR

```
./metricbeat setup --dashboards
```

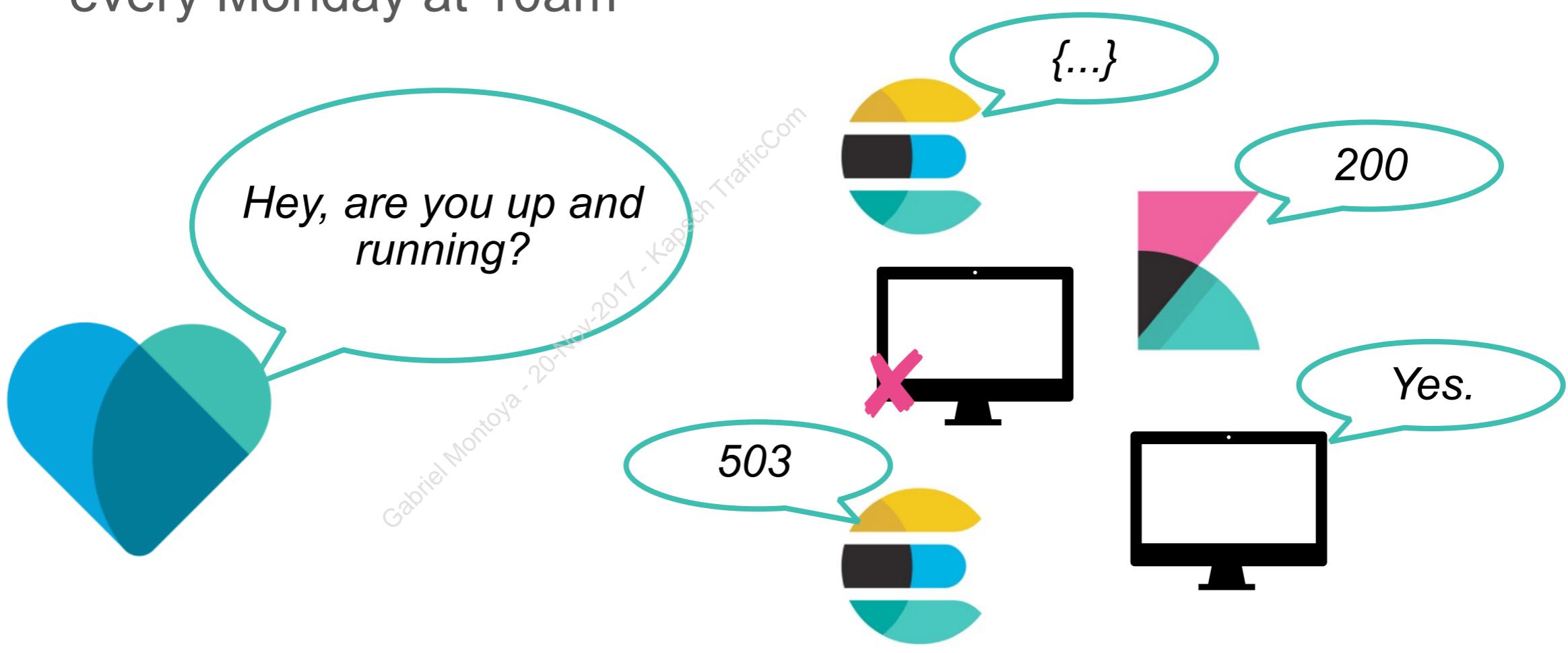
- <name>beat specific settings (covered later)

Heartbeat

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Heartbeat

- Tells you whether one or more services are reachable
- Monitors services based on a cronjob-fashion specified schedule, for example:
 - every 10 minutes
 - every Monday at 10am



Heartbeat Monitors

- ICMP (v4 and v6) Echo Requests
 - requires root access
- TCP
 - verify the endpoint by sending and/or receiving a custom payload
- HTTP
 - verify that the service returns the expected response, such as a specific status code, response header, or content
- Both TCP and HTTP monitors support SSL/TLS and some proxy settings.

Gabriel Montoya - 20-Nov-2011 - Kapsch OfficeCloud

Heartbeat Config

- The config file (**heartbeat.yml**) is in the heartbeat folder

```
heartbeat.monitors:
```

- type: icmp
schedule: '*/5 * * * *'
hosts: ["myhost"]
- type: tcp
schedule: '@every 5s'
hosts: ["otherhost:12345"]
mode: any

Monitors myhost using ICMP exactly every 5s (10:00:00, 10:00:05, and so on)

```
output.elasticsearch:
```

```
hosts: ["monitorcluster:9200"]
```

Monitors otherhost:1234 using TCP every 5s from start time

Store data in Elasticsearch at monitorcluster:9200

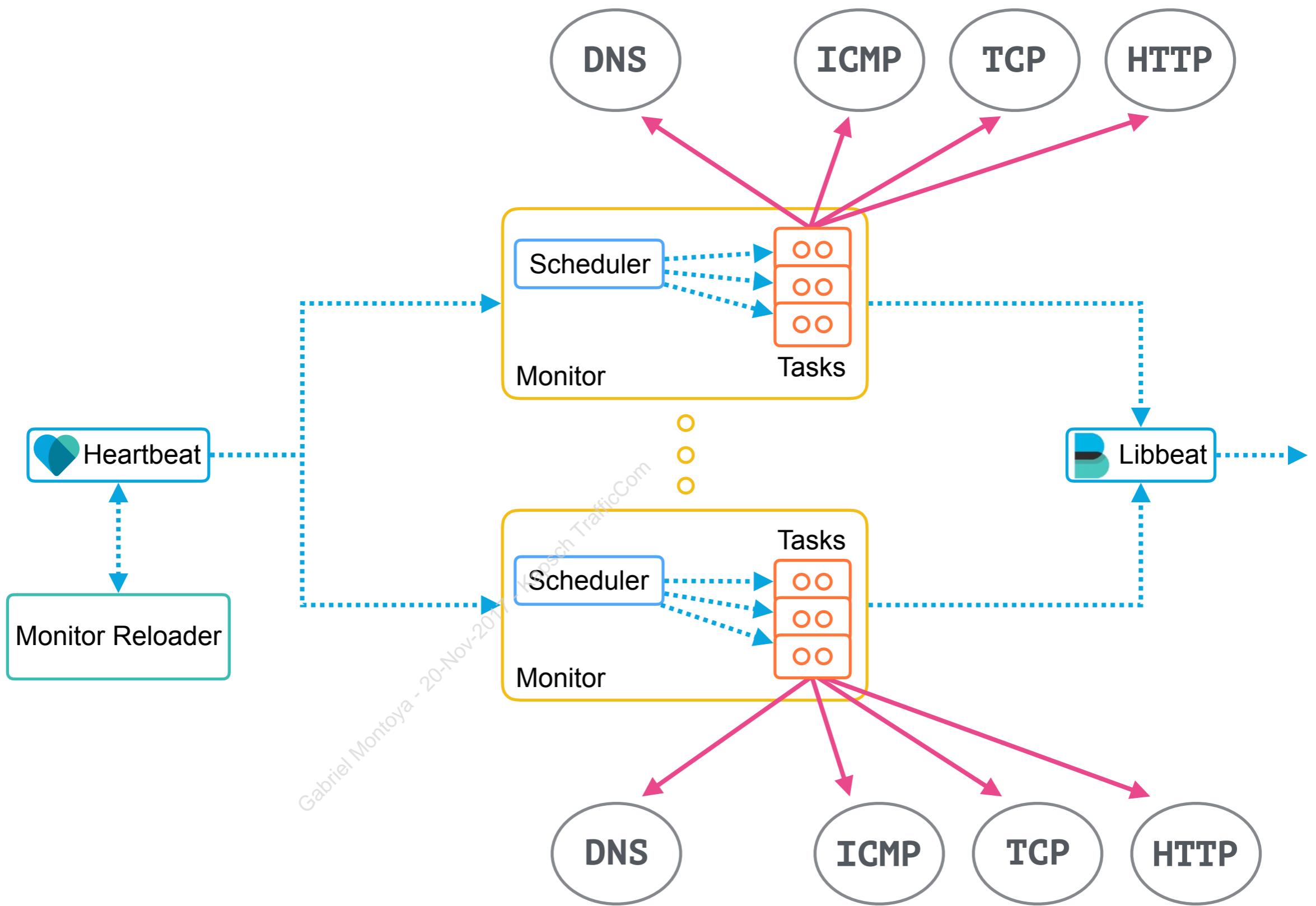
Heartbeat Config Options

- type
 - the type of monitor to run - one of: **icmp**, **tcp**, **http**
- schedule
 - cron-like expression that specifies the task schedule
- mode
 - **any**: pings only one IP address for a hostname (default)
 - **all**: pings all resolvable IPs for a hostname
- timeout
 - the total running time for each ping test (default 16s)

Gabriel Motraya - 20-Nov-2017 - Karish Trattner



Heartbeat Internals

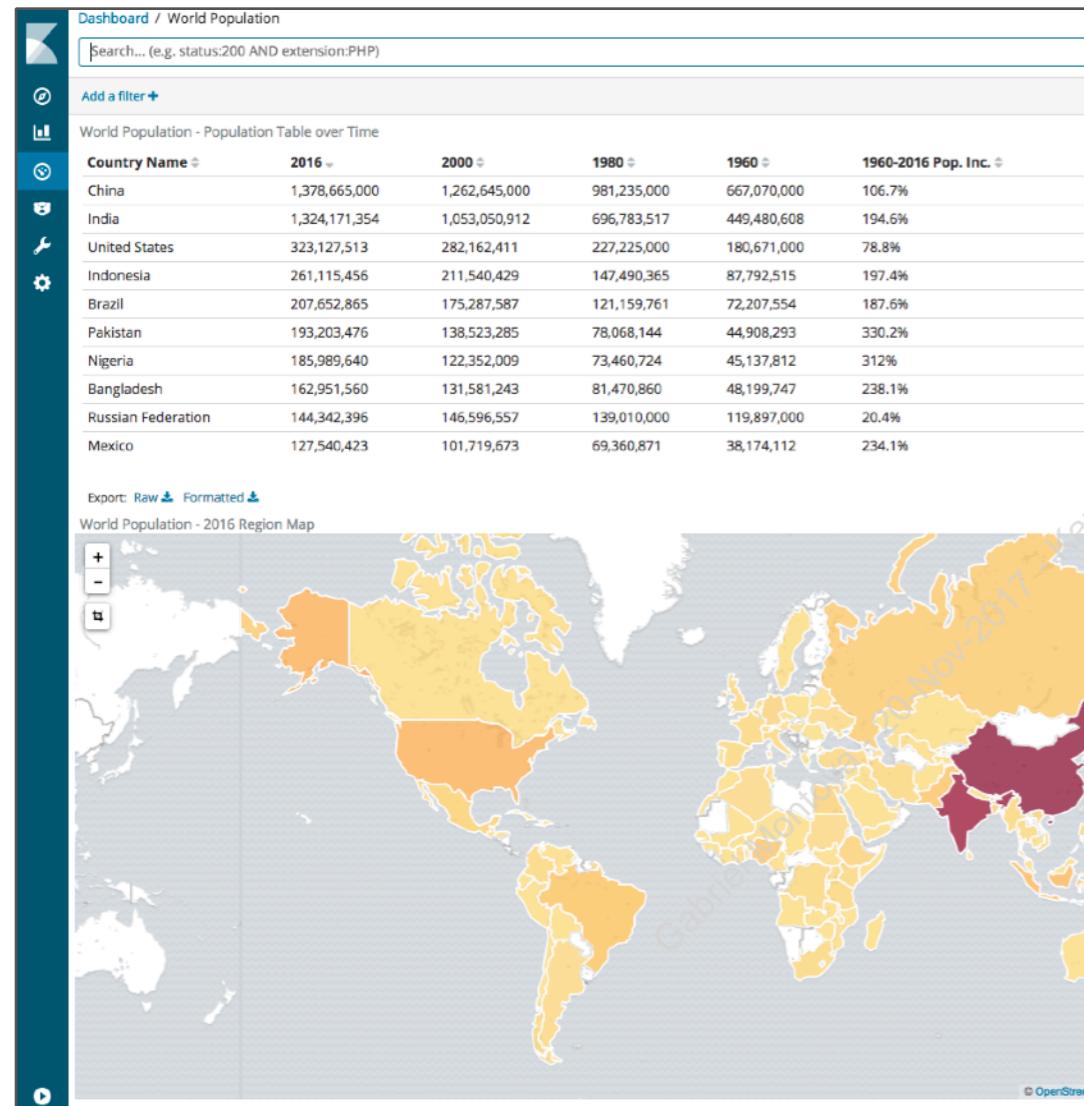


Kibana

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Kibana

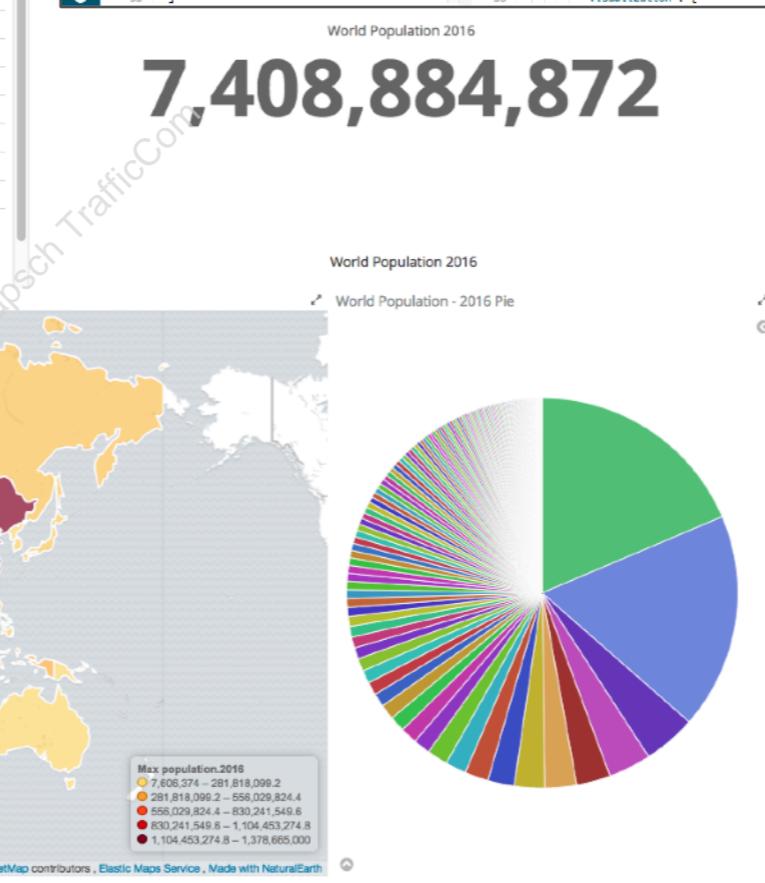
- Data Visualization
 - Stack Configuration
 - Elastic Stack UI



The screenshot shows the Kibana Dev Tools interface with the 'Console' tab selected. The left pane displays a list of recent Elasticsearch queries, and the right pane shows the results for the most recent query, which is a search for 'fb_world_population/_search'. The results include a summary of the search (took 8ms, 12 shards, 12 successful, 0 skipped, 0 failed), the total number of hits (748), and a list of individual document results. Each document result includes fields like '_index', '_type', '_id', '_score', and '_source', which contains detailed information about the document.

```
1 GET _cat/indices
2
3 GET _search
4 {
5   "query": {
6     "match_all": {}
7   }
8 }
9
10 GET fb_world_population/_search
11
12 GET fb_world_population/_search?q=country
13
14 GET ls_world_population/_search
15
16 DELETE ls_world_population
17
18 DELETE world_population
19
20 GET world_population/_search
21 {
22   "size": 0,
23   "aggs": {
24     "NAME": {
25       "geohash_grid": {
26         "field": "geo.location",
27         "precision": 3
28       },
29       "aggs": {
30         "NAME": {
31           "top_hits": {
32             "size": 10
33           }
34         }
35       }
36     }
37   }
38 }
39
40 GET world_population/_search
41 {
42   "size": 81,
43   "query": {
44     "match": {
45       "geo.location.keyword": "0,0"
46     }
47   }
48   , "_source": "country"
49   , "sort": [
50     {
51       "country.keyword": {
52         "order": "asc"
53     }
54   ]
55 }
```

```
1 - {
2   "took": 8,
3   "timed_out": false,
4   "shards": 12,
5   "total": 12,
6   "successful": 12,
7   "skipped": 0,
8   "failed": 0
9 },
10 "hits": {
11   "total": 748,
12   "max_score": 1,
13   "hits": [
14     {
15       "_index": ".kibana",
16       "_type": "doc",
17       "_id": "config:6.0.0-beta1",
18       "_score": 1,
19       "_source": {
20         "type": "config",
21         "config": {
22           "buildNum": 15799,
23           "defaultIndex": "d3f12b60-89a0-11e7-bc55-a5c147502291"
24         }
25       },
26     },
27     {
28       "_index": ".kibana",
29       "_type": "doc",
30       "_id": "visualization:61331190-89a7-11e7-bc55-a5c147502291",
31       "_score": 1,
32       "_source": {
33         "type": "visualization",
34         "visualization": {
35           "title": "World Population - 2016 Pie",
36           "isState": {"title": "World Population - 2016 Pie", "type": "pie", "params": {"type": "pie", "addTooltip": true, "addLegend": true, "legendPosition": "right", "isDonut": false}, "ags": [{"id": 1, "enabled": true, "type": "sum", "schema": "metric", "params": {"field": "population_2016"}}, {"id": 2, "enabled": true, "type": "terms", "schema": "segment", "params": {"field": "country.keyword", "size": 250, "order": "desc", "orderBy": "1"}]}, "uiStateJSON": "{}",
37         "description": "",
38         "version": 1,
39         "kibanaSavedObjectMeta": {
40           "searchSourceJSON": {"index": "d3f12b60-89a0-11e7-bc55-a5c147502291", "filter": [], "query": {"query": "", "language": "lucene"}},
41           "visJSON": []
42         }
43       }
44     },
45     {
46       "_index": ".kibana",
47       "_type": "doc",
48       "_id": "visualization:01bbb7c8-89a8-11e7-bc55-a5c147502291",
49       "_score": 1,
50       "_source": {
51         "type": "visualization",
52         "visualization": {
53           "title": "World Population - 2016 Pie",
```



Discover

kibana 199,946 hits New Save Open Share ⌛ August 31st 2016, 23:54:12.193 to September 3rd 2016, 12:53:31.706 ➔ Uses lucene query syntax 🔎

Discover Add a filter +

Visualize citibike-events

Dashboard

Timelon

Dev Tools

Management

Selected Fields

? _source

Available Fields

t _id

t _index

_score

t _type

bikeid

birth_year

event_id

t event_type

gender

station_id

station_latitude

station_longitude

t station_name

⌚ timestamp

trip_id

userid

t usertype

Count August 31st 2016, 23:54:12.193 - September 3rd 2016, 12:53:31.706 — Auto

Time ▾

_source

September 3rd 2016, 12:52:00.000

```
station_name: E 17 St & Broadway station_longitude: -73.99 trip_id: 117,194 gender: 0
station_id: 497 bikeid: 25,730 usertype: Customer userid: -1 birth_year: -
event_id: 35,773,965 event_type: end station_latitude: 40.737 timestamp: September 3r
d 2016, 12:52:00.000 _id: 35773965 _type: doc _index: citibike-events _score: -
```

September 3rd 2016, 12:51:00.000

```
station_name: W 38 St & 8 Ave station_longitude: -73.991 trip_id: 83,355 gender: 1
station_id: 523 bikeid: 18,667 usertype: Subscriber userid: 3,048 birth_year: 1,972
event_id: 35,469,414 event_type: end station_latitude: 40.755 timestamp: September 3r
d 2016, 12:51:00.000 _id: 35469414 _type: doc _index: citibike-events _score: -
```

September 3rd 2016, 12:50:00.000

```
station_name: 5 Ave & E 78 St station_longitude: -73.964 trip_id: 115,151 gender: 0
station_id: 3,143 bikeid: 17,162 usertype: Customer userid: -1 birth_year: -
event_id: 35,755,578 event_type: end station_latitude: 40.777 timestamp: September 3r
d 2016, 12:50:00.000 _id: 35755578 _type: doc _index: citibike-events _score: -
```

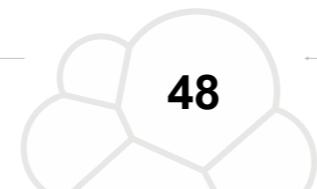
September 3rd 2016, 12:49:00.000

```
station_name: 5 Ave & E 78 St station_longitude: -73.964 trip_id: 115,128 gender: 0
station_id: 3,143 bikeid: 15,406 usertype: Customer userid: -1 birth_year: -
event_id: 35,755,371 event_type: end station_latitude: 40.777 timestamp: September 3r
d 2016, 12:49:00.000 _id: 35755371 _type: doc _index: citibike-events _score: -
```

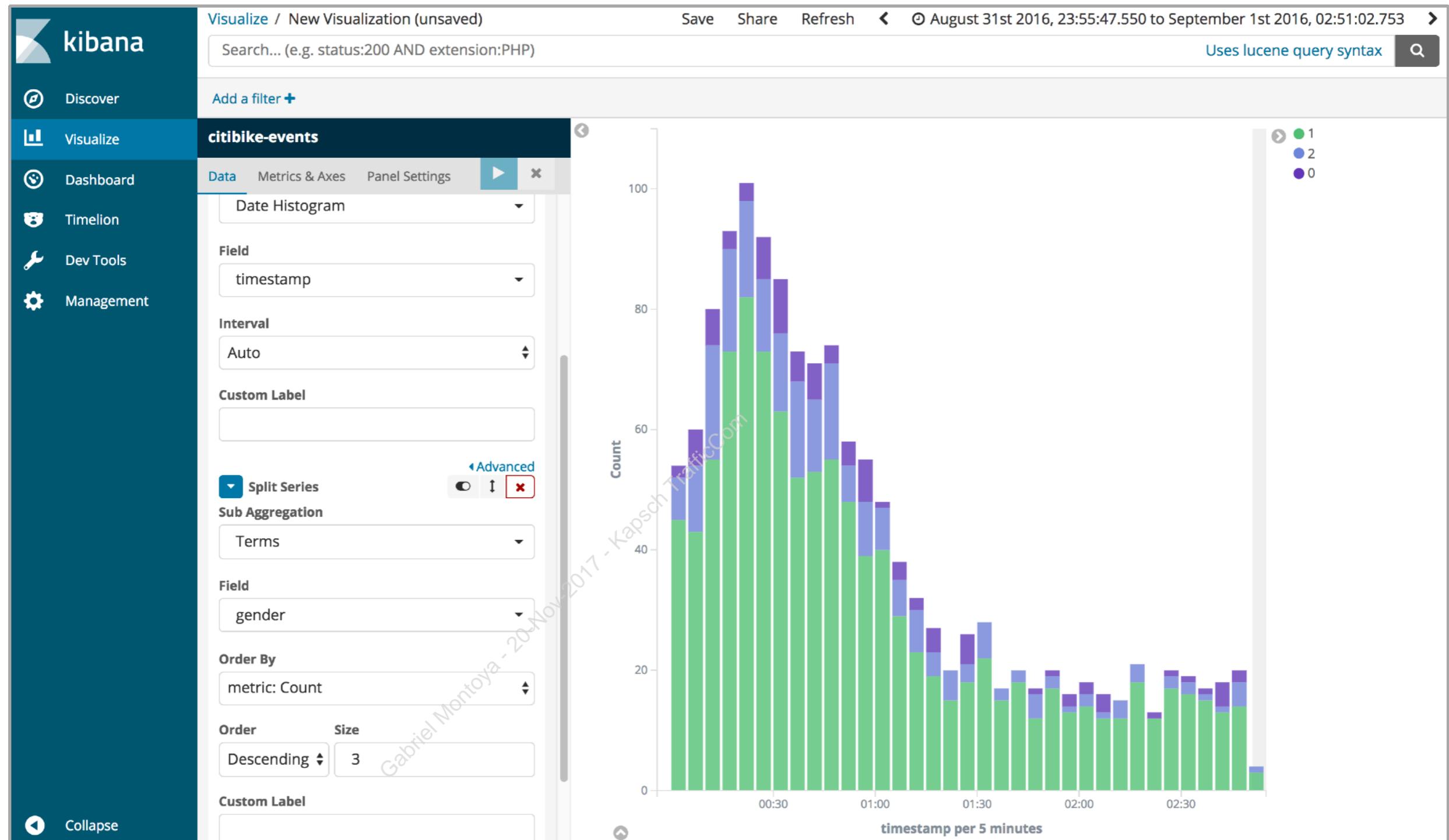
September 3rd 2016, 12:48:00.000

```
station_name: Avenue D & E 12 St station_longitude: -73.974 trip_id: 110,434
```

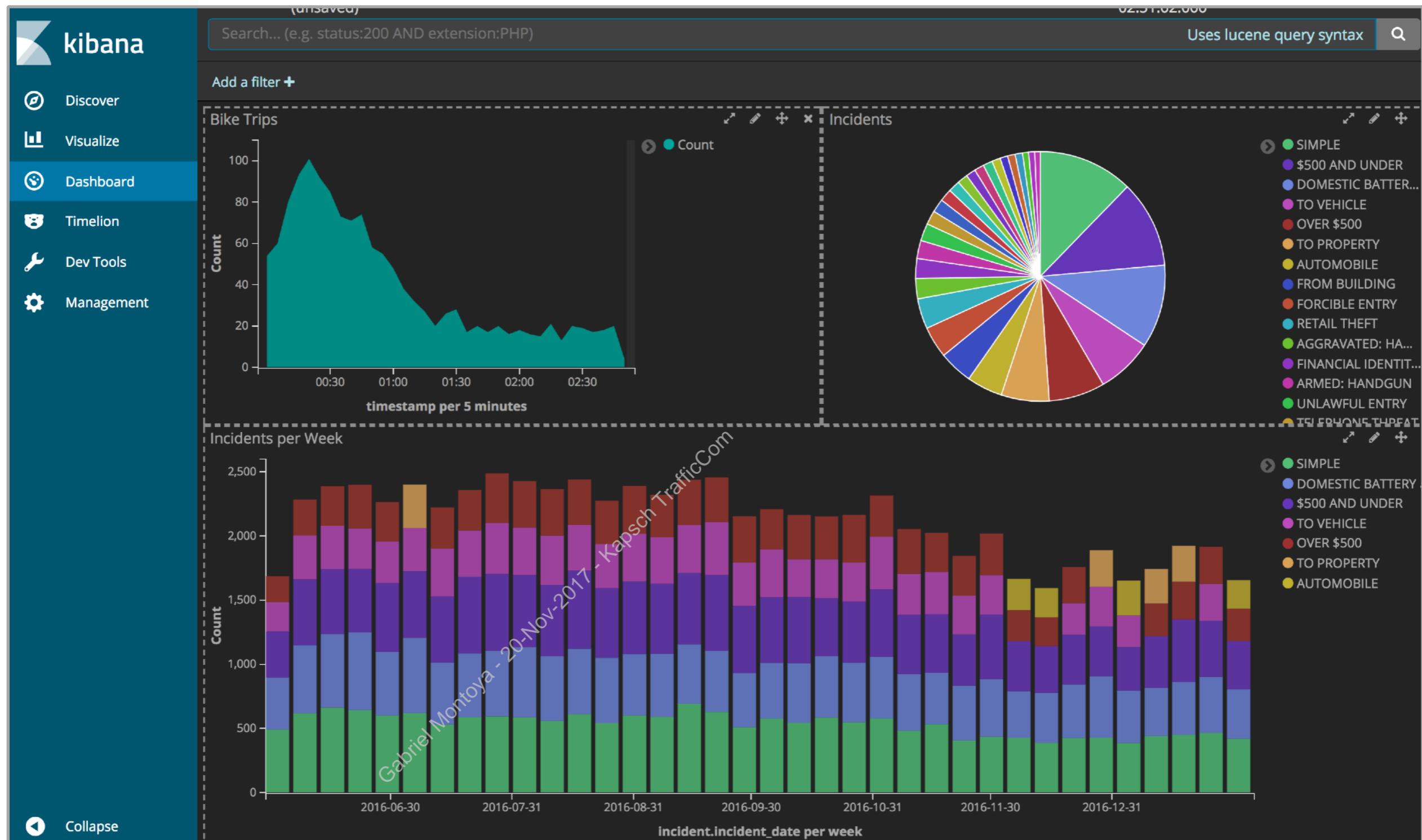
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Visualization



Dashboard



Index Pattern

Management / Kibana

Index Patterns Saved Objects Advanced Settings

+ Create Index Pattern

★ citibike-events

Time Filter field name: timestamp

This page lists every field in the **citibike-events** index and the field's associated core type as recorded by Elasticsearch. While this list allows you to view the core type of each field, changing field types must be done using Elasticsearch's [Mapping API](#).

fields (19) scripted fields (0) source filters (0)

Filter All field types ▾

name	type	format	searchable	aggregatable	excluded	controls
_id	string		✓	✓		
_index	string		✓	✓		
_score	number					
_source	_source					
_type	string		✓	✓		
bikeid	number		✓	✓		
birth_year	number		✓	✓		
event_id	number		✓	✓		
event_type	string		✓	✓		
gender	number		✓	✓		
station_id	number		✓	✓		
station_latitude	number		✓	✓		
station_longitude	number		✓	✓		
station_name	string		✓			

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- The Elastic Stack can be used to extract, enrich and analyze different types of data
- Starting with version 5.0, all of Elastic Stack products will be aligned, tested, and released together
- Beats are lightweight agents that collect data
- There are several Beats to collect different types of data: Heartbeat, Metricbeat, Packetbeat, Filebeat, Winlogbeat, {Community}beat
- Beats can output to Elasticsearch, Logstash, Redis and Kafka
- Heartbeat tells you whether services are reachable or not
- Kibana allows you to visualize data stored in Elasticsearch

Gabriel Montoya - 20-Nov-2017 - Kapsch Trafficsolutions



Quiz

1. Explain two use cases where the Elastic Stack can help?
2. Name three different types of Beats and what they do.
3. **True or False:** Heartbeat only supports ICMP and TCP.
4. **True or False:** You can use Heartbeat to verify if a website is returning a specific response.
5. How do you set Heartbeat to resolve all IPs for a hostname?
6. In Kibana, what is the difference between a Visualization and a Dashboard?
7. **True or False:** An Index Pattern defines the Elasticsearch indices that you want to explore.



Lab 1

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 2

System Metrics

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics**
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Beats Commands and Flags
- Metricbeat
- Metricbeat System Module
- Metricbeat Dashboards
- Beats Filtering and Enhancing

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats Commands and Flags

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats Commands

- Beats is a command-line friendly executable
- You can execute Beats with any of the following commands:
 - export
 - help
 - run
 - setup
 - test
 - version

Example of each option in Heartbeat

```
./heartbeat export  
./heartbeat help  
./heartbeat  
./heartbeat setup  
./heartbeat test config  
./heartbeat version
```

Beats Commands

- **setup** flags can be used to only load the specified one
 - --dashboards
 - --machine-learning
 - --template
- **test** flags define which part will be tested
 - modules (depends on the beat)
 - config
 - output

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats Global Flags

- **-c, --c FILE**
 - Specifies the configuration file to use for Heartbeat
- **-d, --d SELECTORS**
 - Enables debugging for the specified selectors
 - ▶ `-d '*'` to enable debugging for all components
 - ▶ `-d "publish"` displays all the "publish" related messages
- **-e, --e**
 - Logs to stderr and disables syslog/file output
- **-V, --V**
 - Logs INFO-level messages.

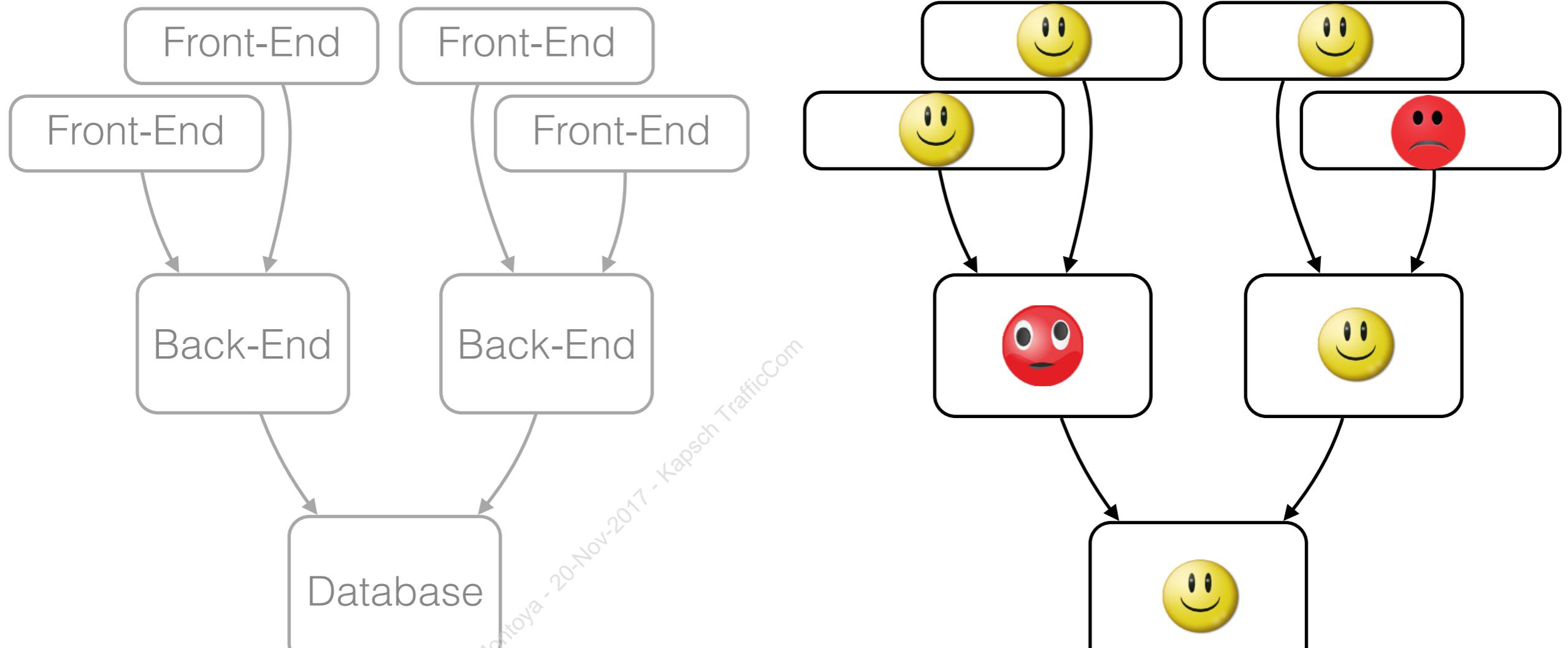
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Metricbeat

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Why Metrics?



Metricbeat

- Collects metrics from the operating system and from services running on the server



System module



Nginx



MongoDB



MySQL



PostgreSQL



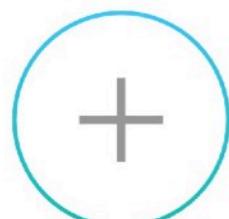
Redis



ZooKeeper



Apache



Gabriel Montoya - 20 Nov 2017 - Kapsch TrafficCom

Why Metricbeat?

- Multiple metrics in one event
- Sends more than just numbers
- No aggregations when data is fetched
- Metricbeat error events

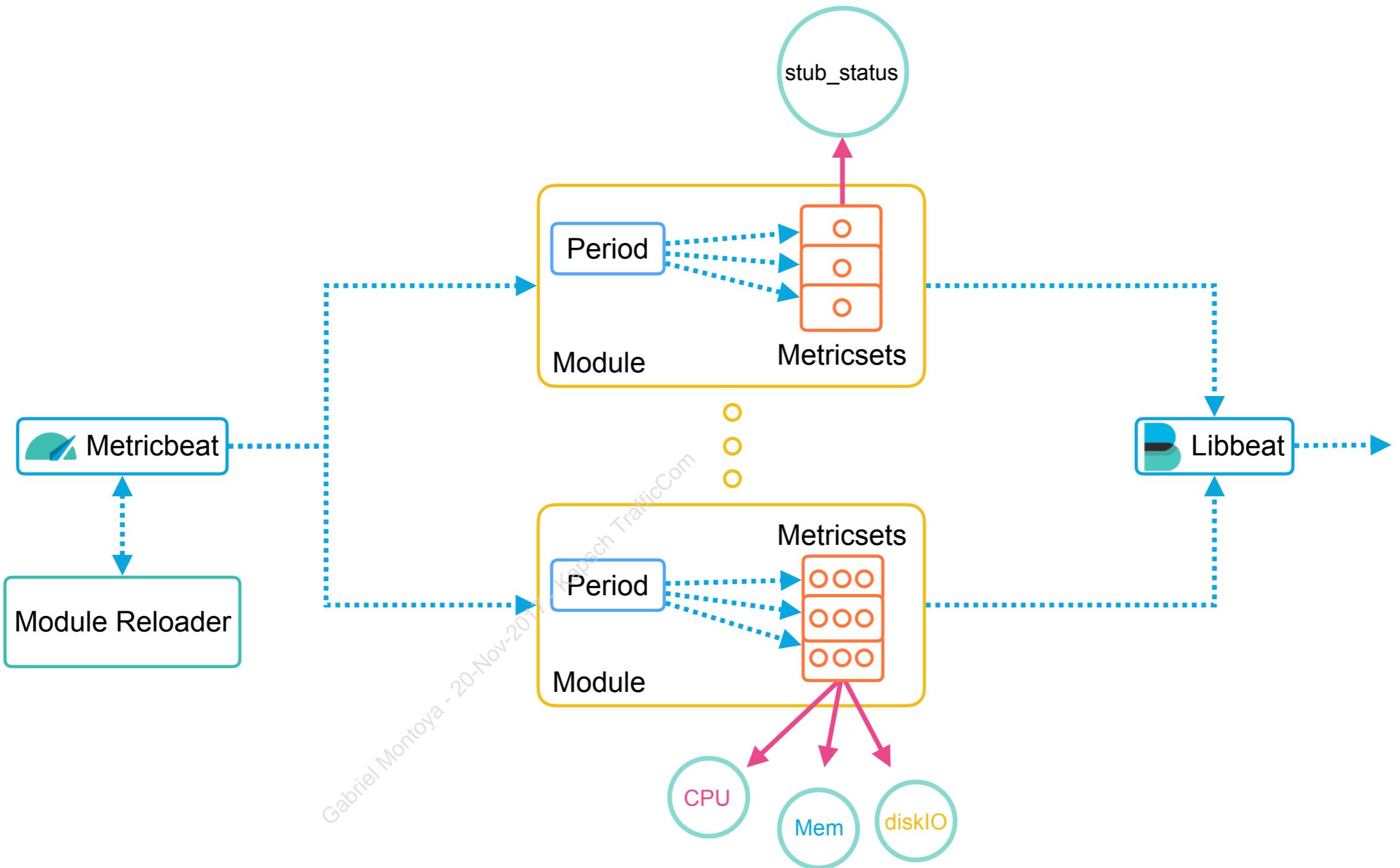
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Metricbeat

- Metricbeat consists of **modules** and **metricsets**
- **Module**
 - defines the basic logic for collecting data from a specific service
 - specifies details about the service, including how to connect and how often to collect metrics
 - each module has one or more metricsets
- **Metricset**
 - periodically fetches and structures the data
 - retrieve a list of multiple related metrics in a single request

Gabriel Montoya
10 Nov 2017 - Kapsch TrafficCom

Metricbeat Internals



Metricbeat

- You can config multiple modules of the same type
- The period is defined per configured module
- Metricbeat reuses connections whenever possible
- If the service is not reachable, Metricbeat generates an error event
 - 10s is the default timeout
- Metricbeat sends the events asynchronously
 - event retrieval is not acknowledged
 - if the configured output is not available, events may be lost

Gabriel Montaraz - 20-Nov-2017 - KaptonTrafficCom

Configuration

- metricbeat.yml

```
metricbeat.config.modules:  
  # Glob pattern for configuration loading  
  path: ${path.config}/modules.d/*.yml
```

- modules.d/

```
ubuntu@ip-172-31-0-79:~/metricbeat$ ls modules.d/  
aerospike.yml.disabled      golang.yml.disabled      memcached.yml.disabled    rabbitmq.yml.disabled  
apache.yml.disabled        haproxy.yml.disabled      mongodb.yml.disabled    redis.yml.disabled  
ceph.yml.disabled          http.yml.disabled        mysql.yml.disabled     system.yml  
couchbase.yml.disabled     jolokia.yml.disabled      nginx.yml.disabled     vsphere.yml.disabled  
docker.yml.disabled        kafka.yml.disabled       postgresql.yml.disabled windows.yml.disabled  
dropwizard.yml.disabled    kibana.yml.disabled      prometheus.yml.disabled zookeeper.yml.disabled  
elasticsearch.yml.disabled kubernetes.yml.disabled  
ubuntu@ip-172-31-0-79:~/metricbeat$
```

system is the only one
enabled by default

Configuration Example

- apache.yml.disabled

```
- module: apache
  metricsets: ["status"]
  period: 10s

  # Apache hosts
  hosts: ["http://127.0.0.1"]
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Metricbeat Document Structure

```
{  
  "@timestamp": "2017-09-02T13:40:14.517Z",  
  "beat": {  
    "name": "ip-172-31-0-79",  
    "hostname": "ip-172-31-0-79",  
    "version": "6.0.0-beta2"  
  },
```

beats metadata

Common to all beats, contains the time **timestamp** of the event and the **beats instance information**

```
"metricset": {  
  "module": "apache",  
  "name": "status",  
  "rtt": 1121,  
  "host": "127.0.0.1"  
},
```

metricset metadata

This is an event from the status **metricset** of the apache **module**, and the **Round Trip Time** to collect the data from **host** 127.0.0.1 was 1121us

```
"apache": {  
  "status": {  
    ...  
  }  
}
```

metricset data

module { metricset { data } } is the object structure, and the data depends on the metricset type



Metricbeat Error Document Structure

```
{  
  "@timestamp": "2017-09-02T13:40:14.517Z",  
  "beat": {  
    ...  
  },  
  
  "metricset": {  
    ...  
  },  
  
  "apache": {  
    status { }  
  },  
  
  "error": {  
    "message": "error making http request: Get  
    http://127.0.0.1/server-status?auto=: dial  
    tcp 127.0.0.1:80: getsockopt: connection refused"  
  }  
}
```

beats metadata

metricset metadata

empty metricset data

error data



Metricbeat System Module

Gabriel Montoya - 20-Nov-2017 - KapschTrafficCom

Metricbeat System Module

- How do we monitor system resources?

CPU

Mem

diskIO

filesystem

load

network

cores

processes

Gabriel Motoya - 20-Nov-2017 - Kapsch TrafficCom



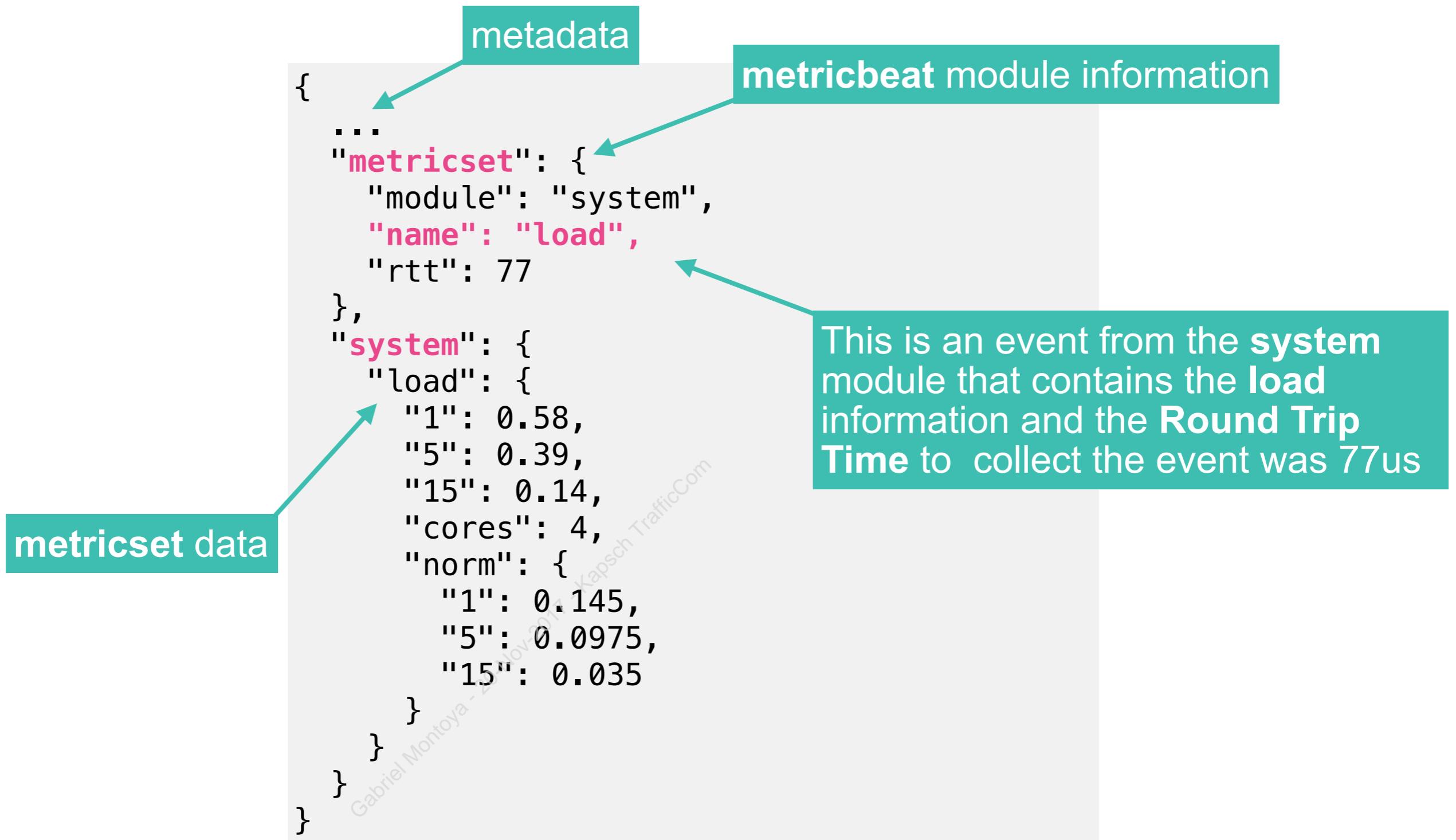
modules.d/system.yml

```
- module: system
  period: 10s
  metricsets:
    - cpu
    - load
    - memory
    - network
    - process
    - process_summary
  processes: ['./*']
  process.include_top_n:
    by_cpu: 5      # include top 5 processes by CPU
    by_memory: 5   # include top 5 processes by memory

- module: system
  period: 1m
  metricsets:
    - filesystem
    - fsstat
  processors:
    - drop_event.when.regexp:
        system.filesystem.mount_point: '^/(sys|cgroup|proc|dev|etc|host|lib)(\$|/)'  
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom
```



Metricbeat System Module (load)



Metricbeat System Module (cpu)

```
{  
  ...  
  "metricset": {  
    "module": "system",  
    "name": "cpu",  
    "rtt": 162  
  },  
  "system": {  
    "cpu": {  
      "cores": 4  
      "iowait": {  
        "pct": 0  
      },  
      "total": {  
        "pct": 0  
      },  
      "user": {  
        "pct": 0  
      },  
      "system": {  
        "pct": 0  
      },  
      ...  
    } } }
```

This is an event from the **system** module that contains a **cpu** information and the **Round Trip Time** to collect the event was 162us

cpu data

Gabriel Montoya - 20-Nov-2017 - Kapch TrafficCom

Metricbeat System Module (process)

```
{  
  ...  
  "metricset": {  
    "module": "system",  
    "name": "process",  
    "rtt": 30105  
  }  
  "system": {  
    "process": {  
      "cmdline": "/sbin/init",  
      "cpu": {  
        ...  
      },  
      "ppid": 0,  
      "username": "root",  
      "pid": 1,  
      "pgid": 1,  
      "memory": {  
        ...  
      },  
      "name": "systemd",  
      "state": "sleeping",  
      "cgroup": {  
        ...  
      }  
    }  
  }  
}
```

This is an event from the **system** module that contains a **process** information and the **Round Trip Time** to collect the event was ~30ms

process data



Metricbeat System Module (filesystem)

```
{  
  ...  
  "metricset": {  
    "module": "system",  
    "name": "filesystem",  
    "rtt": 597  
  },  
  "system": {  
    "filesystem": {  
      "available": 1534222336,  
      "free_files": 1924385,  
      "used": {  
        "pct": 0.0271,  
        "bytes": 42668032  
      },  
      "files": 1924910,  
      "type": "tmpfs",  
      "mount_point": "/run",  
      "free": 1534222336,  
      "device_name": "tmpfs",  
      "total": 1576890368  
    }  
  }  
}
```

This is an event from the **system** module that contains a **filesystem** information and the **Round Trip Time** to collect the event was 597us

filesystem data



Metricbeat Dashboards

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

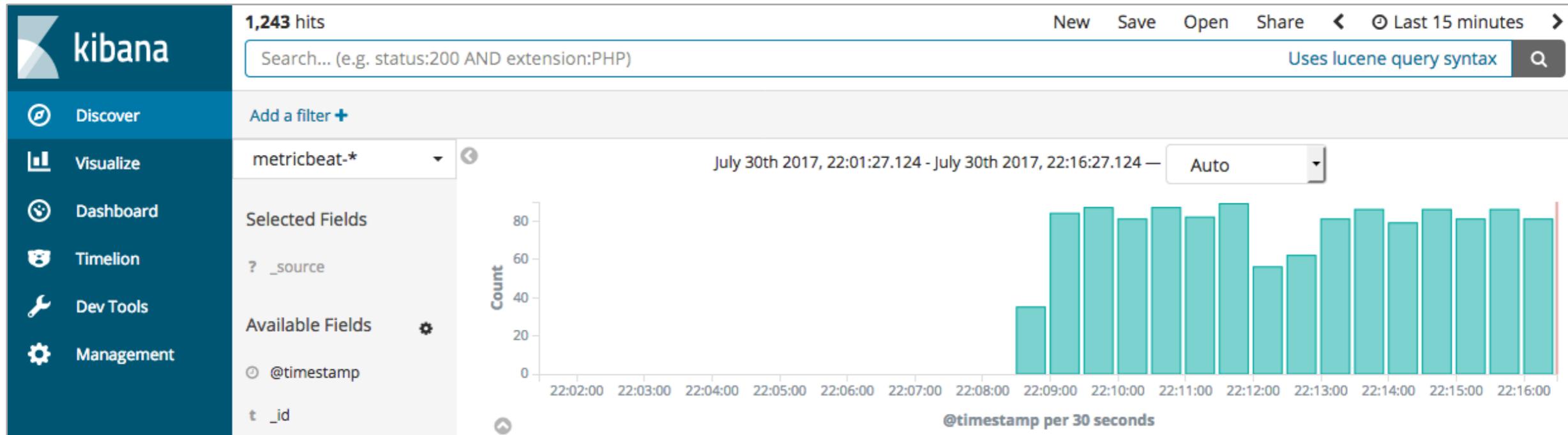


Metricbeat Dashboards

- Easy to start monitoring your servers in Kibana
- Pre-created Metricbeat dashboards
- Customize them to meet your needs
- Load dashboards:
 - metricbeat.yml `setup.dashboards.enabled: true`
 - command `./metricbeat setup --dashboards`
- It also creates a `metricbeat-*` index pattern.

Gabriel Montoya - 20-Nov-2017 - Search Traffic

Metricbeat Dashboards



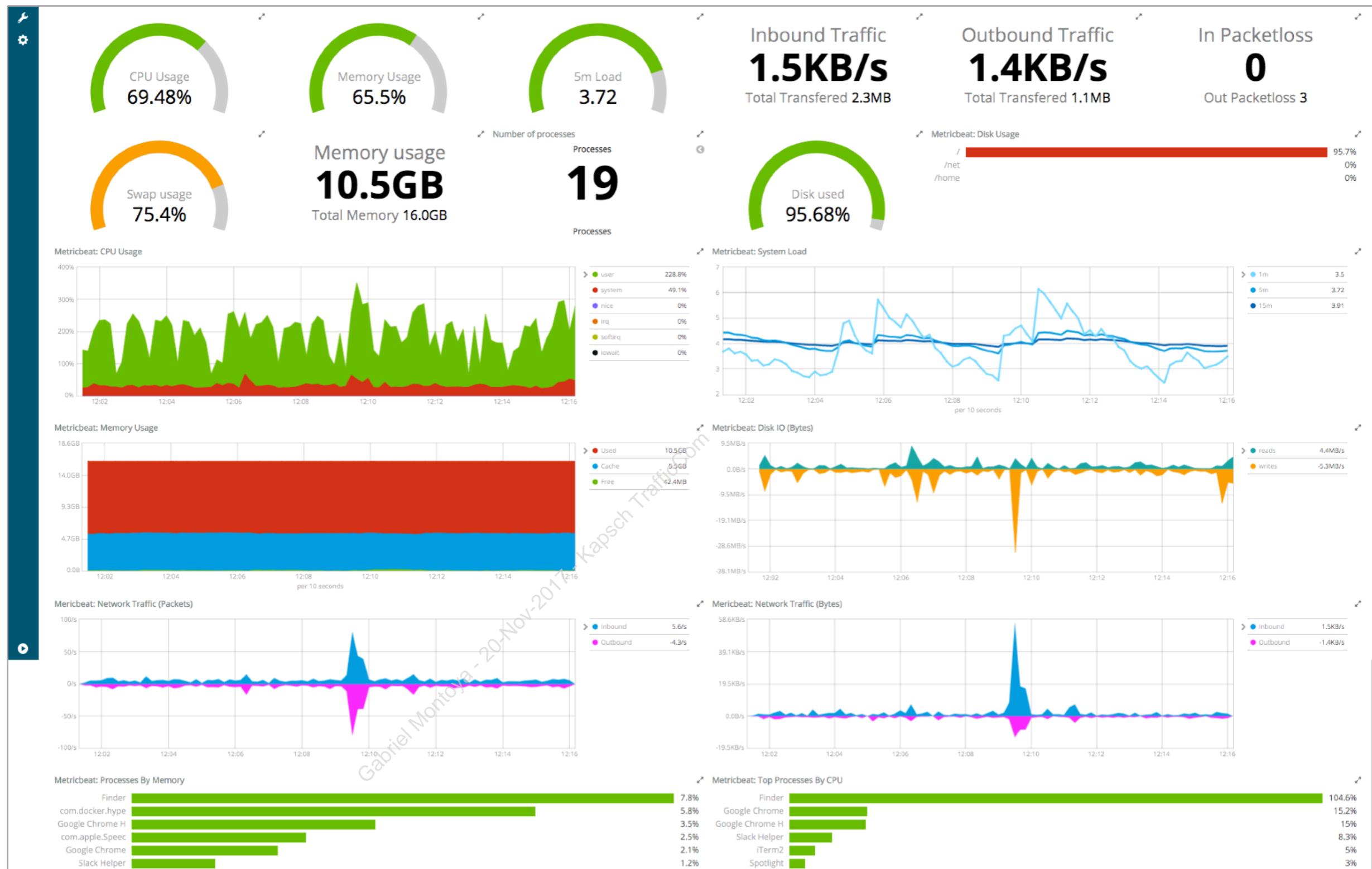
The Kibana Dashboard interface lists six available dashboards. The search bar shows "Met". The list includes:

- Name ▲
- Metricbeat - Apache HTTPD server status
- Metricbeat CPU/Memory per container
- Metricbeat MongoDB
- Metricbeat MySQL
- Metricbeat host overview
- Metricbeat system overview

Page navigation shows "1-6 of 6" with previous and next buttons.



Metricbeat Dashboards



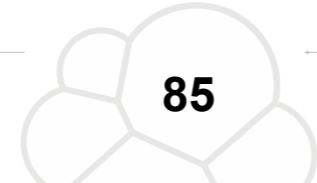
Beats Filtering and Enhancing

Gabriel Montoya - 20-Nov-2017 - KapschTm.Com

Beats Filtering and Enhancing

- Before sending to Elasticsearch you can process your data with "processors".
- The libbeat library provides processors for:
 - Limit which fields you index
 - Enrich data with additional metadata
 - Drop entire events that are not interesting to you
- Added to `{any}beat.yml` configuration file

Gabriel Montoya - 20-Nov-2017 - KapselTrafficCom



Processors Syntax

- Define processors in the config file:

```
processors:  
  - <processor_name>:  
    <optional parameters>  
    when:  
      <condition>  
  - <processor_name>: ←  
    <optional parameters>  
    when:  
      <condition>
```

Set a condition to execute on:

- equals
- contains
- regexp
- range
- or
- and
- not

Add as many processors as you'd like, but mind the used resources.

The type of processor you want to use:

- drop_event
- drop_fields
- add_cloud_metadata
- add_kubernetes_metadata
- add_docker_metadata
- add_locale
- include_fields
- decode_json_fields



Adding Cloud Metadata

- Add cloud metadata to your documents from various cloud platforms
 - AWS, GCE, Digital Ocean, QCloud, ECS

```
processors:  
- add_cloud_metadata: ~
```

This is the *processor_name*

```
{  
  "meta": {  
    "cloud": {  
      "availability_zone": "projects/1234567890/zones/us-east1-b",  
      "instance_id": "192818372378231292",  
      "machine_type": "projects/1234567890/machineTypes/f1-micro",  
      "project_id": "my-gce-packetbeat-project",  
      "provider": "gce"  
    }  
  }  
}
```

In GCE this would add this to your documents:



Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- The **test** command can be used to test config and output
- The **setup** command can be used to load dashboards
- Use **-e** to output to console and **-d "*"** to fully enable debug
- **Metricbeat** collects metrics from the operating system and from services running on the server
- Metricbeat consists of **modules** and **metricsets**
- The **system** module is responsible by collecting metrics from the running operating system
- Pre-created {Metric}beat dashboards can be loaded using the **setup** command
- Processors can be used to drop and mutate events



Quiz

1. How do you test the Beats configuration before executing?
2. **True or False:** Use the **load** command to load pre-built dashboards.
3. **True or False:** Use **-d "elasticsearch"** to display all the elasticsearch related messages.
4. Which beat do you use to monitor an operating system?
5. **True or False:** To monitor an separating system you should add the **system** module to the **metricbeat.yml** file.
6. **True or False:** It is not possible to drop metrics of processes executed by the root user.
7. **True or False:** Each **metricset** inside a module can have a different execution period.

Lab 2

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 3

Service Metrics

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Metricbeat Modules
- Nginx Module
- MySQL Module
- Docker Module
- Elasticsearch
- Elasticsearch CRUD
- Console

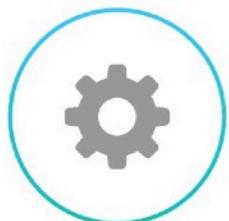
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Metricbeat Modules

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Metricbeat

- Collects metrics from the operating system and from services running on the server.



System module



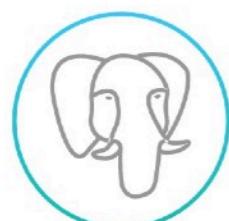
Nginx



MongoDB



MySQL



PostgreSQL



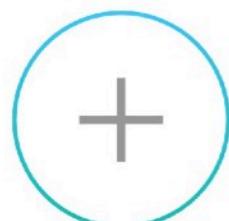
Redis



ZooKeeper



Apache



Adding Modules

- By default, metricbeat will load modules from `modules.d/`

```
metricbeat.config.modules:  
  path: ${path.config}/modules.d/*.yml
```

- By default, only the **system** module is enabled

```
ubuntu@ip-172-31-0-79:~/metricbeat$ ls modules.d/  
aerospike.yml.disabled      golang.yml.disabled      memcached.yml.disabled    rabbitmq.yml.disabled  
apache.yml.disabled        haproxy.yml.disabled      mongodb.yml.disabled    redis.yml.disabled  
ceph.yml.disabled          http.yml.disabled        mysql.yml.disabled      system.yml  
couchbase.yml.disabled     jolokia.yml.disabled      nginx.yml.disabled      vsphere.yml.disabled  
docker.yml.disabled        kafka.yml.disabled        php_fpm.yml.disabled   windows.yml.disabled  
dropwizard.yml.disabled    kibana.yml.disabled      postgresql.yml.disabled zookeeper.yml.disabled  
elasticsearch.yml.disabled kubernetes.yml.disabled  prometheus.yml.disabled  
ubuntu@ip-172-31-0-79:~/metricbeat$
```

- To add a module, simply rename the config file and restart metricbeat

```
mv modules.d/nginx.yml.disabled modules.d/nginx.yml  
./metricbeat test config  
./metricbeat
```



Dynamic Reload

- Metricbeat can dynamically reload modules from config files when there are changes.
- Set `reload.enabled` to `true`

```
metricbeat.config.modules:  
  path: ${path.config}/modules.d/*.yml  
  reload.enabled: true
```

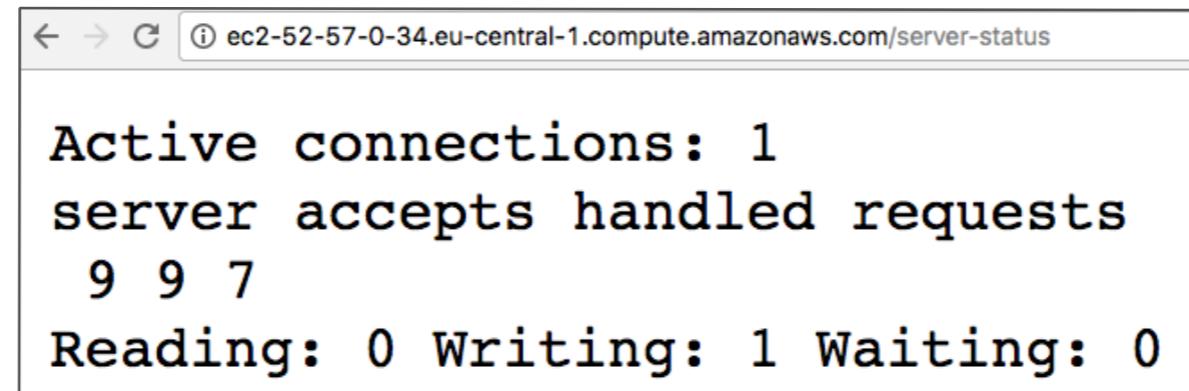
- Now, if there is a change in a module file it will be automatically loaded.
- For example, to enable mysql module just rename the file
`mv modules.d/mysql.yml.disabled modules.d/mysql.yml`
- `reload.period` defines how often metricbeat will check for modifications.

Nginx Module

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Nginx Module

- Periodically fetches metrics from Nginx servers.
- Scrapes the server status data (`ngx_http_stub_status`)



```
Active connections: 1
server accepts handled requests
 9 9 7
Reading: 0 Writing: 1 Waiting: 0
```

- `mv modules.d/nginx.yml.disabled modules.d/nginx.yml`

```
- module: nginx
  metricsets: ["stubstatus"]
  period: 10s

  # Nginx hosts
  hosts: ["http://127.0.0.1"]

  # Path to server status. Default server-status
  #server_status_path: "server-status"
```

Nginx Module Document Structure

```
{  
  ...  
  "metricset": {  
    "module": "nginx",  
    "name": "stubstatus",  
    "host": "127.0.0.1",  
    "rtt": 750  
  },  
  "nginx": {  
    "stubstatus": {  
      "current": 8,  
      "writing": 1,  
      "active": 1,  
      "waiting": 0,  
      ...  
    }  
    "total": {  
      "requests": 8,  
      "reading": 0,  
      "dropped": 0,  
      "accepts": 10,  
      "handled": 10,  
      "hostname": "127.0.0.1"  
    }  
  }  
}
```

Annotations for the Nginx module document structure:

- current number of client requests**: Points to the value 8 under the key "current" in the "stubstatus" section.
- active client connections**: Points to the value 1 under the key "active" in the "stubstatus" section.
- idle client connections waiting for a request**: Points to the value 0 under the key "waiting" in the "stubstatus" section.
- total number of client requests**: Points to the value 8 under the key "requests" in the "total" section.
- dropped client connections**: Points to the value 0 under the key "dropped" in the "total" section.
- handled client connections**: Points to the value 10 under the key "handled" in the "total" section.

MySQL Module

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

MySQL Module

- Periodically fetches metrics from MySQL servers
- Collects data by running a SHOW GLOBAL STATUS; query
- mv modules.d/mysql.yml.disabled modules.d/mysql.yml

```
- module: mysql
metricsets: ["status"]
period: 10s

# Host DSN should be defined as "user:pass@tcp(127.0.0.1:3306)/"
hosts: ["root:secret@tcp(127.0.0.1:3306)"]

# username and password can either be set in the DSN (takes precedence)
# or using the username and password config options.
#username: root
#password: secret

# By setting raw to true, all raw fields from the status metricset will
# be added to the event. This option is intended for advanced use cases.
#raw: false
```

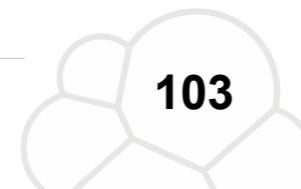


MySQL Module Document Structure

```
{  
  ...  
  "metricset": {  
    "host": "mysql:3306", "module": "mysql", "name": "status", "rtt": 115  
  },  
  "mysql": {  
    "status": {  
      "connections": 11,  
      "flush_commands": 1,  
      "max_used_connections": 3,  
      "opened_tables": 110,  
      "bytes": { "received": 992, "sent": 40657 },  
      "binlog": { ... },  
      "created": { ... },  
      "aborted": { ... },  
      "delayed": { ... },  
      "open": { ... },  
      "threads": { ... },  
      "command": { "delete": 0, "insert": 0, "select": 9, "update": 0 }  
    }  
  }  
}
```

total number of bytes sent and received from all clients

total number of times each query executed since startup



Docker Module

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Docker Module

- Periodically fetches metrics from Docker containers
 - currently not tested on Windows
- mv modules.d/docker.yml.disabled modules.d/docker.yml

```
metricbeat.modules:  
- module: docker  
  metricsets: [  
    "container", "cpu", "diskio", "healthcheck", "info", "memory", "network"  
  ]  
  hosts: ["unix:///var/run/docker.sock"]  
  period: 10s
```

many different metricsets

be careful with periods shorter than 3 seconds

Gabriel Montoya 20-Nov-2017 Kapsch TrafficCom

Docker Module Document Structure

```
{  
  ...  
  "metricset": {  
    "host": "/var/run/docker.sock",  
    "module": "docker", ← docker module  
    "name": "info", ← info metricset  
    "rtt": 115  
  },  
  "docker": {  
    "info": {  
      "containers": {  
        "paused": 0,  
        "running": 11, ← total number of paused/running/stopped containers  
        "stopped": 1,  
        "total": 12 ← total number of existing containers  
      },  
      "id": "CRDS:UHQ6:W3IR:GR0Z:IUG3:VS2E:4QJJ:FJWG:PRQ4:R25R:VNNP:NE62",  
      "images": 119  
    }  
  }  
}
```

total number of existing images

unique Docker host identifier

total number of existing containers

total number of paused/running/stopped containers

info metricset

docker module

Gabriel Montoya - Kapsch TrafficCom Nov-2017

Elasticsearch

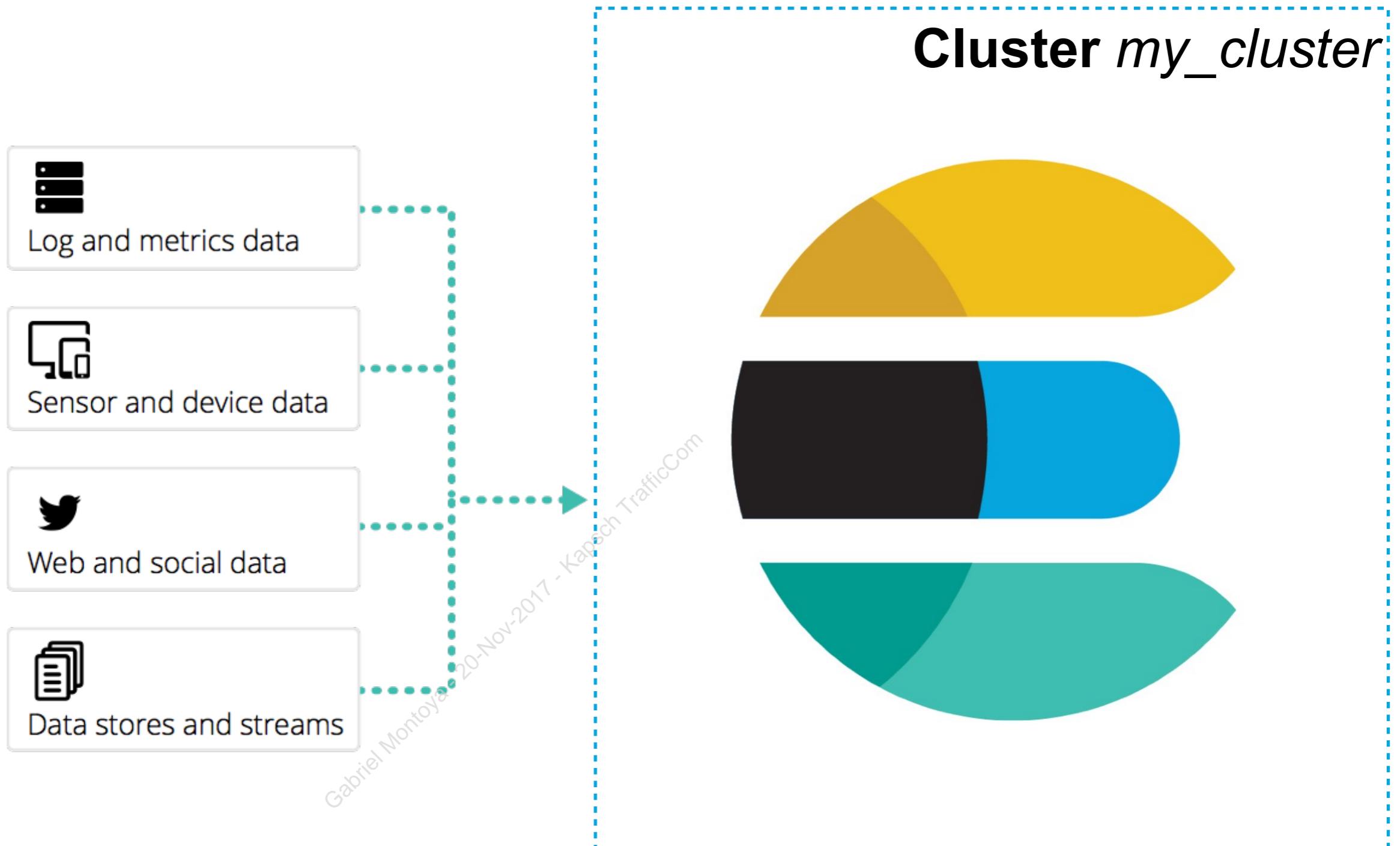
Gabriel Montoya - 20-Nov-2017 - Kapsch Trafficcom



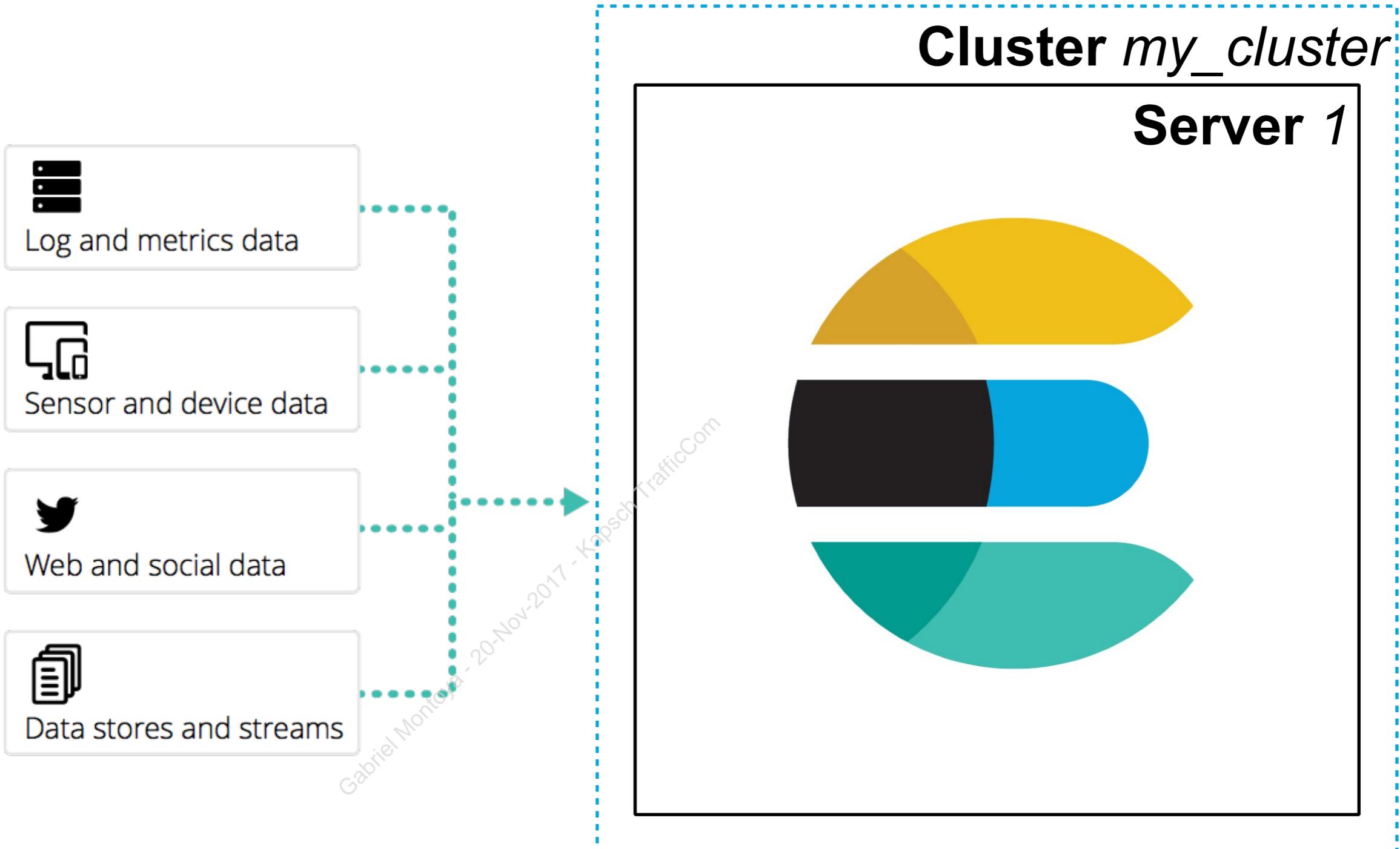
Elasticsearch

- What
 - Highly scalable open-source search and analytics engine
 - search: full-text, structured, geo, suggestions, highlighting, ...
 - analytics: trends, statistics, summarizations, anomalies, ...
 - real time & near-real time
- Why
 - Easy to use: simple RESTful API
 - log and event data analysis
 - full text search
 - alerting and classification
 - recommendations

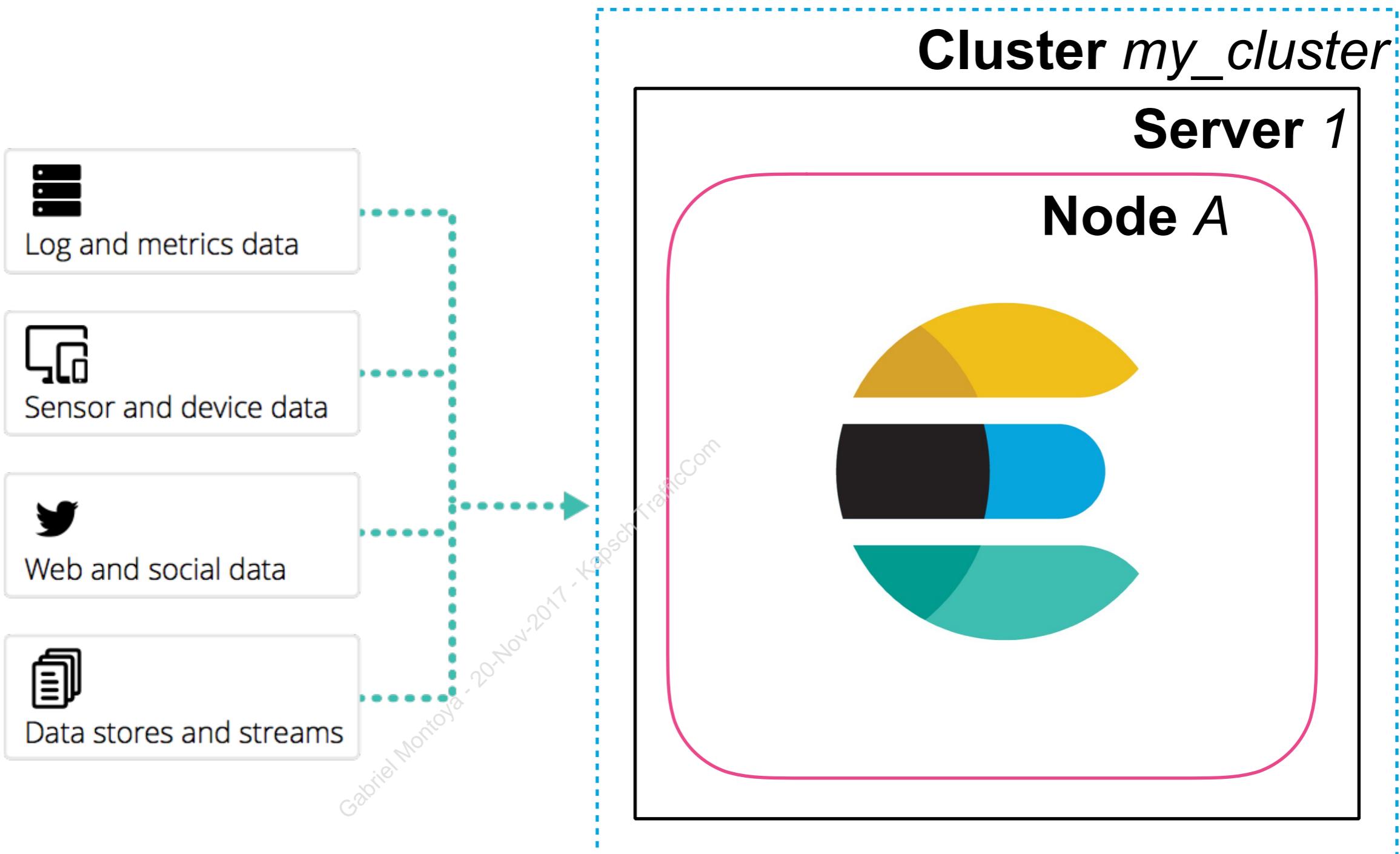
Terminology



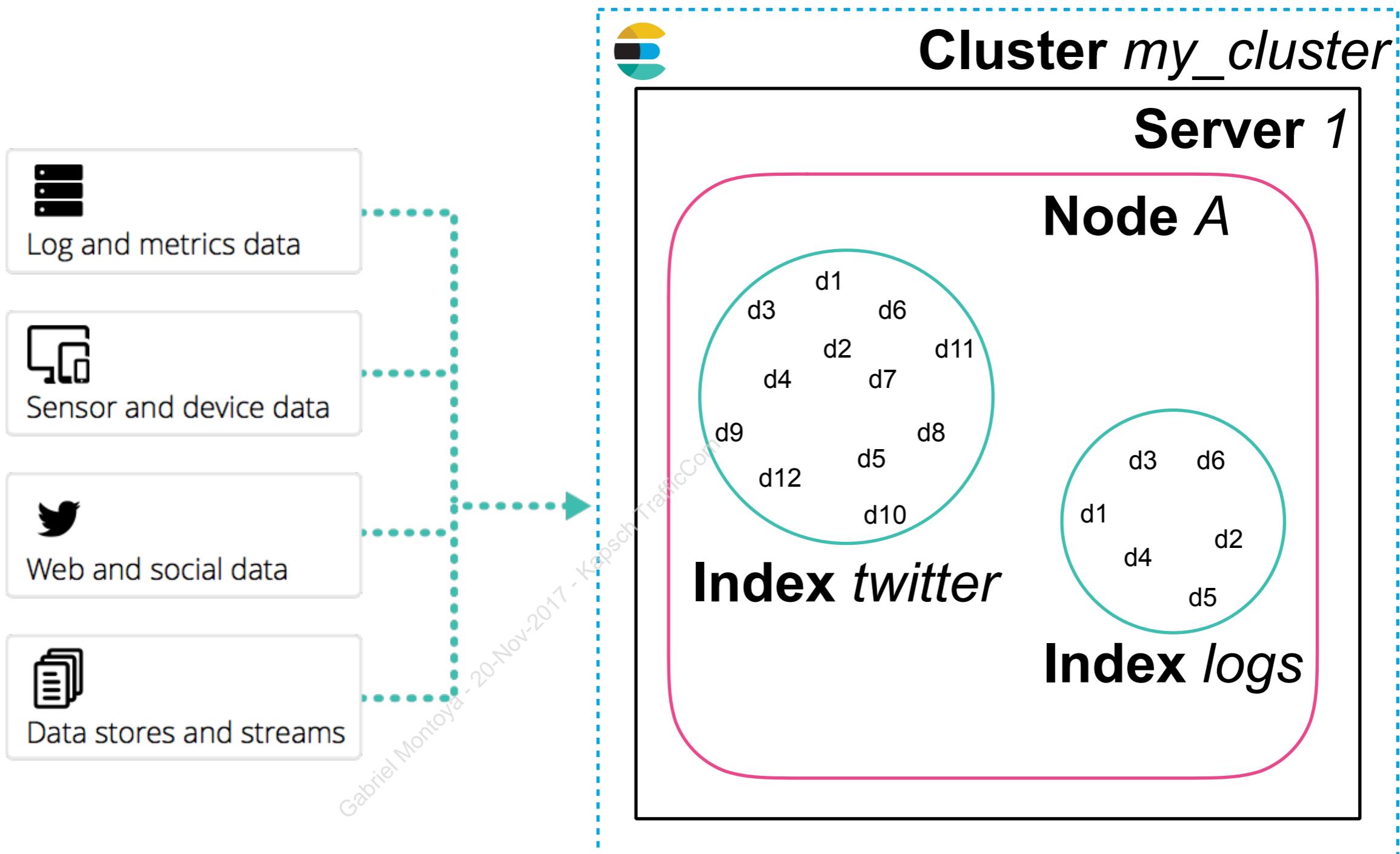
Terminology



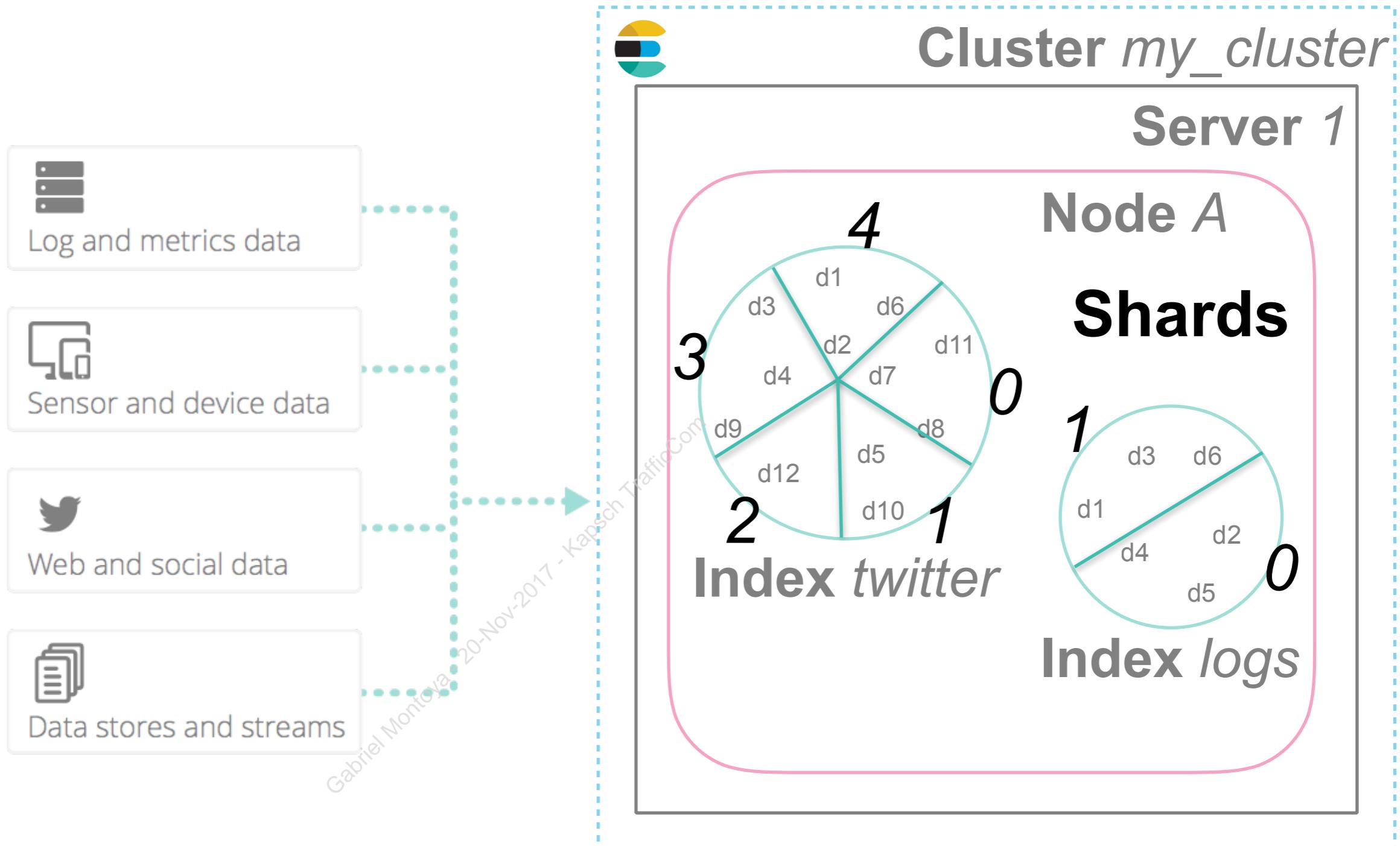
Terminology



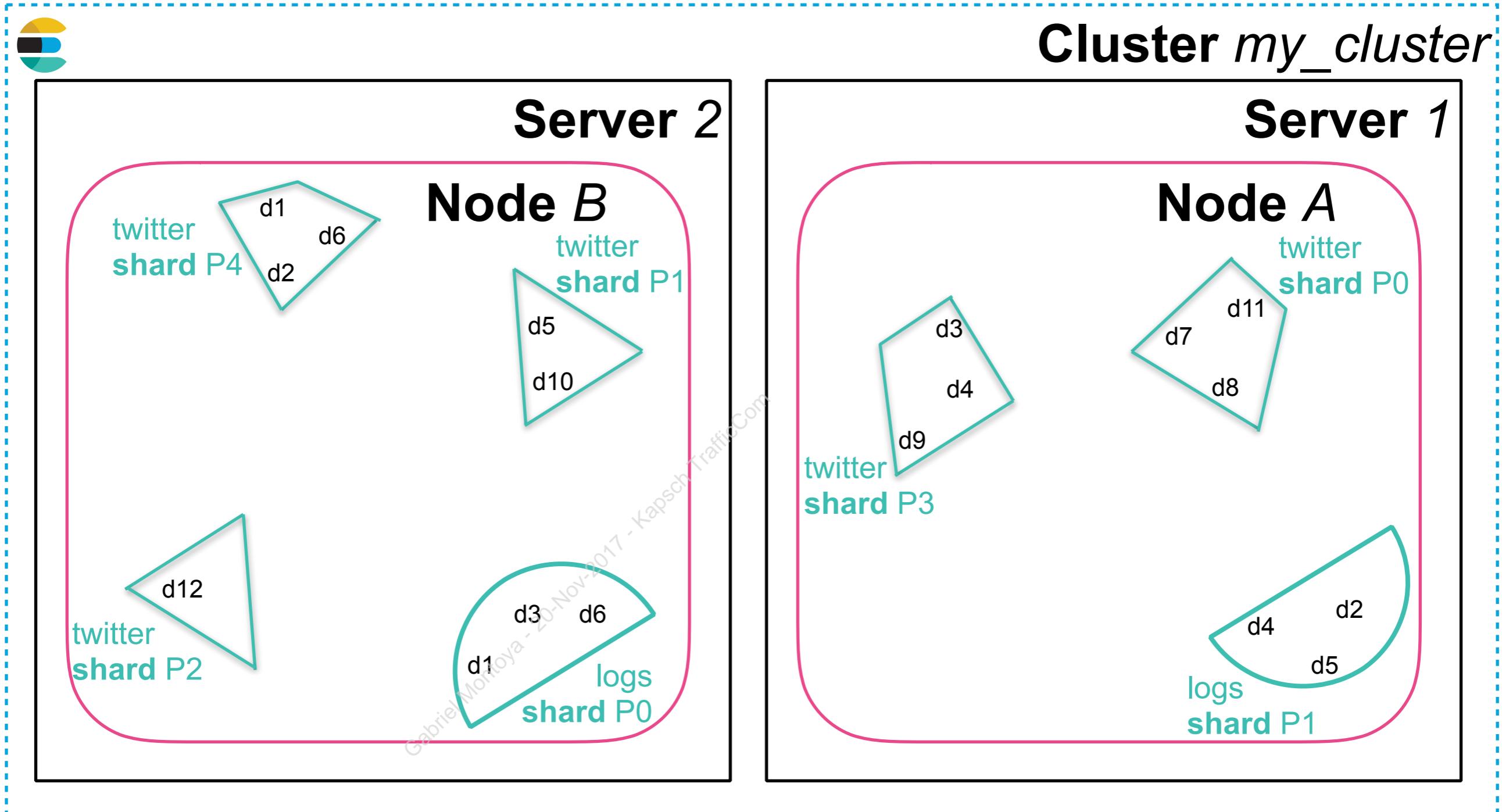
Terminology



Partition

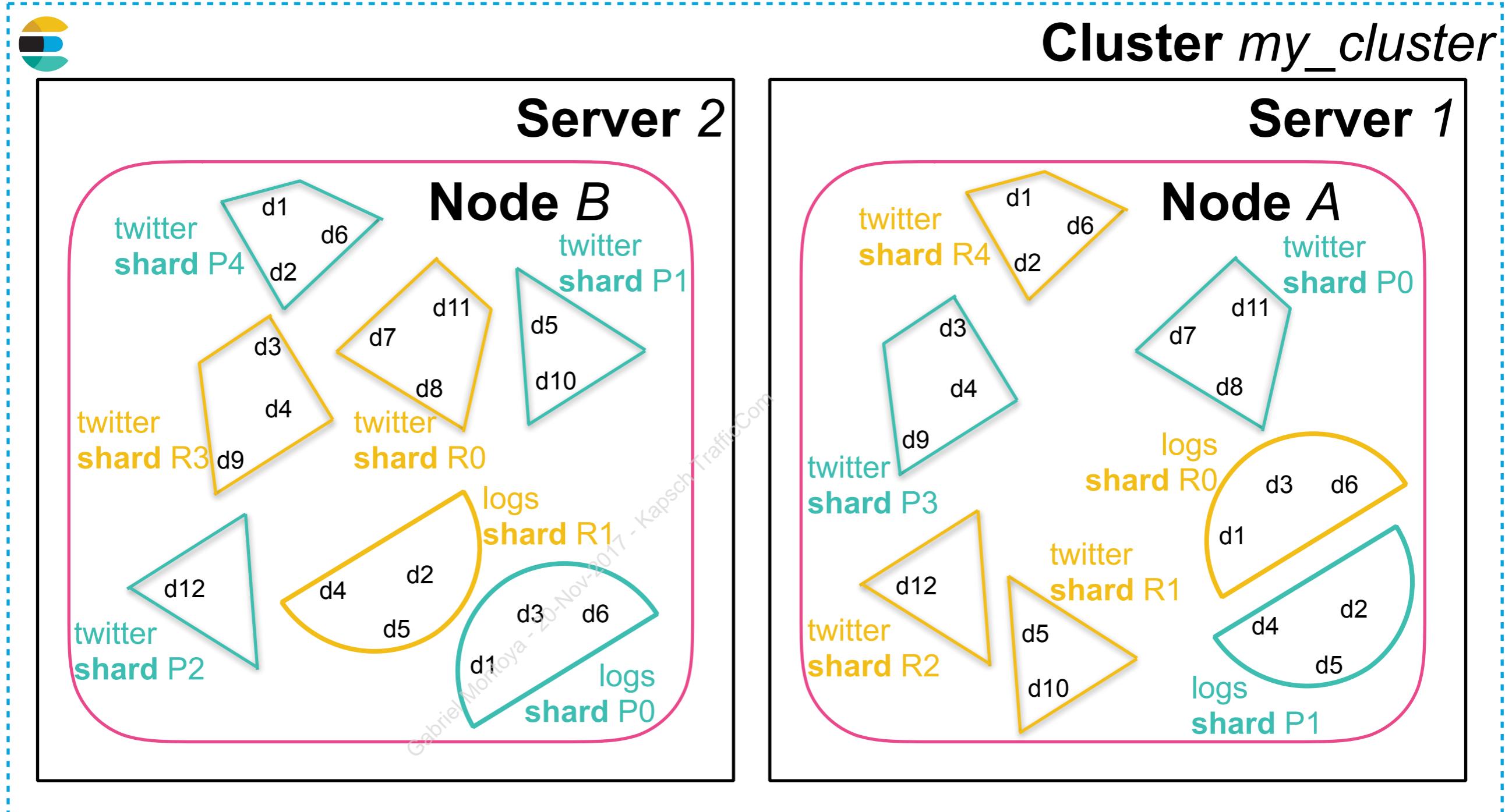


Distribution



Replication

- Primaries
- Replicas



Elasticsearch CRUD

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



CRUD Operations

In Elasticsearch the data is called a **document** and is serialized as JSON

Index	<pre>PUT my_index/doc/4 { "username" : "kimchy", "comment" : "I like search!" }</pre>
Create	<pre>PUT my_index/doc/4/_create { "username" : "kimchy", "comment" : "I like search!" }</pre>
Read	<pre>GET my_index/doc/4</pre>
Update	<pre>POST my_index/doc/4/_update { "doc" : { "comment" : "I love search!!" } }</pre>
Delete	<pre>DELETE my_index/doc/4</pre>



Search

Search criteria

Search

Destination/Hotel Name:
San Francisco

Work Leisure ?

Check-in

Check-out

Rooms: 1
Adults: 2
Children: 0

Search

San Francisco: 454 properties found

3 Reasons to Visit: Golden Gate Bridge, Fisherman's Wharf & cable cars



[Check out the best of San Francisco before you find a place to stay...](#)

Lock in a great price for your stay on these dates:

Jul 27 — Jul 28

Jul 28 — Jul 29

Jul 29 — Jul 30

Jul 30 — Jul 31

Our Top Picks First

Stars ▾

Distance From Downtown

Review Score ▾



Hotel Nikko San Francisco ★★★★

[Union Square, San Francisco – Subway Access](#)

Excellent 8.6

3,706 reviews

[Show prices](#)

Just a 5-minute walk from Union Square, Hotel Nikko San Francisco features an on-site restaurant. Pillow-top bedding and a flat-screen TV are provided in all rooms.

Booked 62 times today



Hotel Zephyr San Francisco ★★★★

[Fisherman's Wharf, San Francisco](#)

Very good 8.3

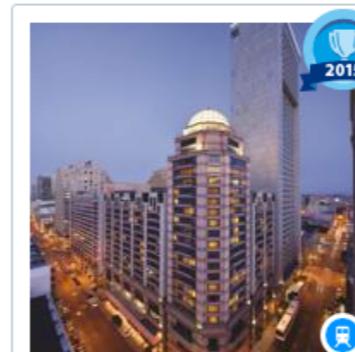
WiFi 8.9

2,747 reviews

[Show prices](#)

Offering a fitness center, Hotel Zephyr San Francisco is located a short 330 yards from Pier 39 Fisherman's Wharf. Free WiFi access is available in this waterfront hotel.

Booked 65 times today



Hilton San Francisco Union Square ★★★★

[Union Square, San Francisco – Subway Access](#)

Good 7.6

3,817 reviews

[Show prices](#)

Located in the heart of downtown San Francisco and just a 5-minute walk from the Powell Street subway station, this Hilton features an on-site Herb n' Kitchen and a beautiful courtyard pool.

Booked 25 times today



Search and Aggregations

Search criteria

Search

Destination/Hotel Name:
San Francisco

Work Leisure

Check-in
Check-in Date

Check-out
Check-out Date

Rooms: 1
Adults: 2
Children: 0

Search

San Francisco: 454 properties found

3 Reasons to Visit: Golden Gate Bridge, Fisherman's Wharf & cable cars

Check out the best of San Francisco before you find a place to stay...

Lock in a great price for your stay on these dates:

Jul 27 — Jul 28

Jul 28 — Jul 29

Jul 29 — Jul 30

Jul 30 — Jul 31

Our Top Picks First Stars Distance From Downtown Review Score



Hotel Nikko San Francisco ★★★★

Union Square, San Francisco – Subway Access

Excellent 8.6

3,706 reviews

Show prices



Hotel Zephyr San Francisco ★★★★

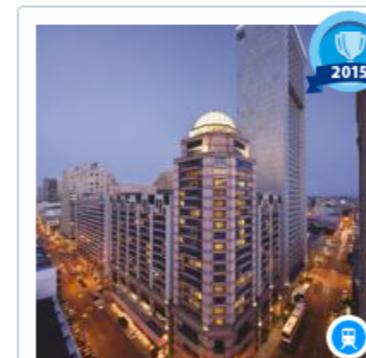
Fisherman's Wharf, San Francisco

Very good 8.3

WiFi 8.9

2,747 reviews

Show prices



Hilton San Francisco Union Square ★★★★

Union Square, San Francisco – Subway Access

Good 7.6

3,817 reviews

Show prices



Aggregations

Filter by:

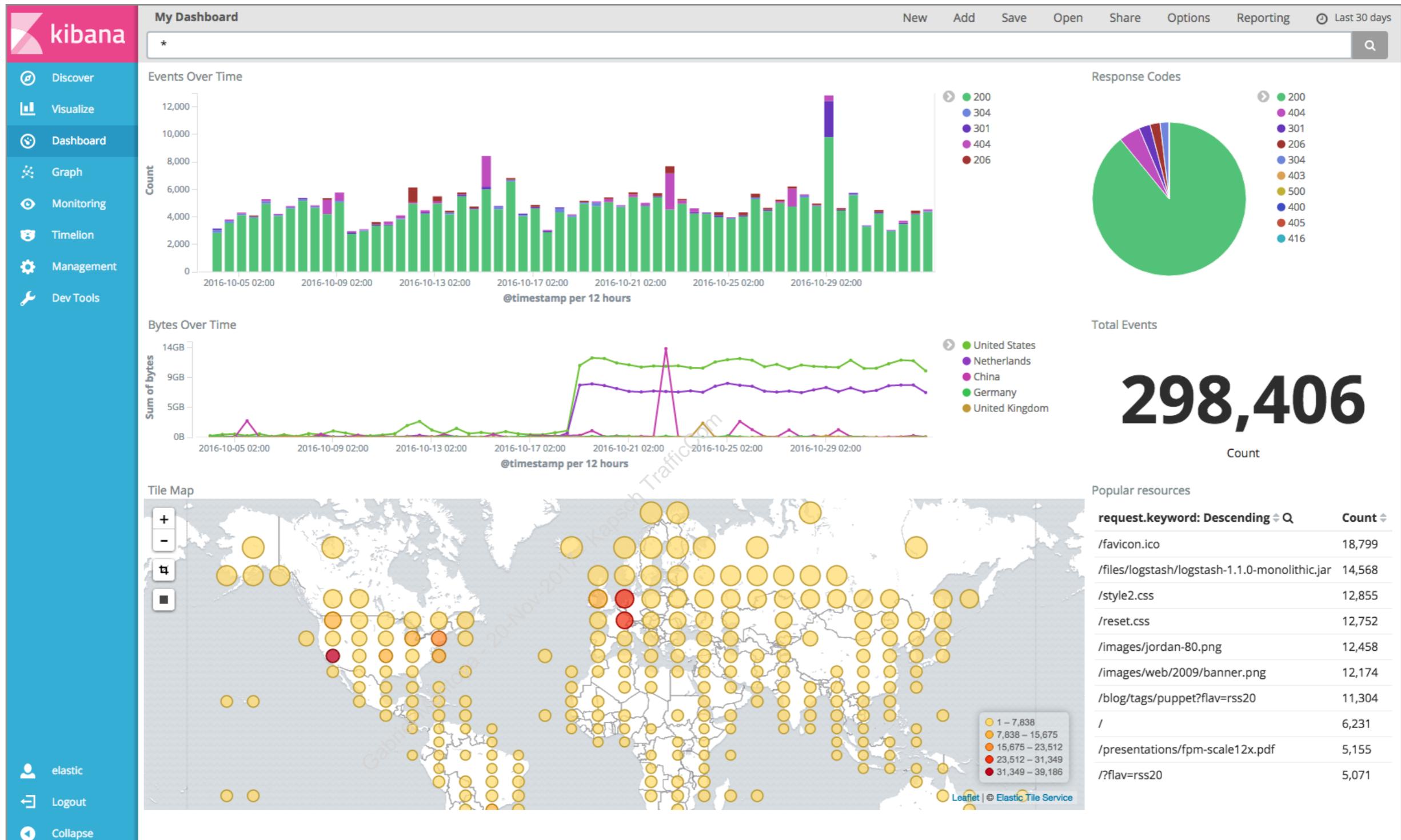
▼ Star Rating

<input type="checkbox"/> No preference	454
<input type="checkbox"/> 1 star	16
<input type="checkbox"/> 2 stars	80
<input type="checkbox"/> 3 stars	72
<input type="checkbox"/> 4 stars	55
<input type="checkbox"/> 5 stars	9
<input type="checkbox"/> Unrated	222

▼ Accommodation Type

<input type="checkbox"/> Apartments	186
<input type="checkbox"/> Hotels	175
<input type="checkbox"/> Motels	39
<input type="checkbox"/> Vacation Homes	21
<input type="checkbox"/> Hostels	18
<input type="checkbox"/> Inns	9
<input type="checkbox"/> Bed and Breakfasts	6

Aggregations in Kibana Visualizations



Console

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Console: IDE for Elasticsearch

Dev Tools

Console

```
1 GET _cat/indices
2
3 GET _search
4 - {
5   "query": {
6     "match_all": {}
7   }
8 }
9
10 GET fb_world_population/_search
11
12 GET fb_world_population/_search?q=country
13
14 GET ls_world_population/_search
15
16 DELETE ls_world_population
17
18 DELETE world_population
19
20 GET world_population/_search
21 {
22   "size": 0,
23   "aggs": {
24     "NAME": {
25       "geohash_grid": {
26         "field": "geo_location",
27         "precision": 3
28       },
29       "aggs": {
30         "NAME": {
31           "top_hits": {
32             "size": 10
33           }
34         }
35       }
36     }
37   }
38 }
39
```

took: 8,
timed_out: false,
_shards: {
 total: 12,
 successful: 12,
 skipped: 0,
 failed: 0
},
hits: {
 total: 748,
 max_score: 1,
 hits: [
 {_index: ".kibana",
 _type: "doc",
 _id: "config:6.0.0-beta1",
 _score: 1,
 _source: {
 type: "config",
 config: {
 buildNum: 15799,
 defaultIndex: "d3f12b60-89a0-11e7-bc55-a5c147502291"
 }
 }
 },
 {_index: ".kibana",
 _type: "doc",
 _id: "visualization:61331190-89a7-11e7-bc55-a5c147502291",
 _score: 1,
 _source: {
 type: "visualization",
 visualization: {
 title: "World Population - 2016 Pie",
 visState: ""{ "title": "World Population - 2016 Pie", "type": "pie",
 position": "right", "isDonut": false},
 aggs: [{"id": "1", "enabled": true, "type": "sum", "enabled": true, "type": "terms", "schema": "segment", "params": {"field": "country.key", "uiStateJSON": "f1"}
 }
 }
 }
]
}

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- To enable a new module to Metricbeat, simply rename the config file in the **modules.d** folder and restart **metricbeat**
- Set **reload.enable: true** to enable configuration changes without restarting Beats
- The MySQL module collects metrics, like the total number of bytes sent and received from all clients, and the total number of times each query executed since startup
- An Elasticsearch cluster is composed by one or more nodes
- Elasticsearch organizes data into indices that are internally split into shards, which are distributed across the nodes
- Console allows users to easily interact with Elasticsearch



Quiz

1. **True or False:** To monitor a new Metricbeat module you need to rename the config file and restart metricbeat.
2. **True or False:** Metricbeat can dynamically reload modules from config files when there are changes.
3. **True or False:** Every document indexed in Elasticsearch belongs to an index.
4. How does Elasticsearch distribute your data within an index?
5. **True or False:** Kibana uses the Elasticsearch aggregation feature to create Visualizations.

Gabriel Montoya / 11 Nov 2017 - Kapsch TrafficCom

Lab 3

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 4

Ingesting File Data

Gabriel Montoya - 20-Nov-2017 - KapsuleTrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

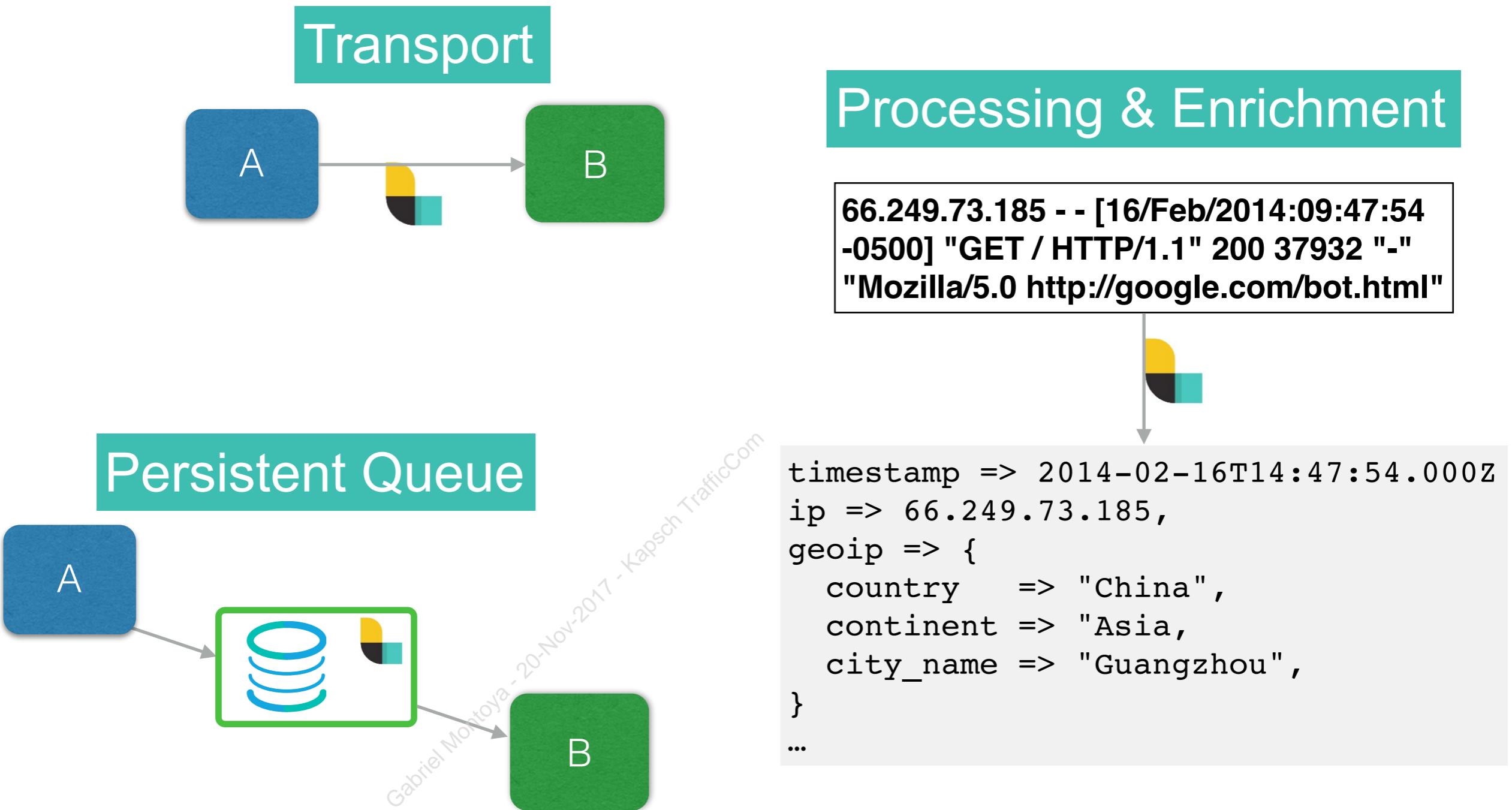
- Logstash
- Reading Events from Files
- Logstash File Input Plugin
- Filebeat
- Resilience
- Multi-line Events
- Logstash or Filebeat?

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logstash

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Why Logstash?



Documentation: <https://www.elastic.co/guide/en/logstash/current/index.html>

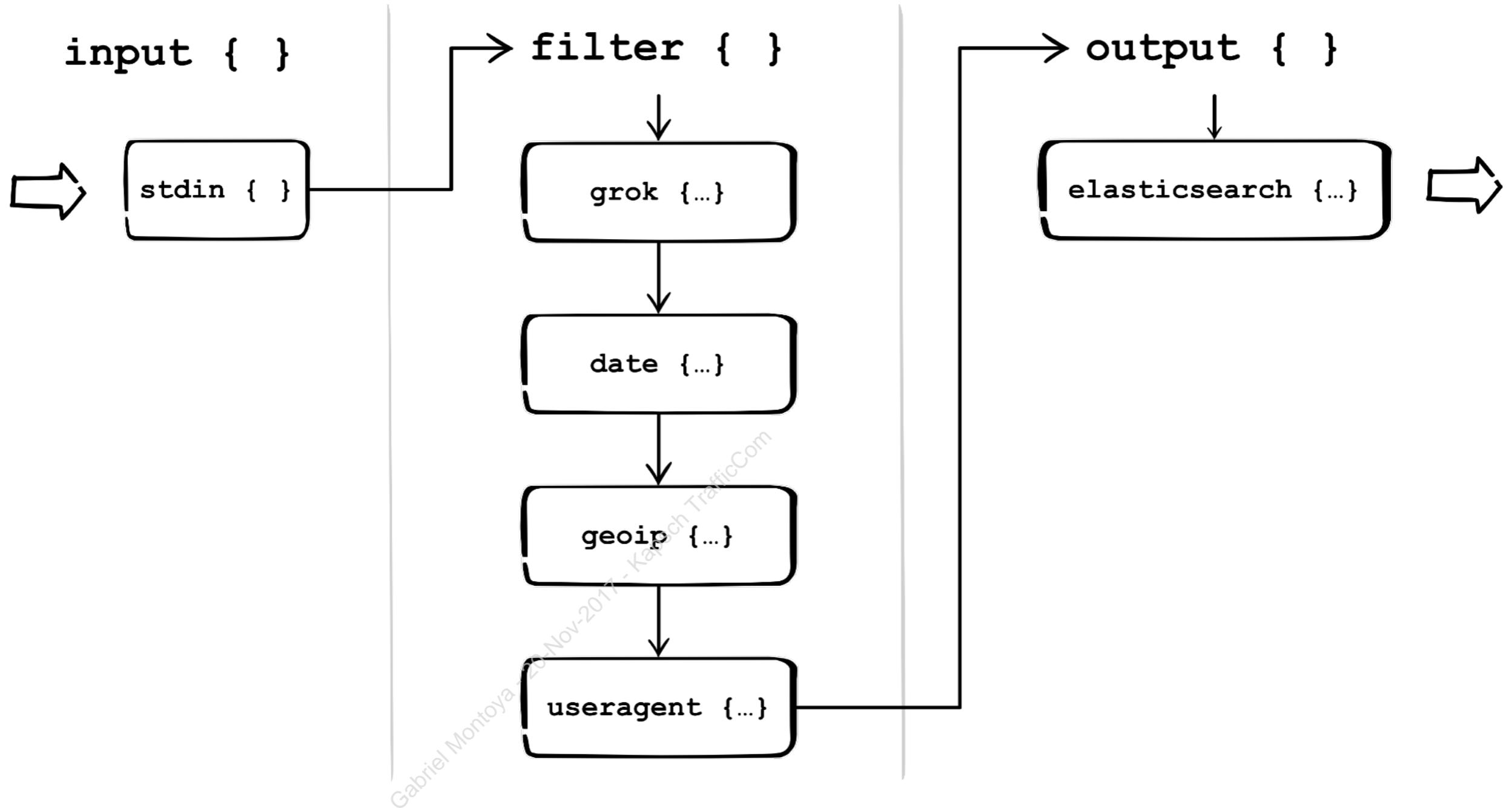


Logstash Terms and Definitions

- Event
 - any piece of data that has a timestamp and some data
- Plugins
 - working units that handle events

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logstash Pipeline



Logstash Plugin Ecosystem



Plugins

- Inputs
 - capture a data stream and turn it into events
- Filters
 - process events
- Outputs
 - send events to destinations
- Codec (operate as part of an Input or Output filter)
 - changes the data representation of an event

Input Configuration

```
input {  
  stdin {}  
}
```

```
input {  
  beats {  
    port => 5044  
  }  
}
```

Diagram annotations:

- A teal box labeled "plugin name" has an arrow pointing to the word "beats".
- A teal box labeled "value" has an arrow pointing to the value "5044".
- A teal box labeled "setting" has an arrow pointing to the brace under "beats".

```
input {  
  twitter {  
    consumer_key => "consumer_key"  
    consumer_secret => "consumer_secret"  
    oauth_token => "access_token"  
    oauth_token_secret => "access_token_secret"  
    keywords => [ "elastic", "elastic stack" ]  
  }  
}
```



Filter Configuration

```
filter {}
```

The diagram illustrates the structure of filter configuration code. It shows two examples of filter blocks with annotations:

- Annotation Labels:**
 - plugin name:** A teal box pointing to the word "filter" in the first example.
 - value:** A teal box pointing to the value "credit_card" in the "remove_field" setting of the first example.
 - setting:** A teal box pointing to the "remove_field" setting in the first example.
- Code Examples:**
 - Example 1:** A single-line filter block: `filter {}`
 - Example 2:** A multi-line filter block with a "mutate" plugin:

```
filter {
  mutate {
    remove_field => "credit_card"
  }
}
```
 - Example 3:** A multi-line filter block with a "csv" plugin:

```
filter {
  csv {
    source => "message"
    columns => [ "country", "country_code", "population" ]
    separator => ";"
  }
}
```

Gabriel Montoya - Nov-2017 - Kapsch TrafficCom

```
filter {}
```

```
filter {
  mutate {
    remove_field => "credit_card"
  }
}
```

```
filter {
  csv {
    source => "message"
    columns => [ "country", "country_code", "population" ]
    separator => ";"
  }
}
```



Output Configuration

```
output {  
  stdout {}  
}
```

```
output {  
  elasticsearch {  
    hosts => [ "my_host1", "my_host2" ]  
  }  
}
```

Diagram illustrating the structure of the output configuration:

- setting**: A green box pointing to the opening brace of the `elasticsearch` block.
- plugin name**: A green box pointing to the word `elasticsearch`.
- value**: A green box pointing to the array of host names `["my_host1", "my_host2"]`.

```
output {  
  pagerduty {  
    service_key => "my_pagerduty_api_key"  
    details => {  
      "message" => "Error in production!"  
    }  
  }  
}
```



Simple Configuration

```
input {  
  stdin {}  
}  
  
filter {}  
  
output {  
  stdout { codec => rubydebug }  
}
```

Good to see if logstash executes without any problems in the machine

```
input {  
  stdin {}  
}  
  
filter {}  
  
output {  
  elasticsearch {}  
}
```

Good to see if logstash can write to Elasticsearch.

hosts => ['localhost:9200']
index => 'logstash-yyyy.MM.dd'



More Advanced Configuration

```
input {  
  beats { port => 5044 }  
}  
  
filter {  
  csv {  
    source => "message"  
    columns => [ "country", "country_code", "dummy1", "population" ]  
  }  
  
  if [country] == "Country" { drop {} }  
  
  mutate { remove_field => "dummy1" }  
}  
  
output {  
  stdout { codec => dots }  
  
  elasticsearch {  
    hosts => [ "my_host1", "my_host2", "my_host3" ]  
    index => "ls_world_population"  
  }  
}
```

receives events from Beats

extracts 4 fields from the csv line

drops the header line

removes useless field

prints a dot for every processed event

outputs to a specific Elasticsearch index using round-robin



Reading Events from Files

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Reading Events from Files



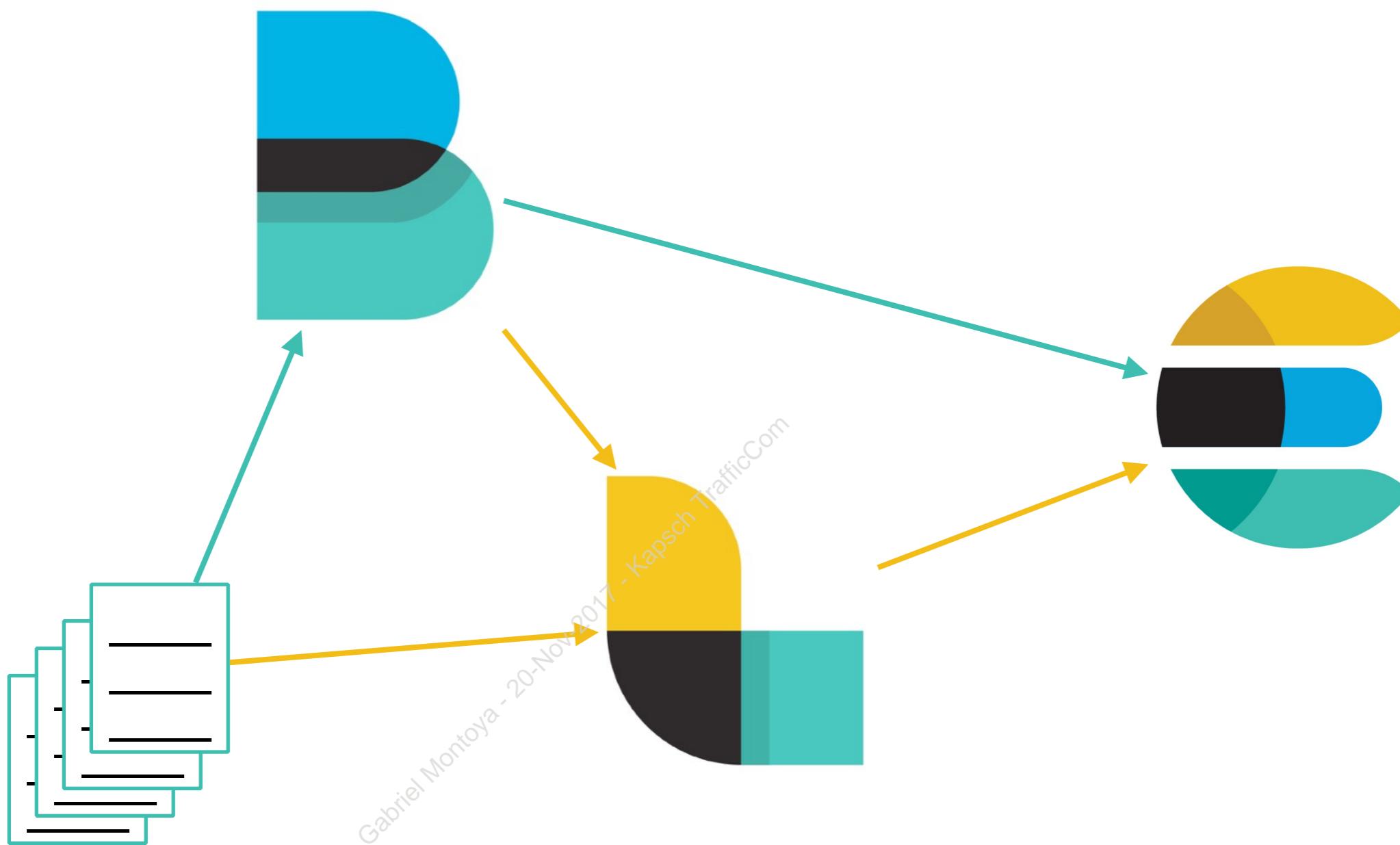
Reading Events from Files

- Why?
 - applications and services often log to files
 - different formats to store data (CSV, XML, ...)
- Steps
 - **read line event (or multi-line)**
 - parse it
 - enrich it
 - send to destination

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Reading Events from Files



Logstash File Input Plugin

Gabriel Montoya - 20-Nov-2017 - KapselTraffic.Com



File Input Plugin

- Stream events from files
- Tail files in a manner similar to `tail -0F`
- Optionally, read files from the beginning
- Track changing files and emit new content when appended

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



File Input Plugin Example

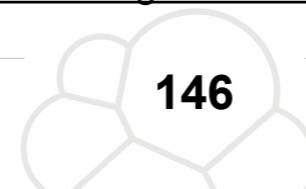
```
input {  
  file {  
    path => "/var/log/apache/access_log"  
  }  
}
```

read new lines from access_log (tail)

```
input {  
  file {  
    path => "/var/log/apache/access_log"  
    stat_interval => 5          # default: 1  
    close_older     => 600      # default: 3600  
  }  
}
```

- check every 5 seconds
- close any files that were last read more than 600 seconds ago

<https://www.elastic.co/guide/en/logstash/current/plugins-inputs-file.html>



File Input Plugin Example

```
input {  
  file {  
    path => "/home/ubuntu/datasets/*"  
    start_position => "beginning"          # default: "end"  
    sincedb_path => "/dev/null"           # default: "$HOME/.sincedb*"  
    ignore_older => 86400                 # seconds (ignore > 24h)  
  }  
}
```

- read all lines from all files in datasets
- don't track current position of files
- ignore files older than 24h

```
input {  
  file {  
    path => "/var/log/apache/*"  
    exclude => "*.gz"  
  }  
}
```

tail all files in apache, but the ones that end in .gz

<https://www.elastic.co/guide/en/logstash/current/plugins-inputs-file.html>

Filebeat

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Filebeat

- Installed as an agent on all servers from which you want to gather logs
- Monitors log directories or specific log files
- Tails these files
- Ship log entries to:
 - Elasticsearch
 - Logstash
 - Kafka
 - Redis

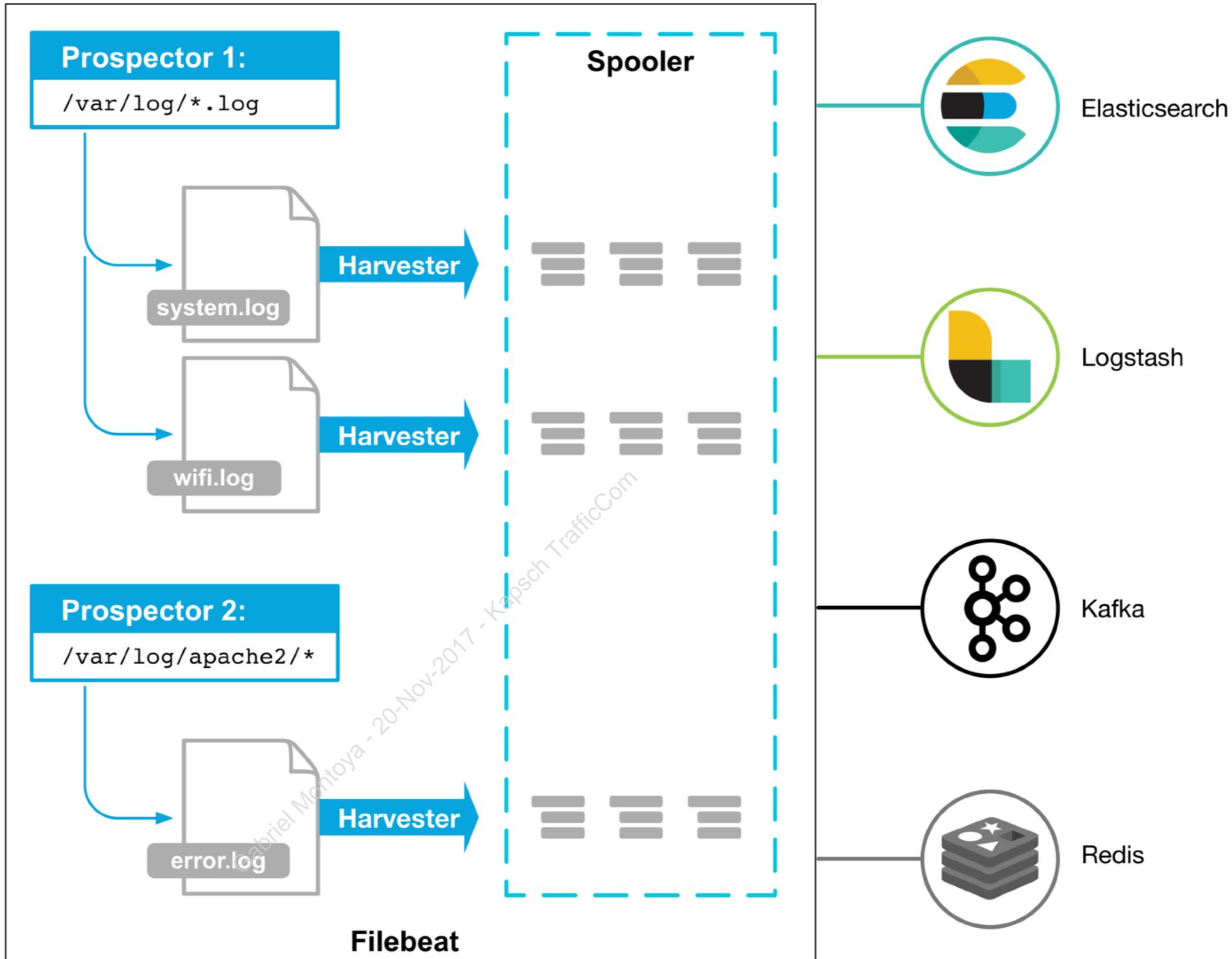
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Why Filebeat?

- Lightweight executable
 - no JVM
 - Filebeat runs as a binary so no runtime is needed
 - easy to deploy across many architectures
 - scale your data shippers independently of your processors

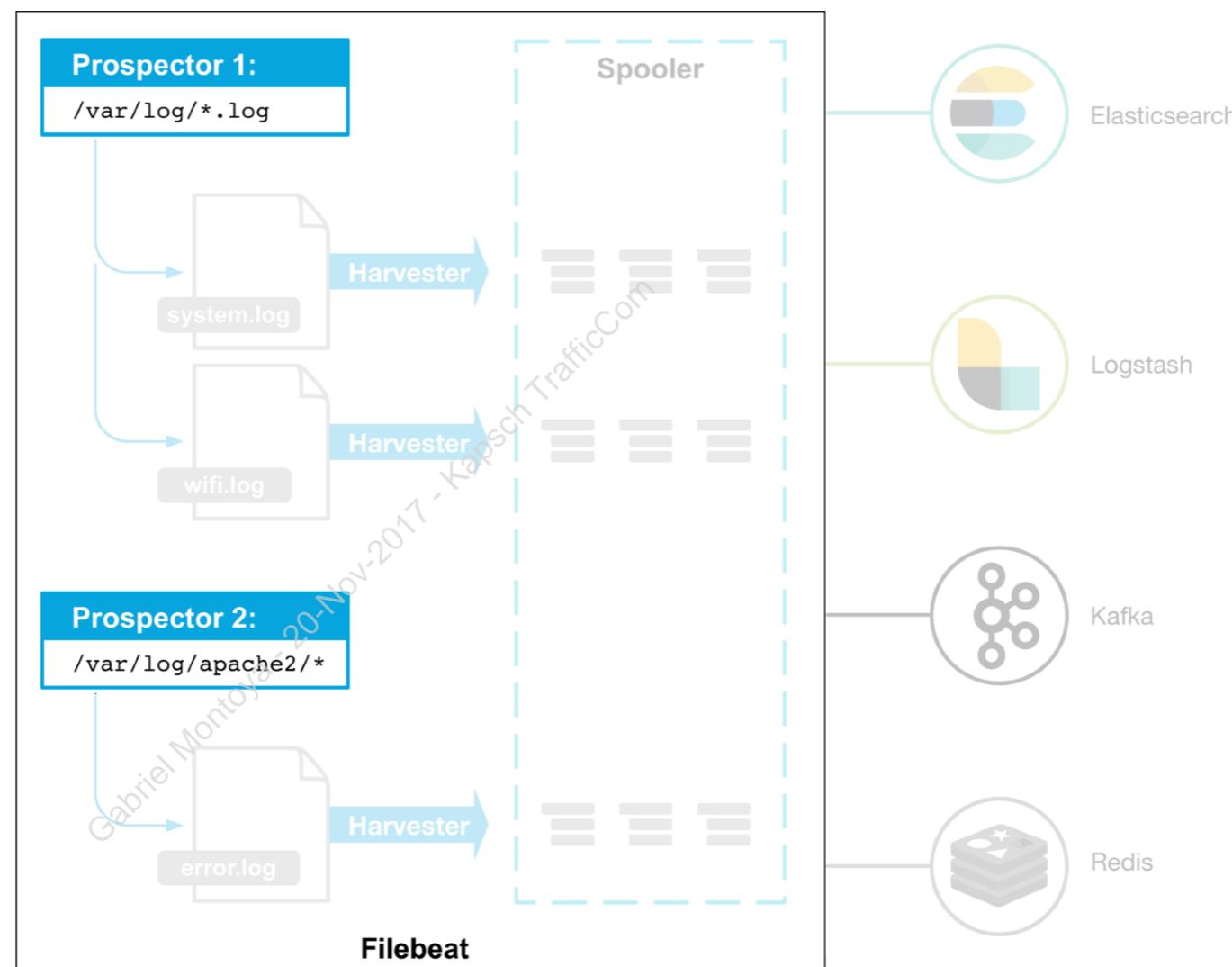
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Architecture



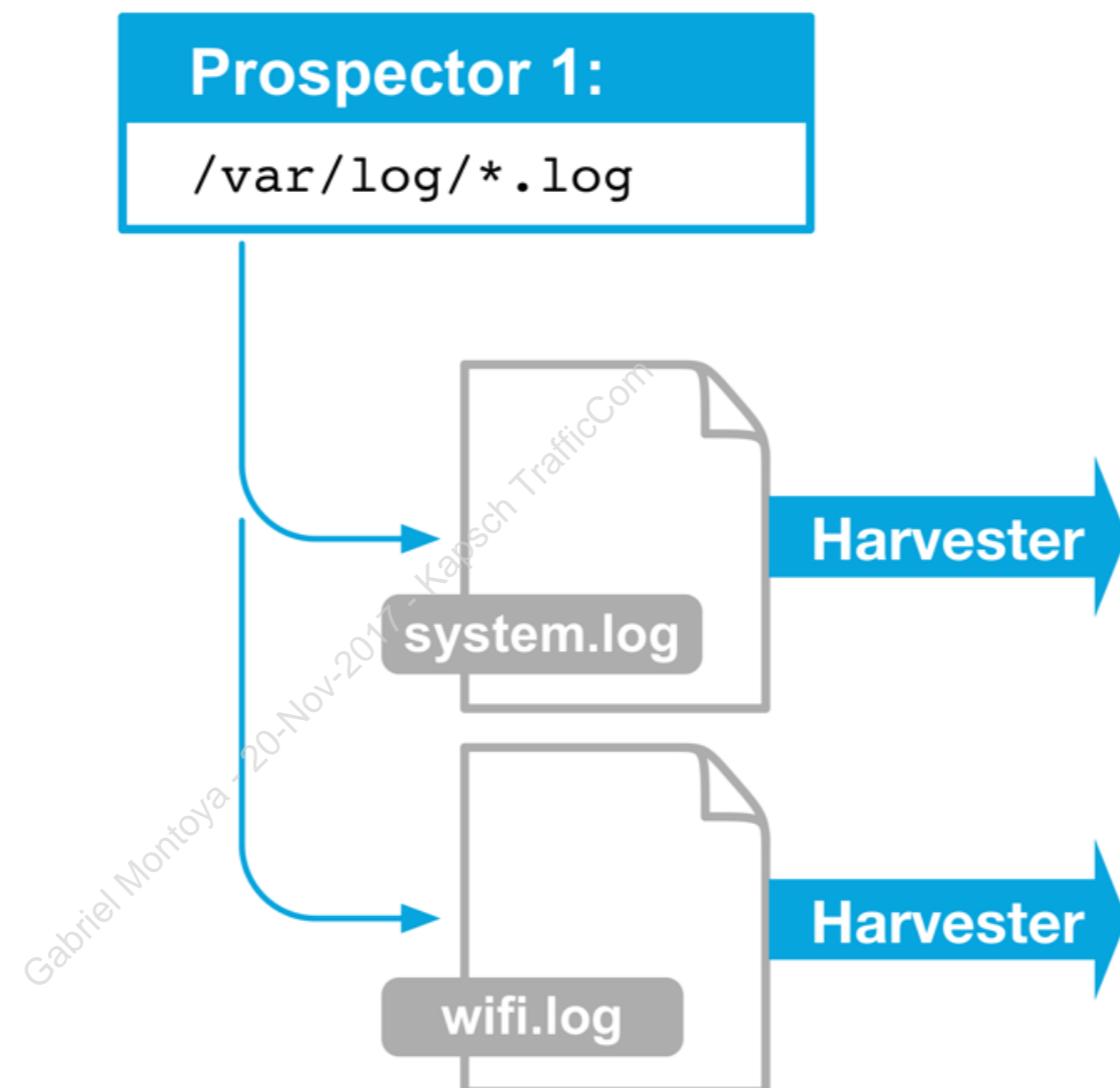
Prospectors

- Each instance of Filebeat can be configured with one or more **prospectors**
- Each prospector can be configured to monitor one or more file paths



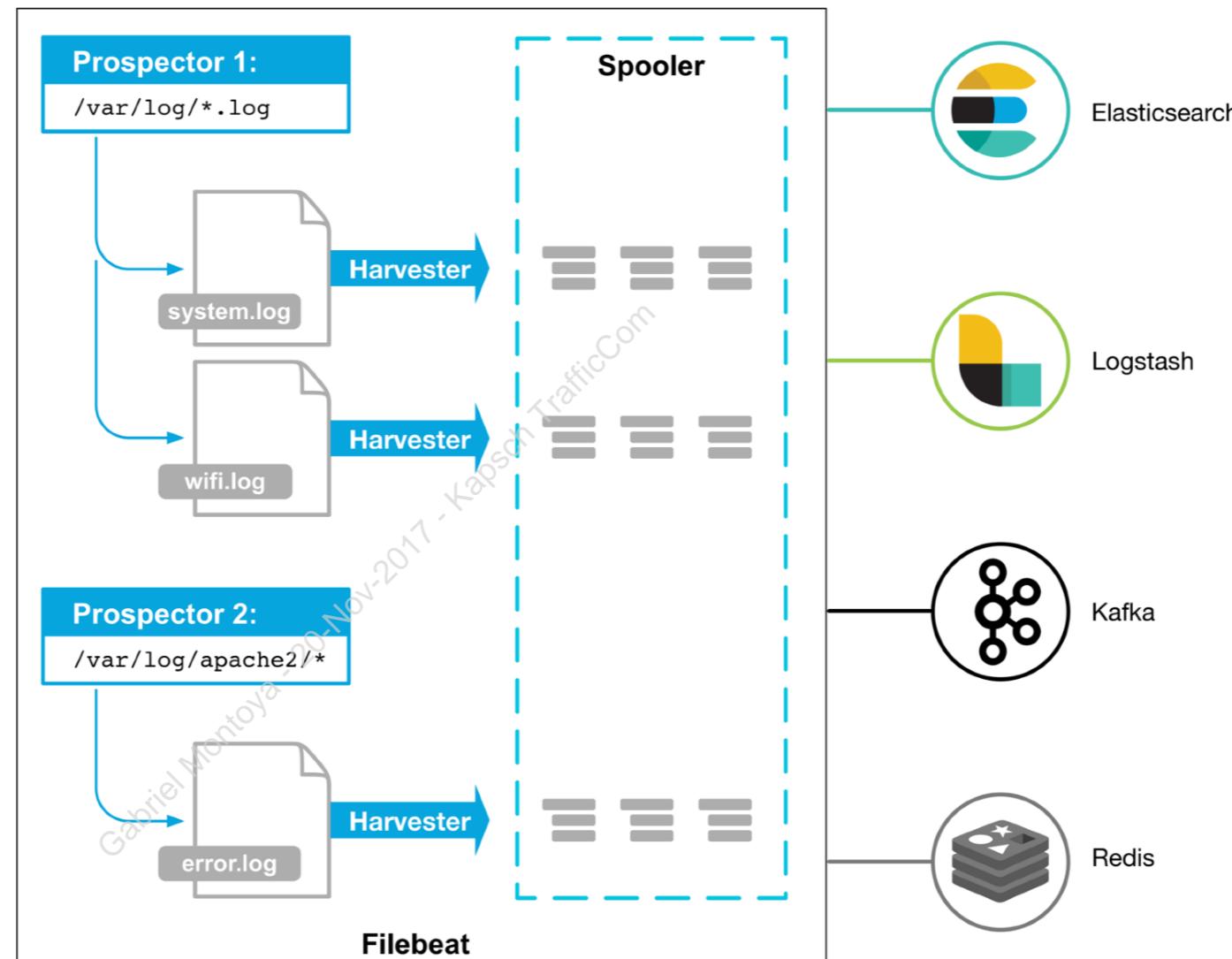
Harvesters

- For each file that a prospector locates, Filebeat starts a **harvester**
- Each harvester reads a single file for new content



Spooler

- Each harvester sends the events to the **spooler**
- The spooler aggregates the events and sends the aggregated data to the output that has been configured



Filebeat Configuration

- Default configuration file is **filebeat.yml**
- Sample configuration:

```
filebeat.prospectors:  
- input_type: log  
  paths:  
    - /var/log/*.log  
  
output.elasticsearch:  
  hosts: ["localhost:9200"]
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Prospector Include/Exclude

- Besides processors, Filebeat allows you to use regular expression to filter exported data:

```
filebeat.prospectors: export any lines that start with "ERR" or "WARN"
- paths:
  - /var/log/myapp/*.log
include_lines: [ '^ERR', '^WARN' ]
```

```
filebeat.prospectors: drop any lines that start with "DBG"
- paths:
  - /var/log/myapp/*.log
exclude_lines: [ '^DBG' ]
```

```
filebeat.prospectors: ignore all the files that the name ends in .gz
- paths:
  - /var/log/myapp/*
exclude_files: [ '\.gz$' ]
```

Resilience

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Checking for New Files and Lines

- How often it checks for **new files**
 - LS: `discover_interval` (default: 15 seconds)
 - FB: `scan_frequency` (default: 10 seconds)
- How often it check for **new lines** in a file
 - LS: `stat_interval` (default: 1 second)
 - FB: implements an exponential backoff
 - `backoff` (initial wait time after EOF, default: 1 second)
 - `backoff_factor` (factor used to multiply waiting time each time no new entries are found, default: 2 seconds)
 - `max_backoff` (maximum time Filebeat waits before checking a file again after EOF is reached, default: 10 seconds)

Recovering

- What happens if you stop Filebeat or Logstash?
- Both Filebeat and Logstash use a file to track progress, which contains:
 - current log files being parsed
 - offset into each log file
 - *inode* and *device* information
- At least-once delivery
- If you want to import the same data again, delete the file. But be careful deleting it in Production.

Gabriel Montoya / ID Nov 2017 - Kapsch TrafficCom



Recovering

- Logstash uses a `sincedb` file:
 - placed in the home directory of the user running Logstash
 - use `sincedb_path` to define another location
 - use `sincedb_write_interval` to control how often Logstash writes to the synced file (default: 15 seconds)
- Filebeat uses a registry file:
 - `data/registry` for `.tar.gz` and `.tgz` archives
 - `/var/lib/filebeat/registry` for DEB and RPM packages
 - `C:\ProgramData\filebeat\registry` for the Windows zip file
 - use `filebeat.registry_file` to define another location



Multi-line Events

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Multiline Events

- Multiline events must be collected at the source, otherwise order is not guaranteed
- Uses regular expressions to define start or end line of the event

```
1  Caused by: java.lang.ExceptionInInitializerError
2      at org.elasticsearch.common.logging.DeprecationLogger.<clinit>(DeprecationLogger.java:138)
3      at org.elasticsearch.common.xcontent.support.AbstractXContentParser.<init>(AbstractXContentParser.java:57)
4      at org.elasticsearch.common.xcontent.json.JsonXContentParser.<init>(JsonXContentParser.java:44)
5      at org.elasticsearch.common.xcontent.json.JsonXContent.createParser(JsonXContent.java:103)
6      at org.elasticsearch.common.settings.Setting.parseableStringToList(Setting.java:832)
7      at org.elasticsearch.common.settings.Setting.lambda$listSetting$27(Setting.java:786)
8      at org.elasticsearch.common.settings.Setting.listSetting(Setting.java:791)
9      at org.elasticsearch.common.settings.Setting.listSetting(Setting.java:786)
10     at org.elasticsearch.common.network.NetworkService.<clinit>(NetworkService.java:50)
11     at org.elasticsearch.client.transport.TransportClient.newPluginService(TransportClient.java:98)
12
13
```

How do you capture all these lines as a single event?

Multiline Event Settings

- pattern
 - regular expression pattern to match
- negate
 - whether the pattern is negated (default false)
- match/what
 - how to combine matching lines into an event
 - LS: next, previous
 - FB: before, after

The **after** setting is equivalent to **previous** in Logstash, and **before** to **next**.

Logstash Multiline

- Multiline **codec**
- Not bundled by default in versions older than 6.0:
`bin/logstash-plugin install logstash-codec-multiline`
- Added to the input plugin

```
input {  
  file {  
    path => "/var/log/*.log"  
    codec => multiline {  
      pattern => "^["  
      negate => true  
      what => "previous"  
    }  
  }  
}
```

any line not starting with [belongs to the previous line that does

Filebeat Multiline

- Defined in the prospectors section of config

```
filebeat.prospectors:  
- type: log  
  enabled: false  
  paths:  
    - /var/log/*.log
```

any line not starting with [belongs to the previous line that does

```
multiline.pattern: '^\\['  
multiline.negate: true  
multiline.match: after
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Multiline Event Settings

negate	what match	result
FALSE	previous after	<i>Consecutive lines that match the pattern are appended to the previous line that doesn't match.</i>
FALSE	next before	<i>Consecutive lines that match the pattern are prepended to the next line that doesn't match.</i>
TRUE	previous after	<i>Consecutive lines that don't match the pattern are appended to the previous line that does match.</i>
TRUE	next before	<i>Consecutive lines that don't match the pattern are prepended to the next line that does match.</i>

↑ ↑

Logstash Filebeat



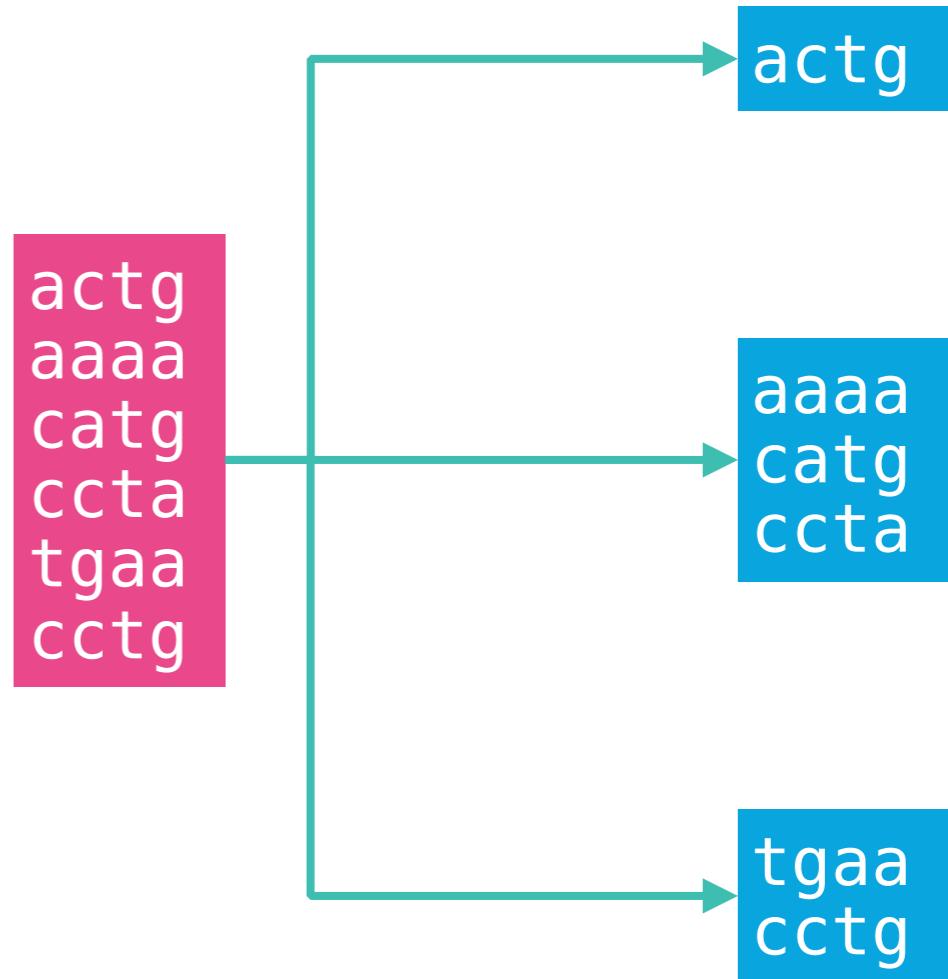
Multiline Event

- Consecutive lines that match the pattern are appended to the previous line that doesn't match

```
codec => multiline {  
    pattern => "^c"  
    negate => "false"  
    what => "previous"  
}
```

```
filebeat.prospectors:  
    multiline.pattern: '^c'  
    multiline.negate: false  
    multiline.match: after
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Multiline Event

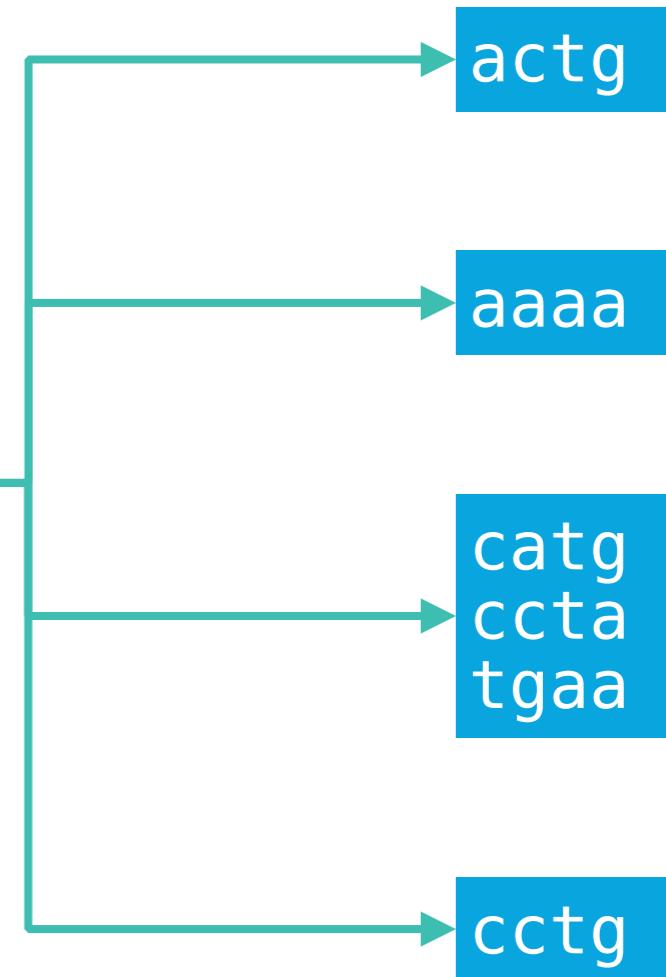
- Consecutive lines that match the pattern are prepended to the next line that doesn't match

```
codec => multiline {  
    pattern => "^c"  
    negate => "false"  
    what => "next"  
}
```

```
filebeat.prospectors:  
    multiline.pattern: '^c'  
    multiline.negate: false  
    multiline.match: before
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

actg
aaaa
catg
ccta
tgaa
cctg



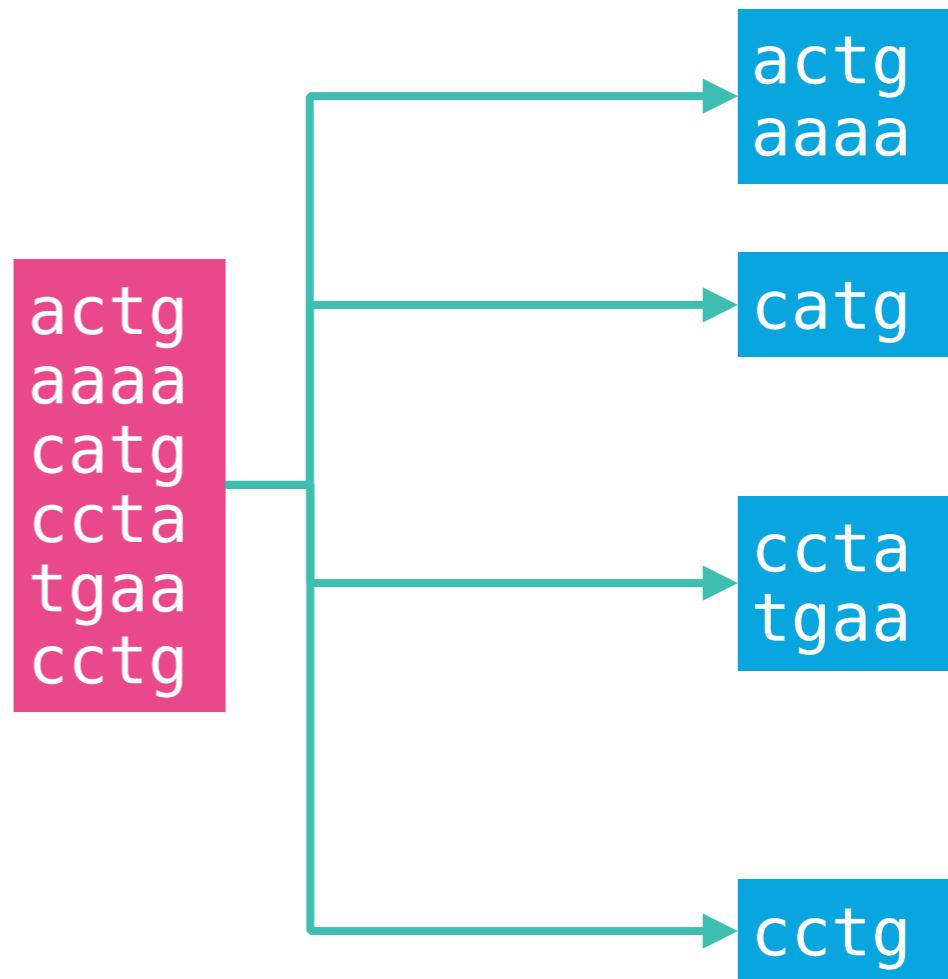
Multiline Event

- Consecutive lines that don't match the pattern are appended to the previous line that does match

```
codec => multiline {  
    pattern => "^c"  
    negate => "true"  
    what => "previous"  
}
```

```
filebeat.prospectors:  
    multiline.pattern: '^c'  
    multiline.negate: true  
    multiline.match: after
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



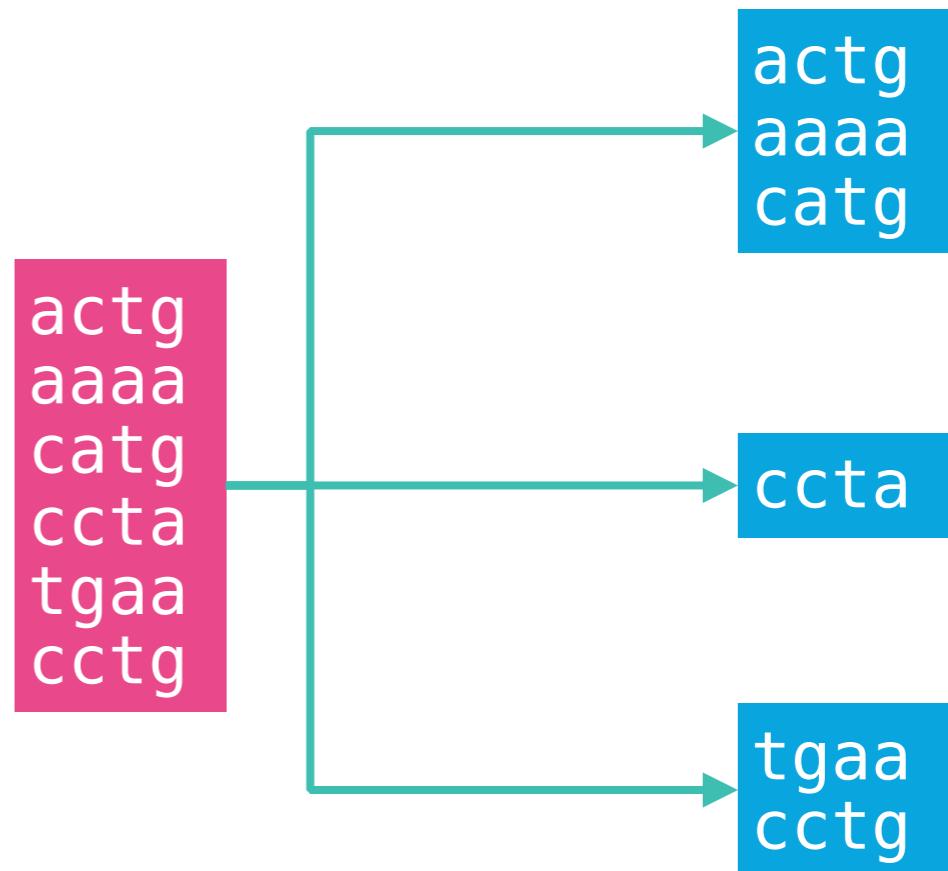
Multiline Event

- Consecutive lines that don't match the pattern are prepended to the next line that does match

```
codec => multiline {  
    pattern => "^c"  
    negate => "true"  
    what => "next"  
}
```

```
filebeat.prospectors:  
    multiline.pattern: '^c'  
    multiline.negate: true  
    multiline.match: before
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Multiline Event

```
input {  
    file {  
        path => "/home/ubuntu/my_java_app.log"  
        codec => multiline {  
            pattern => "^\\s"  
            negate => false  
            what => "previous"  
        }  
    }  
}
```

any line starting with whitespace belongs to the previous line that does not

```
filebeat.prospectors:  
- type: log  
  enabled: false  
  paths:  
    - /home/ubuntu/my_java_app.log
```

```
multiline.pattern: '^\\s'  
multiline.negate: false  
multiline.match: after
```

Logstash or Filebeat?

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Logstash or Filebeat?

- It Depends!
- Application or Server side component?
- Data processing & enrichment?
- You could use both together.
- More about that later!

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- Logstash is an open source data collection engine with real-time pipelining capabilities.
- A Logstash pipeline is composed by inputs->filters->outputs
- Logstash codecs can change input and output data representation.
- Multiple inputs, filters, and outputs available.
- To ingest data from text files, you can use the Logstash file input filter or Filebeat.
- Logstash and Filebeat track progress and can gracefully recover from restarts or crashes.
- Multiline events should be collected at the source.



Quiz

1. What are the 4 plugin types in Logstash?
2. **True or False:** You can have multiple inputs and outputs in the same logstash configuration.
3. **True or False:** If you run logstash with a configuration file without filters, it will return an error and fail to execute.
4. How does Filebeat not miss events after a crash?
5. **True or False:** When using Logstash to read data from files in production, you should use `since_db_path => "/dev/null"`.
6. What will be the result of the following multi-line config?

```
multiline.pattern: '^\\[[0-9]{4}-[0-9]{2}-[0-9]{2}\\]'
```

```
multiline.negate: true
```

```
multiline.match: after
```

Lab 4

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 5

Data Processing

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Data Administration
- Logstash Event Data Access
- drop Filter
- mutate Filter
- csv Filter
- dissect Filter
- translate Filter
- Logstash Config File Reload

Gabriel Montoya / 10 Nov 2017 - Kapsch TrafficCom

Data Administration

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Dataset

Timestamp	Favorite Prog	Favorite Marvel	Favorite OS	Current company industry	Country	Length of service	How long	Current position	What	What	What	What	Family
2017/02/17 5:26:18	Java	spider-man	OSX	Software Products;Web	US	11+	Engineer	1	3	4	4	El	
2017/02/17 5:28:41	HTML	Thor	OSX	Software Products	Netherlands	11+	Manager	0	1	1	2	Da	
2017/02/17 5:28:47	Javascript	don't have one	OSX	Software Products	Canada	2-5	Developer	0	0	2	3	ES	
2017/02/17 5:29:30	Java		OSX	Software Products	Britain	11+			1	2	4	2	Mi
2017/02/17 5:30:27	Python	Dr. Doom											
2017/02/17 5:30:31	Go												
2017/02/17 5:30:52	Java	Iron Man											
2017/02/17 5:33:10	Fortran	Wolverine											
2017/02/17 5:42:14	Python	Ghost Rider											

```
1 Caused by: java.lang.ExceptionInInitial
2   at org.elasticsearch.common.logging
3   at org.elasticsearch.common.xconver
4   at org.elasticsearch.common.xconver
5   at org.elasticsearch.common.xconver
6   at org.elasticsearch.common.setting
7   at org.elasticsearch.common.setting
8   at org.elasticsearch.common.setting
9   at org.elasticsearch.common.setting
10  at org.elasticsearch.common.network
11  at org.elasticsearch.client.transpo
12  at org.elasticsearch.client.transpo<field name="Country or Area" key="ABW">Aruba</field>
13  at org.elasticsearch.client.transport.TransportClient.<init>(TransportClient.java:268)
14  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:125)
15  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:111)
16  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:101)
17  at com.regiocom.bpo.rcease.util.TransportClientFactory.configureClients(TransportClientFactory.java:81)
```



Questions

- What is the most popular programming language in StackOverflow?
- How many countries account for 50% of the world population?
- How many users signed up this week?
- Why user (id: 536) cannot login to the system?

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Unstructured vs. Structured Data

```
"Aruba","ABW","54211","55438","56225","56695","57032"
```

```
{  
  "message": "'Aruba','ABW','54211','55438','56225','56695','57032','57360'"  
}
```

```
{  
  "country": "Aruba",  
  "country_code": "ABW",  
  "population": {  
    "1960": 54211,  
    "1961": 55438,  
    "1962": 56225,  
    "1963": 56695,  
    "1964": 57032,  
    "1965": 57360  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Unstructured vs. Structured Data

```
"83.149.9.216 -- [01/Jun/2015:21:13:45 -0200] \"GET /presentations/  
logstash-monitorama-2013/plugin/notes/notes.js HTTP/1.1\" 200 2892 \"http://  
semicomplete.com/presentations/logstash-monitorama-2013/\\""
```

```
{  
    "clientip"      => "83.149.9.216",  
    "ident"         => "-",  
    "auth"          => "-",  
    "timestamp"     => "01/Jun/2015:21:13:45 -0200",  
    "verb"          => "GET",  
    "request"       => "/presentations/logstash-monitorama-2013/plugin/notes/  
notes.js",  
    "httpversion"   => "1.1",  
    "response"      => 200,  
    "bytes"         => 2892,  
    "referrer"      => "\"http://semicomplete.com/presentations/logstash-  
monitorama-2013/\\""  
}
```



Document Store vs. Relational Databases

```
{  
  "id": "342",  
  "make:: Ford",  
  "m  
} {  
  "p  
    "id": "567",  
    "make:: BMW",  
    "m  
} {  
  "p  
    "id": "845",  
    "make:: Ford",  
    "m  
} {  
  "p  
    "id": "1024",  
    "make:: Honda",  
    "model": "Civic",  
    "price": 50,000  
}
```

id	make	model	price
342	Ford	Fiesta	20,000
567	BMW	M140i	110,00
845	Ford	Focus	30,000
1024	Honda	Civic	50,000

Structure does not need to be defined.
Document defines its own structure.

Structure defined beforehand.
Database defines the document structure.



Document Store vs. Relational Databases

```
{  
  "id": "342",  
  "make": {  
    "name": "Ford",  
    "year": "1903",  
    "country": "US"  
  },  
  "model": "Fiesta",  
  "price": 20,000  
}
```

id	name	year	country
11	Ford	1903	US
8	BMW	1916	DE
5	Honda	1948	JP

id	make	model	price
342	11	Fiesta	20,000
567	8	M140i	110,00
845	11	Focus	30,000
1024	5	Civic	50,000

Usually the document contains all the data in denormalized way.

Usually data is normalized and there are relations between tables.



Index Time vs. Query Time

- It depends! There is always a trade off...
- You need to find the right balance.
- If an expensive query is executed very frequently, you might want to pre calculate part of that response at index time, so you don't have to pay at every request.
- If you rarely query a specific field, you might want to calculate it at query time and save disk space and index processing.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Logstash Event Data Access

Gabriel Montoya - 20-Nov-2017 - KapschInterCom



Logstash Field Reference

- Every event has *properties* (also called *fields*)
- It is often useful to be able to access field values:
 - want to send error messages to a different output
 - want to apply a different filters to different type of events
 - don't want to be alerted for every message
 - create a new field combining other fields
 - set plugin configuration using an existing field

Gabriel Montoya - 20-Nov-2017 - Kappa Traffic Control

Logstash Field Reference

- Consider the event below:

```
{  
  "purchase_id": "15310",  
  "buyer": {  
    "first_name": "Noah"  
    "last_name": "Lotus"  
  }  
  "car": {  
    "make": {  
      "name": "Ford",  
      "year": "1903",  
      "country": "US"  
    },  
    "model": "Fiesta",  
    "price": 20,000  
  }  
  "price": 19,000  
}
```

Gabriel Montoya - 20-May-2017 - Kapsule Traffic Control

Field References

- For the given key/value pair:

```
"price": 19,000
```

- The way to reference this field would be:
 - **[price]**
- Which would have a value of:
 - 19,000

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Field References

- For the given key/value pairs:

```
"buyer": {  
    "first_name": "Noah"  
    "last_name": "Lotus"  
}
```

- The way to reference this nested field would be:
 - **[buyer][first_name]**
- Which would have a value of:
 - "Noah"

Gabriel Montoya - 20-Nov-2017 - Kapsch Trafigura



Field References

- String interpolation for field references
 - What if we want to include a field's value inside a string?
 - What if we want to include a field's value inside another field?

```
filter {  
  mutate {  
    add_field => {  
      "full_name" => "%{[buyer][first_name]} %{[buyer][last_name]}"  
    } } }
```

- If the reference field is not an inner field you can omit the '[]' in the string interpolation.

```
filter {  
  mutate {  
    add_field => {  
      "full_name" => "%{first_name} %{last_name}"  
    } } }
```



Conditionals

- What's an EXPRESSION?
 - comparison tests
 - is 4 greater than 5?
 - is *fieldA* equal to "ERROR"?
 - boolean logic
 - is (4 greater than 5) AND (*fieldA* equal to "ERROR")?

```
if EXPRESSION {  
    ...  
} else if EXPRESSION {  
    ...  
} else {  
    ...  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Conditionals

- Comparison Operators:
 - equality: ==, !=, <, >, <=, >=
 - regexp: =~, !~
 - inclusion: in, not in
- Boolean Operators:
 - and, or, nand, xor
- Unary operators:
 - ! (not)

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Conditionals

- Filter Example:

```
filter {  
  if [price] <= 20,000 {  
    mutate { add_tag => "popular" }  
  }  
}
```

- Output Example:

```
# Send production errors to pagerduty  
output {  
  if [loglevel] == "ERROR" and [deployment] == "production" {  
    pagerduty {  
      service_key => "my_pagerduty_api_key"  
      details => {  
        "message" => "Error in production: ${[message]}"  
      }  
    }  
  }  
}
```

Gabriel Montoya 20-Nov-17 - Karpch TrafficCom

Conditionals

- How about a more complex example?
 - alert Nagios for apache events with response 5xx
 - alert Hipchat for apache events with response 4xx
 - count all apache response-code hits via StatsD
 - record all logs in Elasticsearch

```
output {  
  if [type] == "apache" {  
    if [response] =~ /^5\d\d/ {  
      # these need to be fixed with higher priority  
      nagios {...}  
    } else if [response] =~ /^4\d\d/ {  
      # send 400s to a specific irc channel for analysis/cleanup  
      hipchat {...}  
    }  
    # increment all apache response code counters  
    statsd { increment => "apache.response.%{[status]}" }  
  }  
  # send to elasticsearch no matter what type  
  elasticsearch {...}  
}
```



drop Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

drop

- What?
 - Used to drop events before sending it to the output
 - Usually combined with conditionals
- Why?
 - events that you are not interested
 - reduce network
 - reduce cpu with other filters

Gabriel Montoya - 20-Nov-2017 - Kappa TrafficCom

drop Config

- Remember the CSV header file?

```
filter {  
  if [country] == "Country" {  
    drop{}  
  }  
}
```

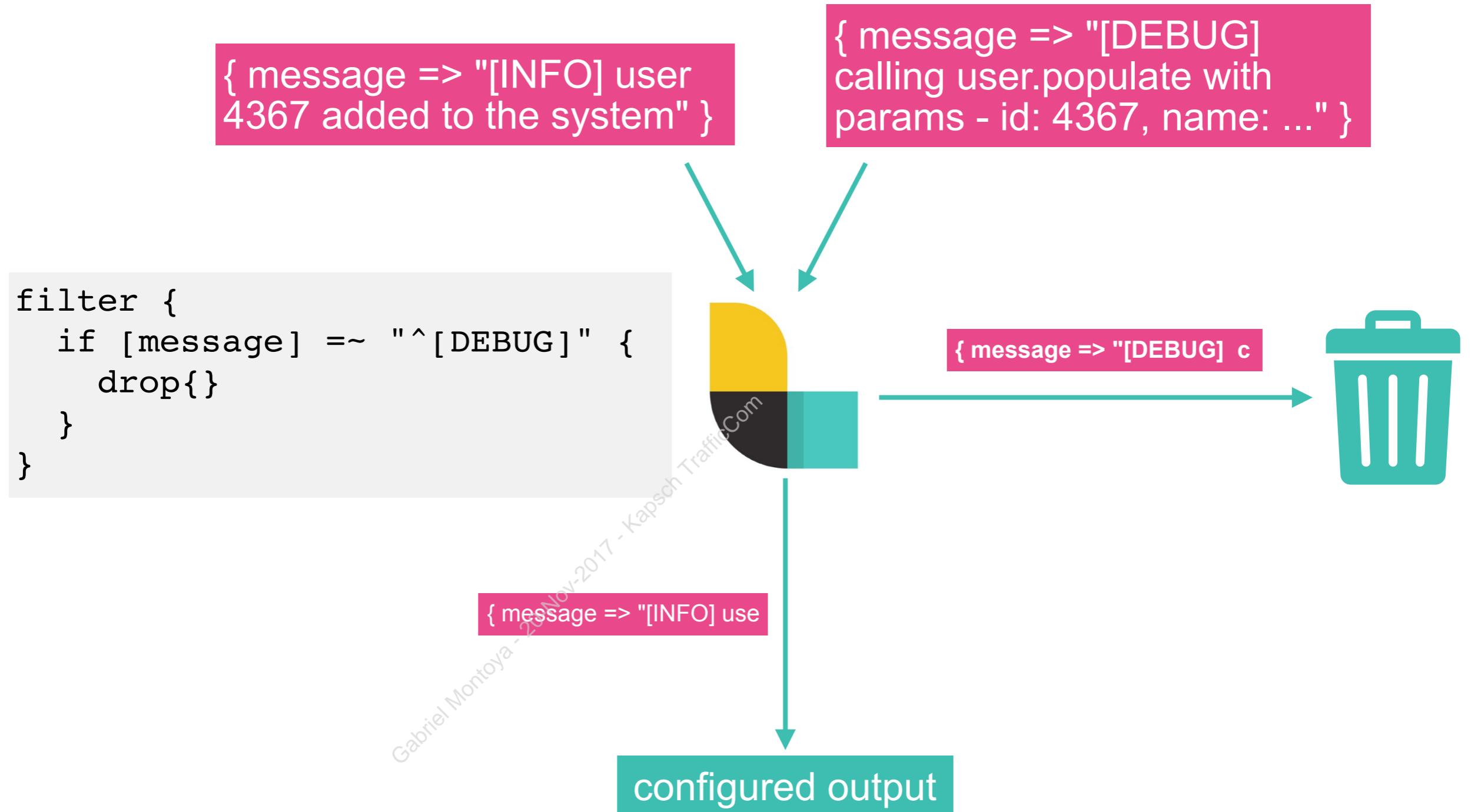
- What about logs from dev or stage environments?

```
filter {  
  if ("dev" in [tags]) OR ("stage" in [tags]) {  
    drop{}  
  }  
}
```

- Debug messages?

```
filter {  
  if [message] =~ "[ DEBUG ]" {  
    drop{}  
  }  
}
```

drop Example



mutate Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

mutate

- What?
 - allows you to perform general mutations on fields:
 - rename
 - remove
 - replace
 - modify (convert, gsub, lowercase, ...)
- Why?
 - string should be an integer
 - field name is not descriptive
 - remove useless fields

mutate Config

```
filter {  
  mutate {  
    add_field => {  
      "full_name" => "%{[first_name]} %{[last_name]}"  
    }  
  }  
}
```

```
filter {  
  mutate {  
    remove_field => [ "message" ]  
  }  
}
```

```
filter {  
  mutate {  
    convert => {  
      "bytes" => "integer"  
    }  
  }  
}
```



mutate Example

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "first_name" => "Alan",  
  "last_name" => "Turing"  
}
```



```
filter {  
  mutate {  
    add_field => {  
      "full_name" => "%{[first_name]} %{[last_name]}"  
    }  
  }  
}
```

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "first_name" => "Alan",  
  "last_name" => "Turing",  
  "full_name" => "Alan Turing"  
}
```



CSV Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

CSV

- What?
 - takes an event field containing CSV data, parses it, and stores it as individual fields
 - can optionally specify the names
 - any separator
- Why?
 - create a structured JSON document out of CSV files

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



csv config

```
filter {  
  csv {  
    columns => ["field1", "field2", "field3"]  
  }  
}
```

```
filter {  
  csv {  
    columns => ["field1", "field2", "field3"]  
    separator => ";" # default:  
    quote_char => "" # default:  
    source => "fieldA" # default: message  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - KafkaChirp.com



csv Example

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "message" => "\"Marvel\", \"Wolverine\", \"Canada\""  
}
```



```
filter {  
  csv {  
    columns => ["Publisher", "Name", "Country"]  
  }  
}
```

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "message" => "\"Marvel\", \"Wolverine\", \"Canada\"",  
  "Publisher" => "Marvel",  
  "Name" => "Wolverine",  
  "Country" => "Canada"  
}
```



dissect Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

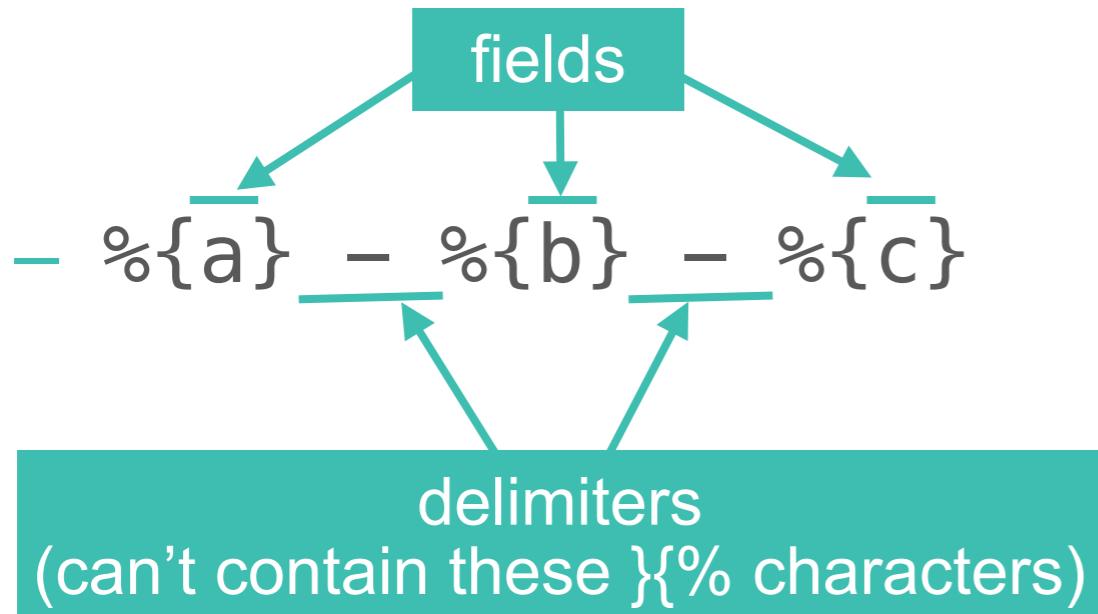
dissect

- What?
 - a kind of split operation
 - applies a set of delimiters # to a string value
 - does not use regular expressions and is very fast
- Why?
 - create a structured JSON document out of a string
 - very fast

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

dissect config

- A dissection is described using a set of `%{}` sections:



- The remaining text is stored in the last field
- The config might look like this:

```
filter {  
  dissect {  
    mapping => {  
      "message" => "%{field1} %{field2} - [%{field3} %{field4}]"  
    }  
  }  
}
```

dissect config

```
filter {  
  dissect {  
    mapping => {  
      "message" => "%{ts} %{+ts} %{+ts} %{[]}%{src}"  
    }  
  }  
}
```

use an empty brackets to not add
the found value to the event

use + to append the value to another
value or store if its the first value seen.
(The delimiter found before the field is
appended with the value.)

May 15 10:40:48 - user login failed (userid: 10)

ts => May 15 10:40:48
src => user login failed (userid: 10)

dissect config

```
filter {  
  dissect {  
    mapping => {  
      "message" => "error: %{?err}, %{&err}"  
    }  
  }  
}
```

use & to add the found value to the event using
the found value of another field as the key

use ? to not add the found value to the
event, but store it internally.

error: some_error, some_description

some_error => some_description



translate Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

translate

- What?
 - A general search and replace tool that uses a configured hash and/or a file to determine replacement values.
 - Dictionary lookup
 - JSON { "Brazil": "-14.235004,-51.92528" }
 - YML Brazil: "-14.235004,-51.92528"
 - CSV Brazil,"-14.235004,-51.92528"
 - Array ["Brazil", "-14.235004,-51.92528"]
- Why?
 - Enrich the event with external information

translate config

```
filter {  
  translate {  
    dictionary => [  
      "keyA", "valueA",  
      "keyB", "valueB",  
      "keyC", "valueC",  
    ]  
    field => "fieldM"  
  }  
}
```

```
filter {  
  translate {  
    dictionary_path => "/home/ubuntu/datasets/countries_lat_lon.csv"  
    field => "country"  
    destination => "location" ← default: "translation"  
    fallback => "0,0"  
  }  
}
```

translate Example

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "country" => "Brazil"  
}
```



```
filter {  
  translate {  
    dictionary_path => "/home/countries_lat_lon.csv"  
    field => "country"  
    destination => "location"  
  }  
}
```

```
{  
  "@timestamp" => 2017-10-20T13:43:02.914Z,  
  "country" => "Brazil",  
  "location" => "-14.235004,-51.92528"  
}
```



Logstash Config File Reload

Gabriel Montoya - 20-Nov-2017 - KapschTrafficCom

Logstash Config File Reload

- Detect and reload configuration changes automatically
- Use --config.reload.automatic (or -r) command-line option

```
./logstash/bin/logstash -f logstash.conf --config.reload.automatic
```
- Use --config.reload.interval to define how often Logstash checks for configuration changes (default: every 3 seconds)
- If Logstash is already running without auto-reload enabled, you can force Logstash to reload the config file and restart the pipeline by sending a SIGHUP (signal hangup) to the process running Logstash. For example:

```
kill -1 1475
```

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- You can store unstructured data, but not analyze it.
- **Logstash** implements a large set of filters to help you transform unstructured data into structured data.
- Logstash allows you to use the current event values to test conditions and set filters and output settings.
- The **drop** filter help to reduce data noise and network traffic.
- The **mutate** filter allows you to better model your event.
- Filters like **csv** and **dissect** help extract fields from strings.
- The **translate** filter allows you to enhance the event information by performing Dictionary lookups.



Quiz

- 1. True or False:** Most business questions can be answered by analyzing unstructured data.
- 2. What is wrong in the following code?**

```
filter {  
  if %{[last_name]} {  
    mutate {  
      add_fields => { "full_name" => "[first_name] [last_name]" }  
    }  
  }  
}
```

- 3. True or False:** The **dissect** filter is faster than the **csv** filter?
- 4. Which filter would you use to tag events based on a dictionary?**
- 5. Which filter would you use to exclude events you are not interested in?**

Lab 5

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 6

Data Enrichment

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

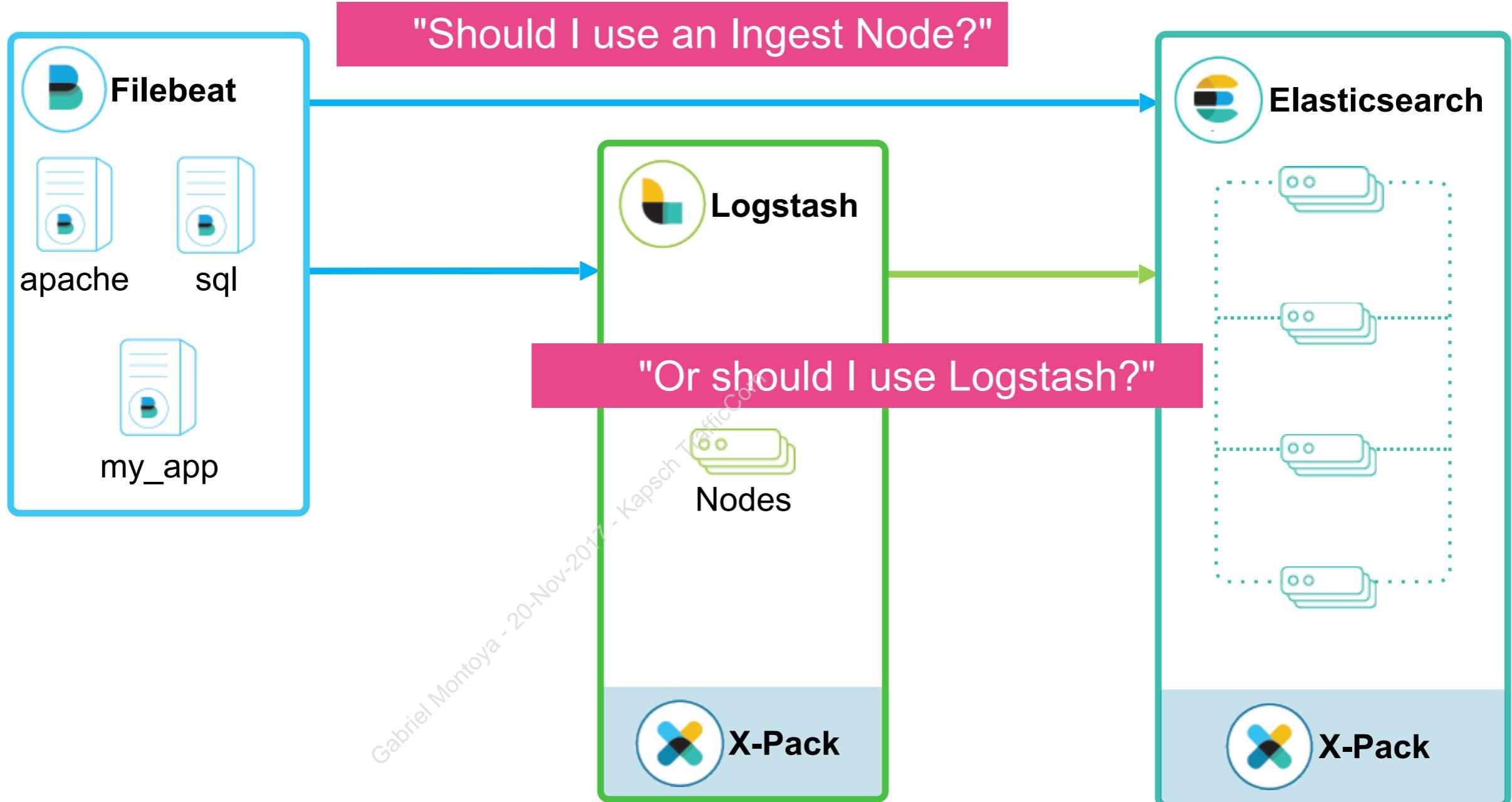
- Ingest Node or Logstash?
- Grok Filter
- Date Filter
- GeolP Filter
- User Agent Filter
- Ruby Filter
- Elasticsearch Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Ingest Node or Logstash?

Gabriel Montoya - 20-Nov-2017 - KapschTecCom

Choosing an Architecture



Ingest Nodes or Logstash?

- Ingest nodes and **Logstash** offer similar functionality:

Ingest Node Pro's:

- Lightweight
- Good for parsing simple data
- Great for reindexing
- Clustered

Logstash Pro's

- Many plugins to extend functionality
- Can parse just about anything
- Isolate your architecture
- Can queue messages

Ingest Node Con's

- Limited functionality
- Adds complexity to the cluster
- No message queuing

Logstash Con's

- More complex to configure
- Another thing to deploy
- Can be heavier

- Gather your requirements, examine your scale, and plan for the future!

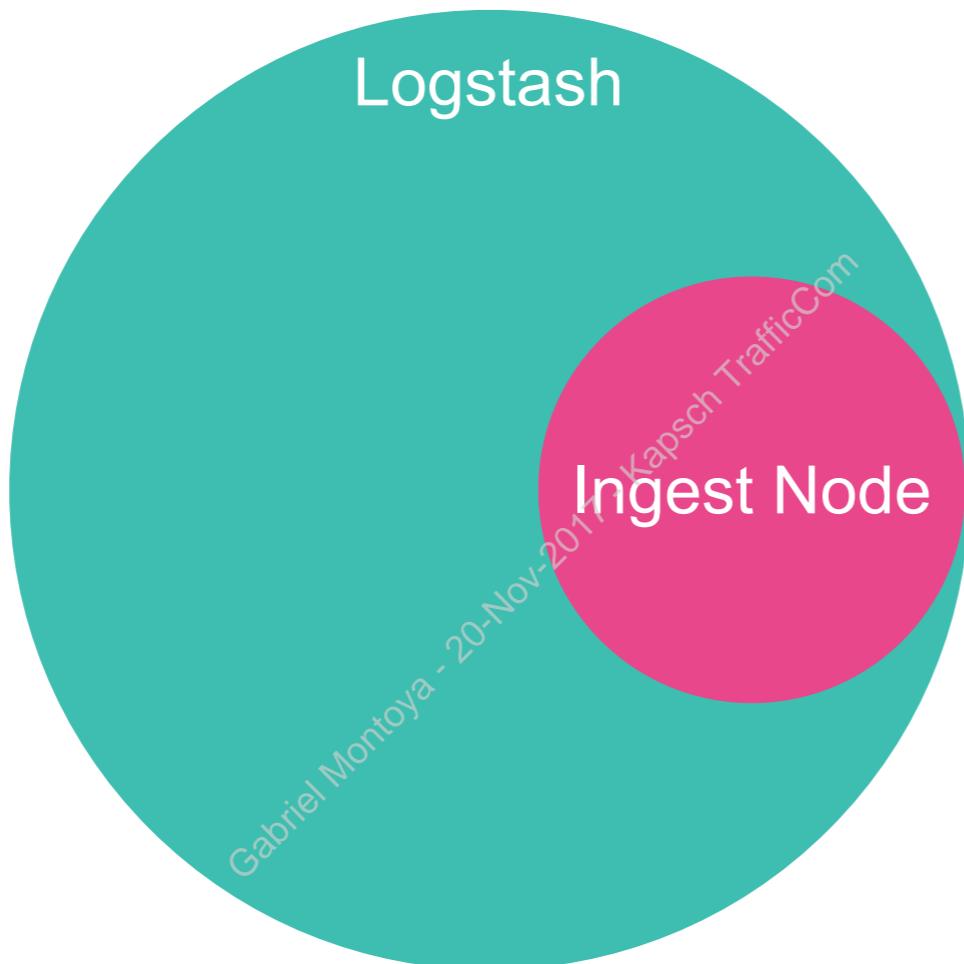


Functionality Comparison

- Both have many features, but ingest nodes are a subset of Logstash

Many shared features:

- grok
- rename
- date



But some missing:

- different inputs/outputs
- user agents
- lookups

Choosing an Architecture

- For our use case we want to convert this:

```
212.123.230.34 - - [18/Mar/  
2014:18:31:15 -0500] "GET /  
kibana-3.0.0 milestone5/app/panels/  
query/meta.html HTTP/1.1" 200 1349  
"http://localhost/  
kibana-3.0.0 milestone5/"  
"Mozilla/5.0 (Macintosh; Intel Mac OS X  
10_9_2) AppleWebKit/537.74.9 (KHTML,  
like Gecko) Version/7.0.2 Safari/  
537.74.9"
```

This is an **Apache** log line - we want to expand the ip address and user agent data highlighted in pink in addition to structuring this into a document

Choosing an Architecture

- Into this:

We need IP addresses expanded into lat/lon and geoip data

```
{  
  "_index": "logstash-2013.08.27",  
  "_type": "log",  
  "_id": "AVlouXF7hVomG41PDmki",  
  "_score": 1,  
  "_source": {  
    "request": "/admin",  
    "agent": "\"Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3\"",  
    "geoip": {  
      "timezone": "Europe/Amsterdam",  
      "ip": "212.123.230.34",  
      "latitude": 52.35,  
      "continent_code": "EU",  
      "city_name": "Amsterdam",  
      "country_code2": "NL",  
      "country_name": "Netherlands",  
      "country_code3": "NL",  
      "region_name": "North Holland",  
      "location": [  
        4.9167,  
        52.35  
      ],  
      "postal_code": "1091",  
      "longitude": 4.9167,  
      "region_code": "NH"  
    },  
    "offset": 176,  
    "auth": "kurt",  
    "ident": "-",  
    "input_type": "log",  
    "verb": "GET",  
    "useragent": {  
      "patch": "3",  
      "os": "Windows XP",  
      "major": "3",  
      "minor": "6",  
      "name": "Firefox",  
      "os_name": "Windows XP",  
      "device": "Other"  
    },  
    "source": "/path/to/lab/logs/sample7.log",  
    "message": "212.123.230.34 - kurt [18/May/2011:01:48:10 -0700] \"GET /admin HTTP/1.1\" 301 566 \"-\" \"Mozilla/5.0 (Windows; U;  
Windows NT 5.1; en-US; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3\"",  
    "type": "log",  
    "tags": [  
      "beats_input_codec_plain_applied"  
    ]  
  }  
}
```

We want to easily structure the user agents

Choosing an Architecture

- To do this we must use **Logstash**
 - provides *user agent* filters
- Filters required to make this transformation:
 - **grok** - for parsing the log line after input into Logstash
 - **date** - to transform the timestamp
 - **geoip** - to enrich the IP address into geo-coordinates and locality information
 - **useragent** - to transform and structure the user agent text
- Luckily, some of these can be reused if we decided to use an ingest node!

Gabriel Montoya
10-Nov-2017
Kapsch TrafficCom



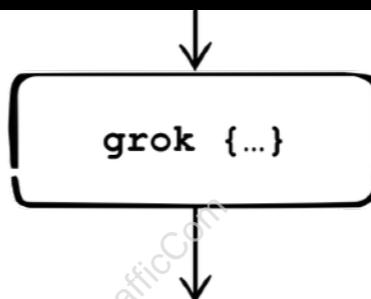
Grok Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Grok

Creating structured data from unstructured data

```
{  
  "@version"    => "1",  
  "@timestamp"  => "2015-06-02T08:42:13.562Z",  
  "host"        => "test.example.org",  
  "message"     => "83.149.9.216 - - [01/Jun/2015:21:13:45 -0200] \"GET /presentations/logstash-monitorama-2013/plugin/notes/notes.js HTTP/1.1\" 200 2892 \"http://semicomplete.com/presentations/logstash-monitorama-2013/\" \"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/537.36\""  
}
```



```
{  
  "clientip"      => "83.149.9.216",  
  "ident"         => "-",  
  "auth"          => "-",  
  "timestamp"     => "01/Jun/2015:21:13:45 -0200",  
  "verb"          => "GET",  
  "request"       => "/presentations/logstash-monitorama-2013/plugin/notes/notes.js",  
  "httpversion"   => "1.1",  
  "response"      => 200,  
  "bytes"         => 2892,  
  "referrer"      => "\"http://semicomplete.com/presentations/logstash-monitorama-2013/\"",  
  "agent"         => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/537.36\""  
}
```

<https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html>



Why grok?

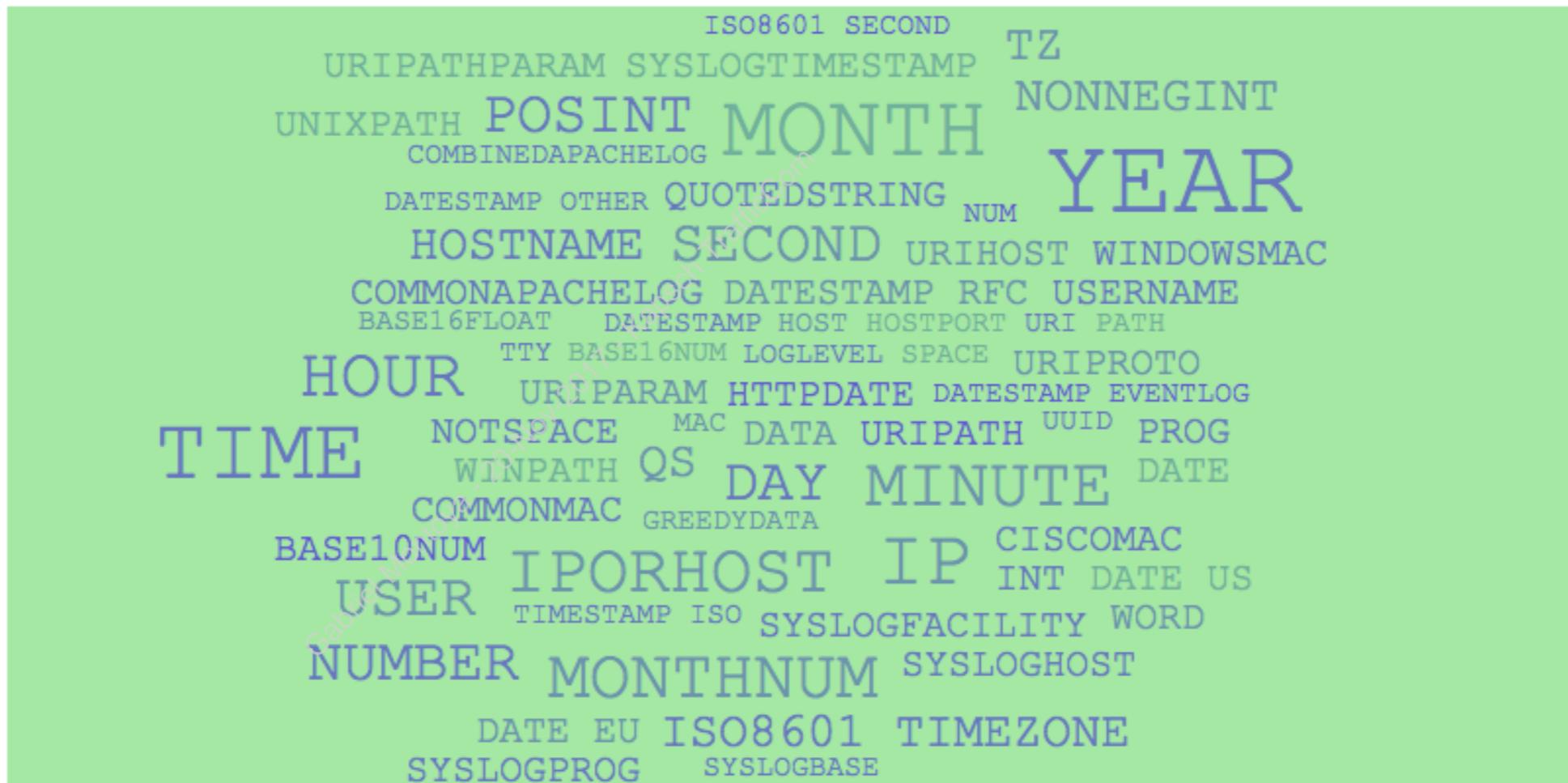
What does this regular expression match?

```
(( (?:(\b(?:[0-9A-Za-z][0-9A-Za-z-]{0,62}))(?:\.(?:[0-9A-Za-z][0-9A-Za-z-]{0,62}))*(\.\.?|\b))|((?<![0-9])(?:(:25[0-5]|2[0-4][0-9]| [0-1]?[0-9]{1,2})[.](?:25[0-5]|2[0-4][0-9]| [0-1]?[0-9]{1,2})[.](?:25[0-5]|2[0-4][0-9]| [0-1]?[0-9]{1,2})[.](?:25[0-5]|2[0-4][0-9]| [0-1]?[0-9]{1,2})[.](?:!([0-9])))|(([a-zA-Z0-9_-]+)|([a-zA-Z0-9_-]+)) \[(((?:(:0[1-9])|(:[12][0-9])|(:3[01])|[1-9])) /(\b(?:Jan(?:uary)?|Feb(?:ruary)?|Mar(?:ch)?|Apr(?:il)?|May|Jun(?:e)?|Jul(?:y)?|Aug(?:ust)?|Sep(?:tember)?|Oct(?:ober)?|Nov(?:ember)?|Dec(?:ember)?)\b)/((?>\d\d){1,2}):((?!<[0-9])(?:2[0123]|[01][0-9]):((?:[0-5][0-9]))(?::((?:[0-5][0-9]|60)(?:[.,][0-9]+))))(?![0-9])) ((?:[+-]?(?:[0-9]+))))\]" (?:(\b\w+\b) (\$+)(?: HTTP/((?:((?<![0-9.+-])(?>[+-]?(?:(:[0-9]+(:\.[0-9]+))|(:\.[0-9]+))))))?)|-") ((?:((?<![0-9.+-])(?>[+-]?(?:(:[0-9]+(:\.[0-9]+))|(:\.[0-9]+)))))) (?:(?:((?<![0-9.+-])(?>[+-]?(?:(:[0-9]+(:\.[0-9]+))|(:\.[0-9]+))))))|-") (((?>(?<!\\"))(?>"(?>\\.|[^\\\"])+"+|"|"|(?>'(?>\\.|[^\\"']+')+')|' '|(?>`(?>\\.|[^\\\"])+`)|`|(?>`(?>\\.|[^\\\"])+`)|`))) (((?>(?<!\\"))(?>"(?>\\.|[^\\\"])+"+|"|"|(?>'(?>\\.|[^\\\"])+')+'))))
```



grok

- Uses regular expressions, but reusable
- Built-in library of known patterns (100+)
 - <https://github.com/logstash-plugins/logstash-patterns-core/tree/master/patterns>
- Extensible: Add your own custom patterns



High-level grok

- Format:

```
%{SYNTAX:fieldname}
```
- Or alternatively:

```
%{SYNTAX:fieldname:type}
```

The syntax represents a *regular expression*

SYNTAX must be capitalized

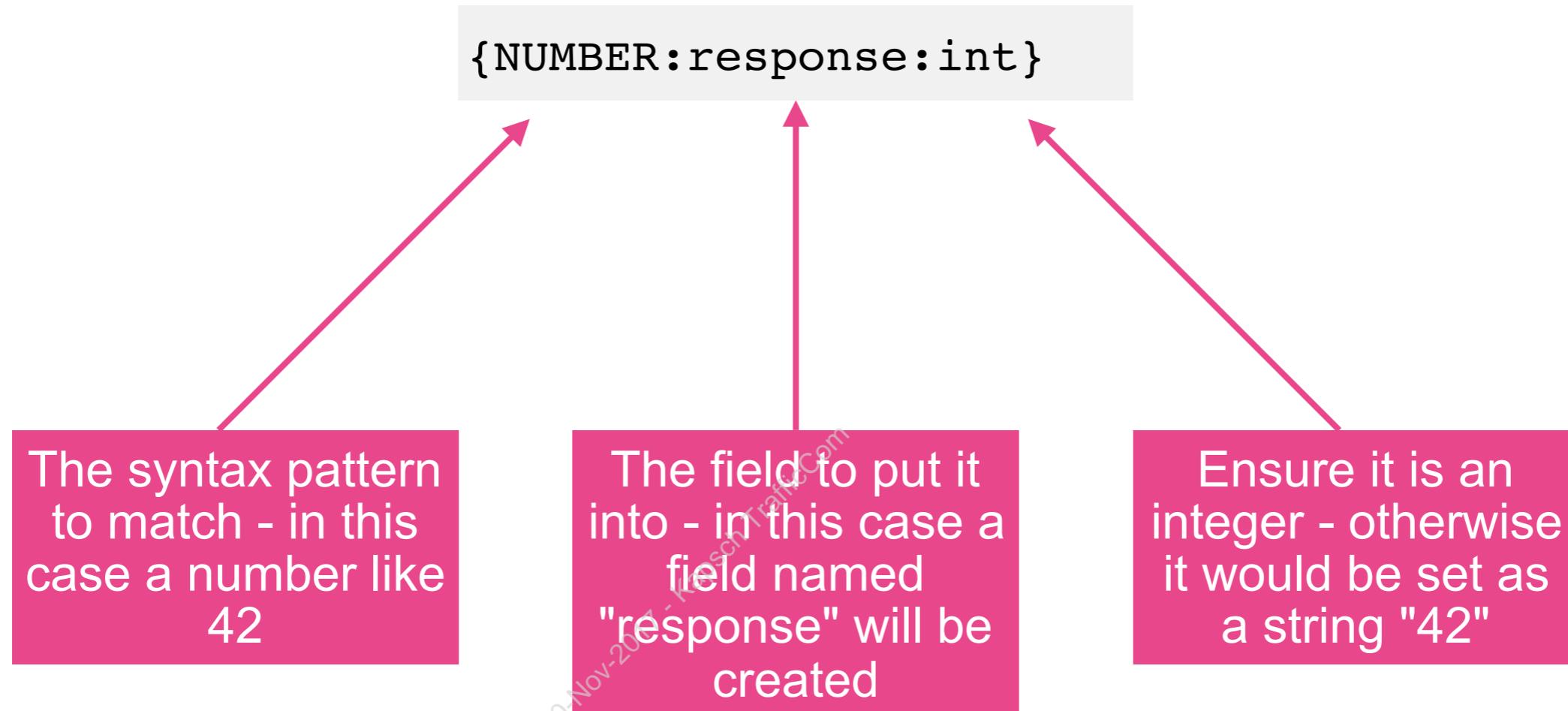
type defaults to **string**

You **must** convert to **int** or **float**

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Concrete grok

- This would match a *numeric* field:



There are many grok patterns - this is only 1 of them!

Parsing a Log Line with grok

- How can we match this Apache log line?

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrainCom

grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[%{HTTPDATE:timestamp}\] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```

grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```



grok

```
212.123.230.34 - - [18/Mar/2014:18:31:15 -0500]
"GET /kibana-3.0.0milestone5/app/panels/query/
meta.html HTTP/1.1" 200 1349 "http://localhost/
kibana-3.0.0milestone5/"
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2)
AppleWebKit/537.74.9 (KHTML, like Gecko)
Version/7.0.2 Safari/537.74.9"
```

```
%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp} \] "%{WORD:verb} %
{DATA:request} HTTP/%{NUMBER:httpversion}" %
{NUMBER:response:int} (?:- | %{NUMBER:bytes:int})
%{QS:referrer} %{QS:agent}
```

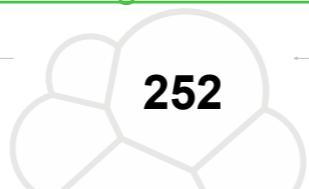


grok in the Configuration File

```
filter {
  grok {
    match => {
      "message" => '%{IPORHOST:clientip} %{USER:ident} %{USER:auth}
\[ %{HTTPDATE:timestamp}\] "%{WORD:verb} %{DATA:request} HTTP/%
{NUMBER:httpversion}" %{NUMBER:response:int} (?:-|%
{NUMBER:bytes:int}) %{QS:referrer} %{QS:agent}'
    }
  }
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

<https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html>



Creating Grok Patterns

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Creating Grok Patterns

- Grok is easier than plain regular expression
- Still, it can be very hard to write the right Grok Expression
- Grok Debugger can help!

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



How to Create and Debug a grok Pattern?

http://grokdebug.herokuapp.com

Grok Debugger Debugger Discover Patterns

Input

Pattern

Add custom patterns Keep Empty Captures Named Captures Only Singles Autocomplete

::1 - - [18/Mar/2014:18:31:15 -0500] "GET /kibana-3.0.0milestone5/app/panels/query/meta.html HTTP/1.1" 200 1349 "http://localhost/kibana-3.0.0rr

Pattern

Add custom patterns Keep Empty Captures Named Captures Only Singles Autocomplete

{
}



How to Create and Debug a grok Pattern?

::1 -- [18/Mar/2014:18:31:15 -0500] "GET /kibana-3.0.0milestone5/app/panels/query/meta.html HTTP/1.1" 200 1349 "http://localhost/kibana-3.0.0m
%{IPORHOST:clientip}

Add custom patterns Keep Empty Captures Named Captures Only Singles Autocomplete

```
{  
  "clientip": [  
    [  
      "::1"  
    ]  
  ],  
  "HOSTNAME": [  
    [  
      null  
    ]  
  ],  
  "IP": [  
    [  
      "::1"  
    ]  
  ],  
  "IPV6": [  
    [  
      "::1"  
    ]  
  ],  
  "IPV4": [  
    [  
      null  
    ]  
  ]  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



How to Create and Debug a grok Pattern?

- *Named Captures Only + Singles* can make the output easier to read

The screenshot shows the Elasticsearch Grok pattern configuration interface. At the top, there is a preview window showing a log entry and its parsed version. Below the preview are several configuration options: "Add custom patterns", "Keep Empty Captures", "Named Captures Only" (which is checked), "Singles" (which is checked), and "Autocomplete". The main area contains the Grok pattern definition:

```
{  
  "clientip": [  
    "::1"  
  ]  
}
```

A watermark "Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom" is diagonally across the interface.



How to Create and Debug a grok Pattern?

- Add **one** pattern at a time and check the output

```
::1 -- [18/Mar/2014:18:31:15 -0500] "GET /kibana-3.0.0milestone5/app/panels/query/meta.html HTTP/1.1" 200 1349 "http://localhost/kibana-3.0.0m  
%{IPORHOST:clientip} %{USER:ident} %{USER:auth} \[%{HTTPDATE:timestamp}\]\n
```

Add custom patterns Keep Empty Captures Named Captures Only Singles Autocomplete

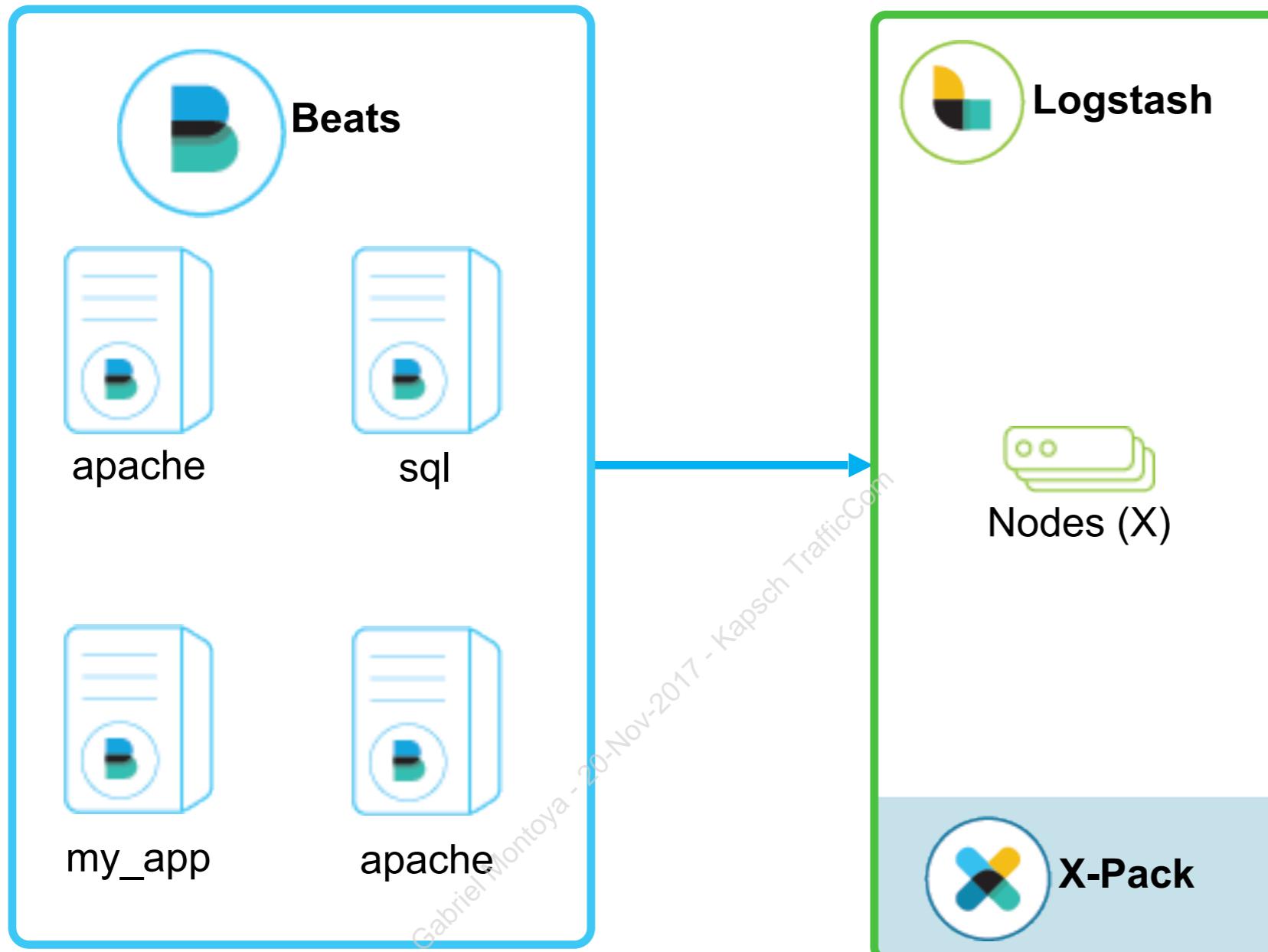
```
{  
  "clientip": [  
    "::1"  
  ],  
  "ident": [  
    "_"  
  ],  
  "auth": [  
    "_"  
  ],  
  "timestamp": [  
    "18/Mar/2014:18:31:15 -0500"  
  ]  
}
```



Using Grok On Multiple Log Types

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

What if logs are different?



3 types of logs being sent to Logstash

This is not something a single grok pattern can parse!

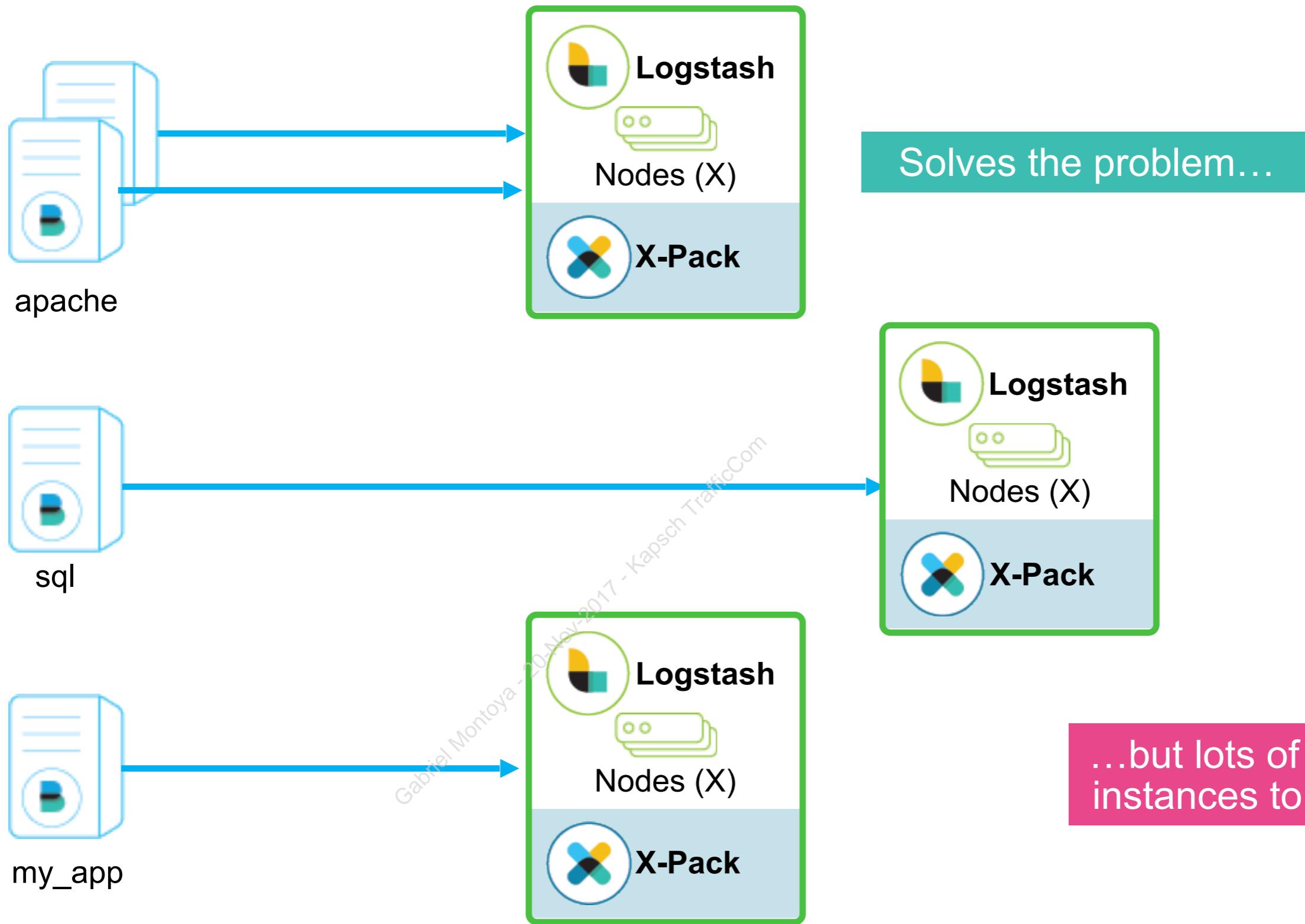
You end up with
"`_grokparsefailure`"
errors in Logstash

What if logs are different?

- 2 approaches can work
 1. Use different **Logstash** instances for each log type
 2. Use **conditionals** in the Logstash config to check the type of each log set in **Filebeat**, and use the correct grok pattern from there

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

1. Using Separate Logstash Instances



2. Using Conditionals in grok

1. Set the *type* different for each log type in **Beats**

```
input {  
  file {  
    path => "/var/log/apache/access_log"  
    type => "apache"  
  }  
}
```

2. Match that type using conditionals in **Logstash**

```
if [type] == "apache" {  
  grok {  
    match => ["message", "your apache pattern"]  
  }  
}  
else if [type] == "my_app" {  
  grok {  
    match => ["message", "your my-app pattern"]  
  }  
}
```

A more elegant solution that uses the "*type*" each log line is tagged with - could use tags too



Handling `_grokparsefailure`

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

_grokparsefailure

- When grok fails to parse a message correctly you'll see this:

_grokparsefailure

- This happened because your message wasn't able to be parsed - but why?
 - a rogue message came into **Logstash**?
 - the log line structure changed?
 - a 1-off occurrence that wasn't planned for?
 - wrong *tag* or *type* set in **Filebeat**?
- You will want to handle this...



Troubleshooting _grokparsefailure

- **Prevention:** Use conditionals in your grok filters to ensure that only messages tagged with a certain type or tag are parsed
- **Investigating:** You may still want to store these messages to investigate why it occurred

```
if [type] == "apache" {  
    grok {  
        match => ["message", "your apache pattern"]  
        tag_on_failure => ["apache_parse_fail"]  
    }  
}  
else if [type] == "my-app" {  
    grok {  
        match => ["message", "your my-app pattern"]  
        tag_on_failure => ["my_app_parse_fail"]  
    }  
}
```

Add a tag to this messages that says I am an "apache_parse_fail"

Tag me as "my_app_parse_fail"

Excluding bad messages

- **Dropping:** you could drop the messages as well if you don't want to investigate:

```
if [type] == "apache" {  
    grok {  
        match => ["message", "your apache pattern"]  
        tag_on_failure => ["apache_parse_fail"]  
    }  
}  
else if [type] == "my-app" {  
    grok {  
        match => ["message", "your my-app pattern"]  
        tag_on_failure => ["my-app_parse_fail"]  
    }  
}  
if "_grokparsefailure" in [tags] {  
    drop { }  
}
```

*"Drop me because I have
_grokparsefailure in my tags"*

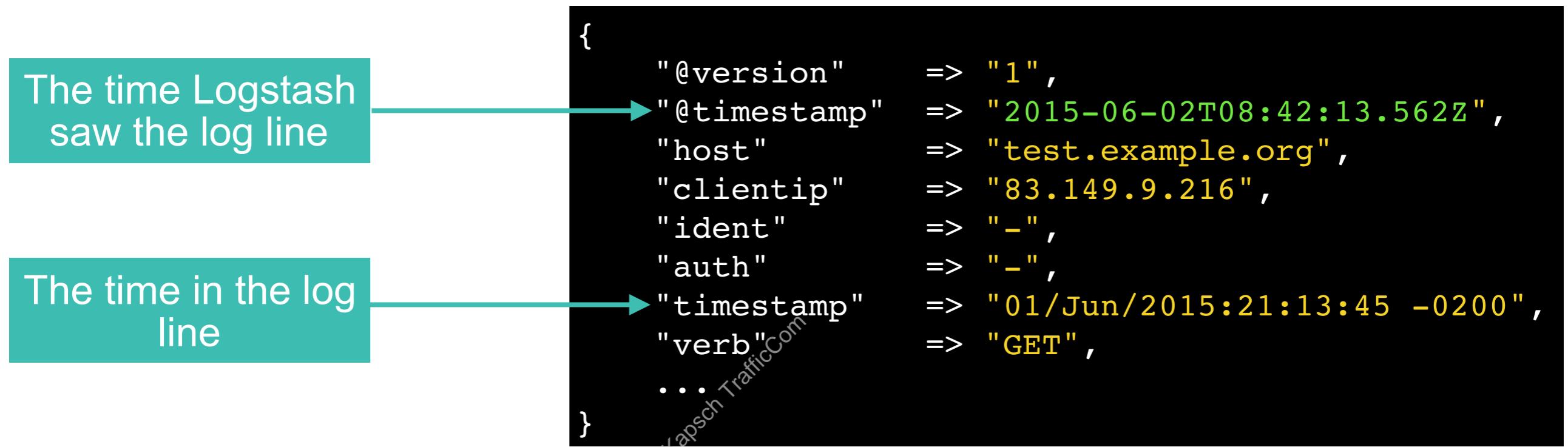


Date Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

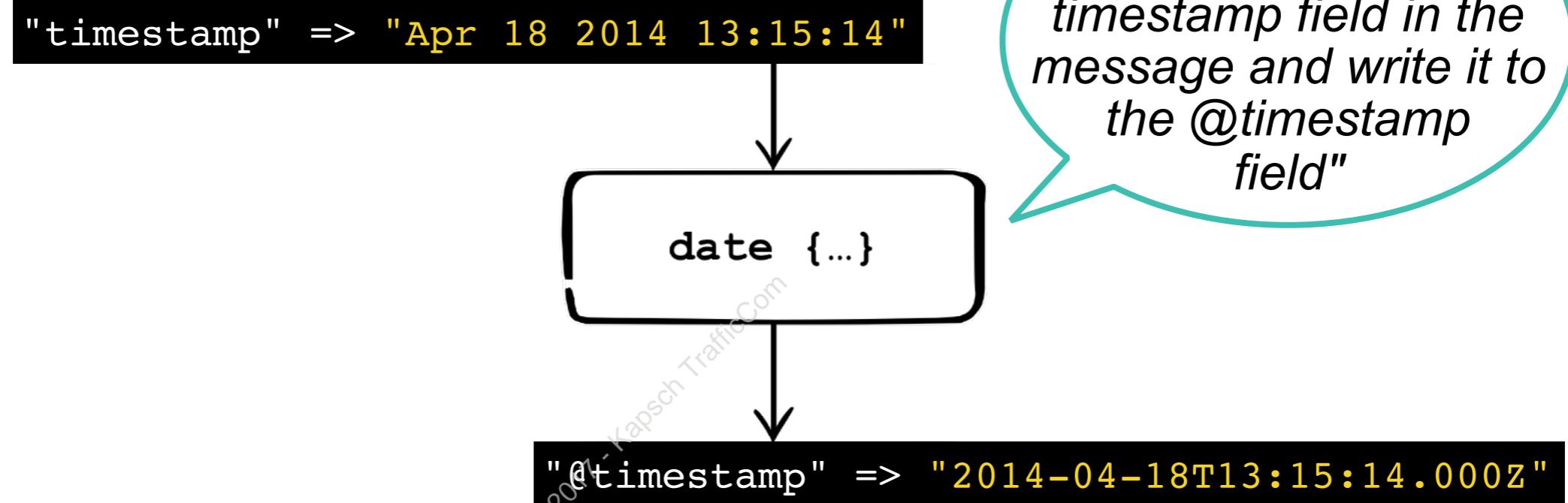
Date filter output

- We currently have 2 time stamps



Date filter

What time did this event occur?



Date Filter Configuration

- Format: 18/Mar/2014:18:31:15 -0500

dd/MMM/YYYY:HH:mm:ss Z

- Configure using the match directive:

```
filter {  
    date {  
        match => [ "timestamp", "dd/MMM/YYYY:HH:mm:ss Z" ]  
        # target => "@timestamp" # this is the default  
    }  
}
```

- By default, the date filter overwrites the **@timestamp** field, but this can be changed by providing an explicit target field

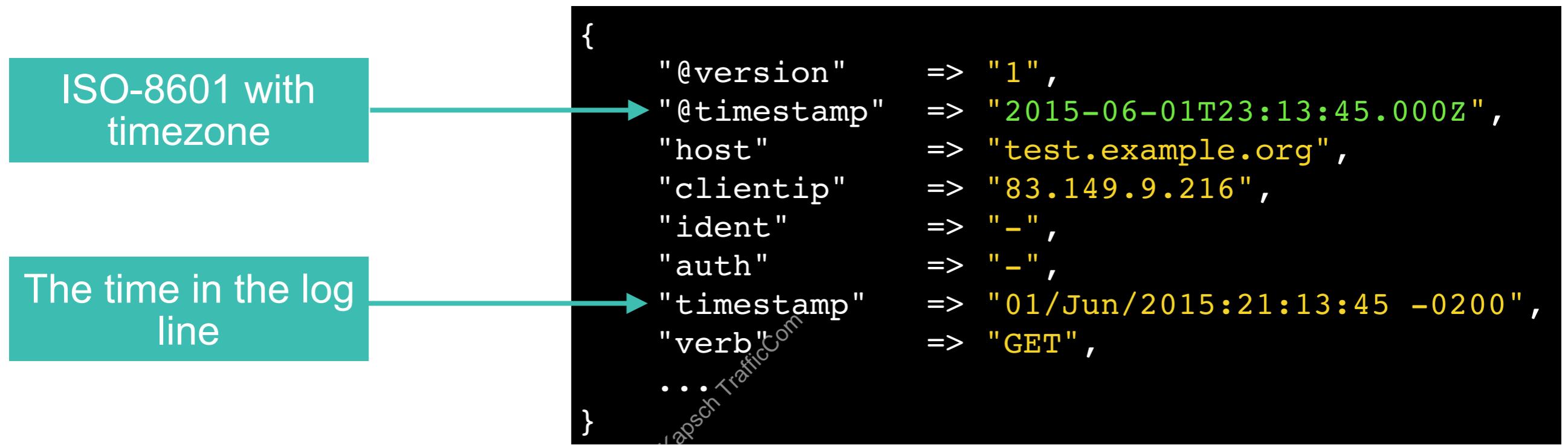
Date Filter Table of Symbols

Symbol	Meaning	Example
Y	year	2014
M	month of year	July; Jul; 07
d	day of month	10
a	half-day of day	AM; PM
H	hour of day (0-23)	0
m	minute of hour	30
s	second of minute	15
S	fraction of second	978
Z	time zone offset/id	-0800; -08:00; America/Los_Angeles



Date Filter @timestamp Set

- Reconciled timestamps - the "processed" timestamp has been replaced



Remove or keep the processed date?

- Sometimes, you may want to keep the original **@timestamp**, to know when Logstash processed an event
 - keep a record of when the event was generated and when it was processed
- You can copy it to another field, and next replace **@timestamp** with the timestamp in your log

```
filter {  
    mutate {  
        rename { "@timestamp" => "processed_at" }  
    }  
    date {  
        match => [ "timestamp", "dd/MMM/YYYY:HH:mm:ss z" ]  
    }  
}
```

*"I'm going to
create a new field
called **processed_at**
and then write the
event timestamp to
@timestamp"*

Removing the Superfluous Timestamps

- If you want to remove the old timestamp then use the `remove_field` setting

```
filter {  
    date {  
        match => [ "timestamp", "dd/MMM/YYYY:HH:mm:ss z" ]  
        remove_field => "timestamp"  
    }  
}
```

since the timestamp of
the log line has been
copied to `@timestamp`, it
can be removed from its
original field

Timezones & Localities

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Timezones

- A word about **timezones**...
 - if unspecified, will default to server local time (what the output from the date command shows)
 - to manually specify, use the timezone configuration option

```
filter {  
    date {  
        timezone => "America/Los_Angeles"  
        # ... other configuration ...  
    }  
}
```

Localities

- A word about *locales*...
 - locale sets the spoken language of the logs
 - the locale is mostly necessary to be set for parsing month names and weekday names. (e.g. French or German day/month names)

```
filter {  
    date {  
        locale => "en" # english  
        # ... other configuration ...  
    }  
}
```

GeoIP Filter

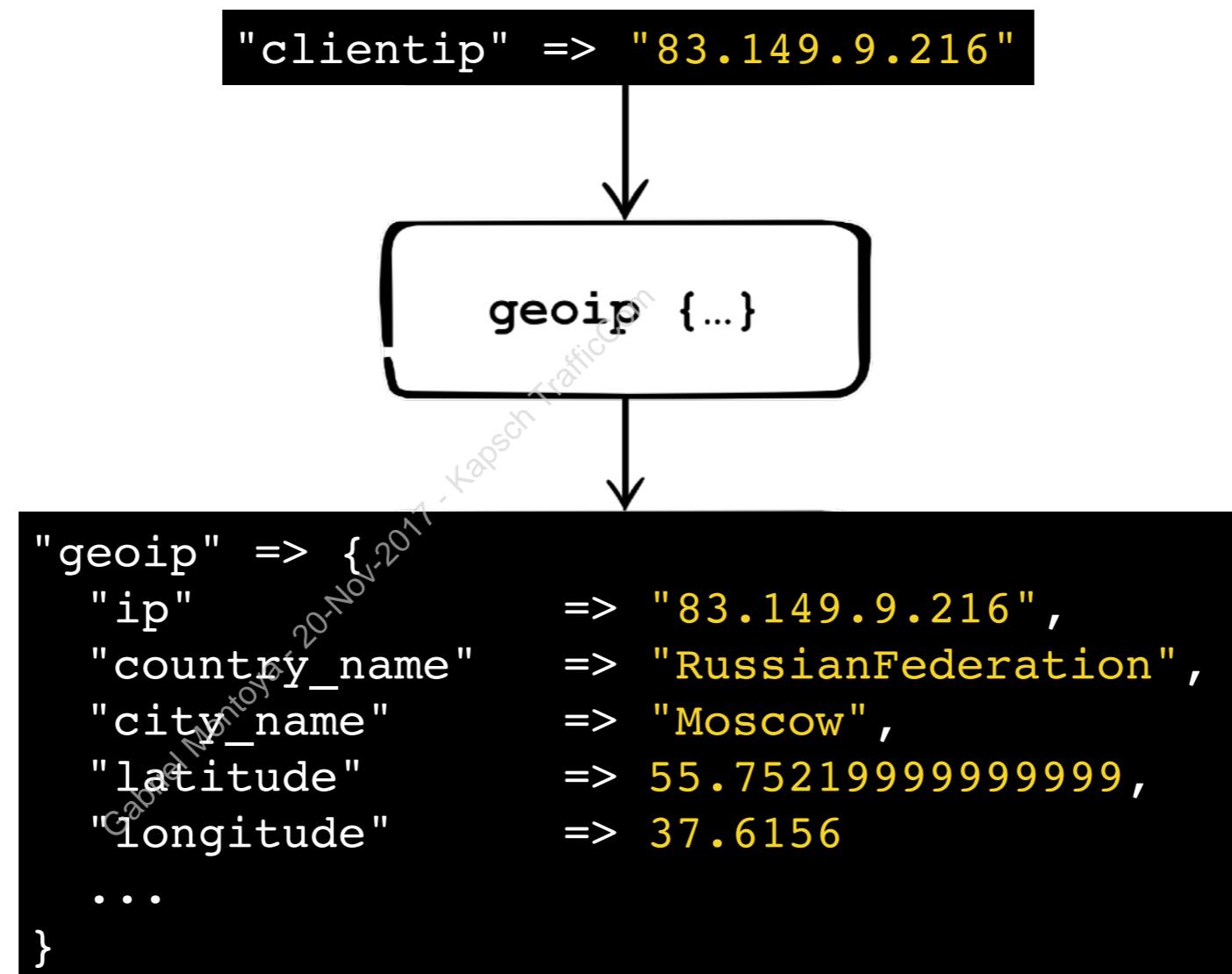
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

The geoip Filter

- Based on an IP, looks up a geographical location

"Where are customers buying things from?"

"Where are security attacks coming from?"



How geoip Works

- Performs lookup against GeoLiteCity database for valid IP addresses and populates fields with results
- Logstash ships with the GeoLiteCity database made available from Maxmind with a CCA-ShareAlike 3.0 license
- GeoCityLite is a local file read by **Logstash**, and so no network call is made. Fast!
- For more details on GeoLite, see <http://www.maxmind.com/en/geolite>
- Can also generate your own IP database, see <https://github.com/mteodoro/mmutils>



Geoip Output

- Result:

```
"geoip" => {  
    "ip"                  => "83.149.9.216",  
    "country_name"        => "RussianFederation",  
    "continent_code"     => "EU",  
    "city_name"           => "Moscow",  
    "timezone"            => "Europe/Moscow",  
    "latitude"            => 55.75219999999999,  
    "longitude"           => 37.6156,  
    "continent_code"     => "EU",  
    "region_name"         => "48",  
    "country_code2"       => "RU",  
    "country_code3"       => "RUS",  
    "real_region_name"   => "MoscowCity",  
    "location": [  
        37.6156,  
        55.75219999999999  
    ]  
}
```

Now we can build tile maps using geopoints!

Gabriel Montoya - 20-Nov-2017 - elasticsearch-tricks.com

User Agent Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

User Agents

- Often, log lines from a Web server will contain information about the user agent the client is accessing

```
"agent" => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/537.36\""
```

- But it's a mess. We could grok this line but there is a special filter designed to do this for you

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

The useragent Filter

```
"useragent" => {  
    "name"      => "Chrome",  
    "os"        => "Mac OS X 10.9.1",  
    "agent"     => "\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_1) AppleWebKit/  
537.36 (KHTML, like Gecko) Chrome/32.0.1700.77 Safari/537.36\""  
    "os_major"  => "",  
    "device"    => "Other",  
    "major"     => "32",  
    "minor"     => "0",  
    "patch"     => "1700"  
}
```

Add structure to the user agent so you can query and aggregate on specific properties

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Ruby Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Why a Ruby Filter?

- Sometimes there is functionality you need that isn't built into Logstash
 - Could write your own plug-in - but that's a large task
 - Use the Ruby filter to manipulate your data during processing

```
filter {  
  ruby {  
    init => "last = ::Time.parse('2014-09-25T04:00:00+00:00');  
            @shift = ::Time.now - last"  
    code => "event.set('@timestamp',  
Logstash::Timestamp.new(event.get('@timestamp') + @shift))"  
  }  
}
```

This example takes a timestamp and then shifts it ahead from "today", which is whenever this filter is run

This is not a real world example but what we do in this class to make the old log lines "current" to when this class is running



Elasticsearch Filter

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Elasticsearch Filter

- Execute a `_search` in Elasticsearch
 - query (based on the `query_string`)
 - `query_template`
- Use results to enhance current event
- Be careful, no cache on Logstash side

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Elasticsearch Filter

```
filter {  
  elasticsearch {  
    hosts => [ "localhost:9200" ]  
    query => "type:start AND operation: %{[opid]}"  
    # query_template => "template.json"  
    fields => { "@timestamp" => "started" }  
  }  
}
```

query_string

file containing the
query using the
query DSL

```
# template.json  
{  
  "query": {  
    "query_string": {  
      "query": "type:start AND operation: %{[opid]}"  
    }  
  },  
  "_source": ["@timestamp", "started"]  
}
```

Gabriel Montoya 20-Nov-2017 Opsch TrafficCom



Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- **Grok** is an easier way to create regular expressions, as it allows you to reuse pre-defined patterns and a lot of pre-defined patterns already exist.
- **Grok debugger** is the best way to write and test your grok patterns.
- **Date filter** normalizes the event timestamp to ISO8601
- **GeolP filter** uses a local file to lookup IPs.
- **Ruby filter** allows users to write any custom code.
- **Elasticsearch filter** performs a lookup by using the `_search` API and adds the result to the event.



Quiz

1. What are the advantages of using Grok?
2. **True or False:** If a Grok pattern fails to match, Logstash drops the event.
3. **True or False:** You can have multiple grok patterns in the same grok plugin.
4. **True or False:** GeolIP lookups are really fast.
5. What are the pros and cons of the Ruby filter?
6. **True or False:** When using the Elasticsearch lookup filter, the Elasticsearch node should be running on another machine.

Gabriel Montoya - 2017-03-28 Kapsch TrafficCom



Lab 6

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 7

Data Store Integration

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Overview
- JDBC Input Plugin
- JDBC Streaming Filter Plugin
- Elasticsearch Hadoop
- Data Integrity

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Overview

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Datastore Integration Overview

- Relational Datastores
 - Mysql, MSSQL, Oracle
- NoSQL
 - Document oriented
 - MongoDB, CouchDB
 - Columnar
 - Cassandra, HBase
- **Hadoop Ecosystem**



Gabriel Montoya - 20 Nov 2017 - Kapsch TrafficCom

Use Cases

- Ingest data for search / analytics
 - Employee Directory
 - Service Management Database (eg., Help Desk tickets)
 - E-Commerce Product Catalog
 - Machine Learning Result-Set from Big Data
- Lookup and Enrich
 - Enrich Log Files with Asset Database

Gabriel Montoya - 20-Nov-2017 - KarmicTrafficCom

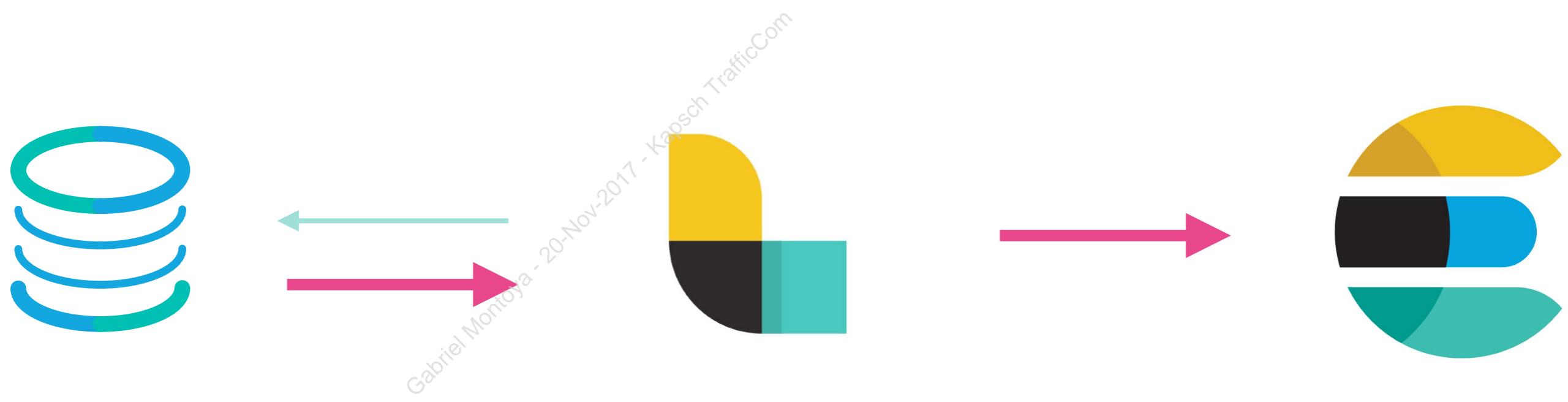


JDBC Input Plugin

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

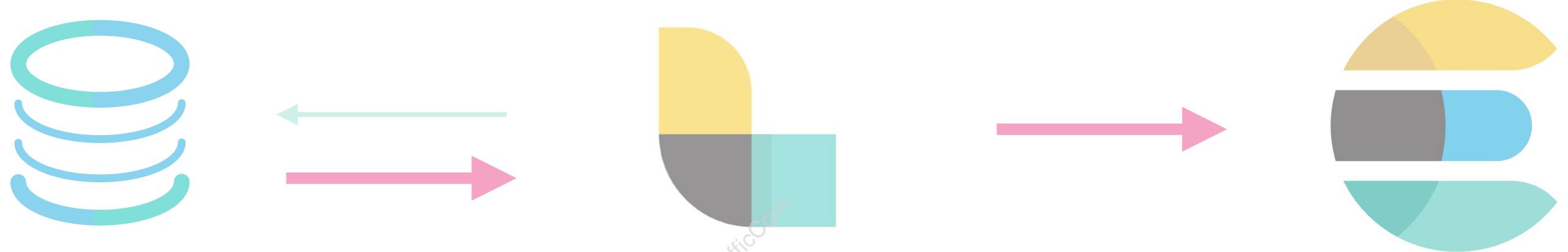
JDBC Input Plugin - Overview

- Logstash plugin
- Can ingest from any DB with JDBC interface
- Can be run on a CRON schedule
- Rows = Events
- Columns = Fields



JDBC Input Plugin - Install

- Install (versions older than 6.0)



```
bin/logstash-plugin install logstash-input-jdbc
```

JDBC Input Plugin - Configuration

```
input {  
  jdbc {  
    jdbc_driver_library => "/path/to/mysql-connector-java-bin.jar"  
    jdbc_driver_class => "com.mysql.jdbc.Driver"  
    jdbc_connection_string => "jdbc:mysql://localhost:3306/people_db"  
    jdbc_user => "logstash_service"  
    jdbc_password => "passwd"  
    schedule => "0 3 * * *"  
    statement => "SELECT * from EMP where WHERE ts > :sql_last_value"  
  }  
}
```

first_name	last_name	department	...
Erika	Rawdales	Marketing	...
Ezequiel	Wilflinger	Marketing	...
Robb	Dowd	Research	...
Harlan	Stroyan	Services	...



```
{  
  {  
    {  
      {  
        "first_name" : "Erika",  
        "last_name" : "Rawdales",  
        "department" : "Marketing",  
        ...  
        ...  
      }  
    }  
  }  
}
```

JDBC Input Plugin - Configuration

- Drivers not included with plugin
 - must be explicitly passed

```
input {  
  jdbc {  
    jdbc_driver_library => "/path/to/mysql-connector-java-bin.jar"  
    jdbc_driver_class => "com.mysql.jdbc.Driver"  
    jdbc_connection_string => "jdbc:mysql://localhost:3306/people_db"  
    jdbc_user => "logstash_service"  
    schedule => "0 3 * * *"  
    statement => "SELECT * from EMP where WHERE ts > :sql_last_value"  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TraCom



JDBC Input Plugin - Configuration

- CRON style scheduling
 - This job will be run at 3 AM everyday

```
input {  
  jdbc {  
    jdbc_driver_library => "/path/to/mysql-connector-java-bin.jar"  
    jdbc_driver_class => "com.mysql.jdbc.Driver"  
    jdbc_connection_string => "jdbc:mysql://localhost:3306/people_db"  
    jdbc_user => "logstash_service"  
    schedule => "0 3 * * *"  
    statement => "SELECT * from EMP where WHERE ts > :sql_last_value"  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TraCom



JDBC Input Plugin - Configuration

- **sql_last_value** stored as metadata file
 - Updated on every run

```
input {  
  jdbc {  
    jdbc_driver_library => "/path/to/mysql-connector-java-bin.jar"  
    jdbc_driver_class => "com.mysql.jdbc.Driver"  
    jdbc_connection_string => "jdbc:mysql://localhost:3306/people_db"  
    jdbc_user => "logstash_service"  
    schedule => "0 3 * * *"  
    statement => "SELECT * from EMP where WHERE ts > :sql_last_value"  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TraCom



JDBC Input Plugin - Configuration

- Can also be passed-in from a file
- Used when the SQL statement is large or cumbersome
- Use **statement_filepath**

```
input {  
    jdbc {  
        jdbc_driver_library => "/path/to/mysql-connector-bin.jar"  
        jdbc_driver_class => "com.mysql.jdbc.Driver"  
        jdbc_connection_string => "jdbc:mysql://localhost:3306/people_db"  
        jdbc_user => "logstash_service"  
        schedule => "0 3 * * *"  
        statement_filepath => "/usr/share/mysql/queries/big_query.sql"  
    }  
}
```

JDBC Input Plugin - Handling Failures

- Handling connection failures:
 - **connection_retry_attempts** (default 1)
 - maximum number of times to try connecting to database
 - **connection_retry_attempts_wait_time** (default 0.5)
 - number of seconds to sleep between connection attempts

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

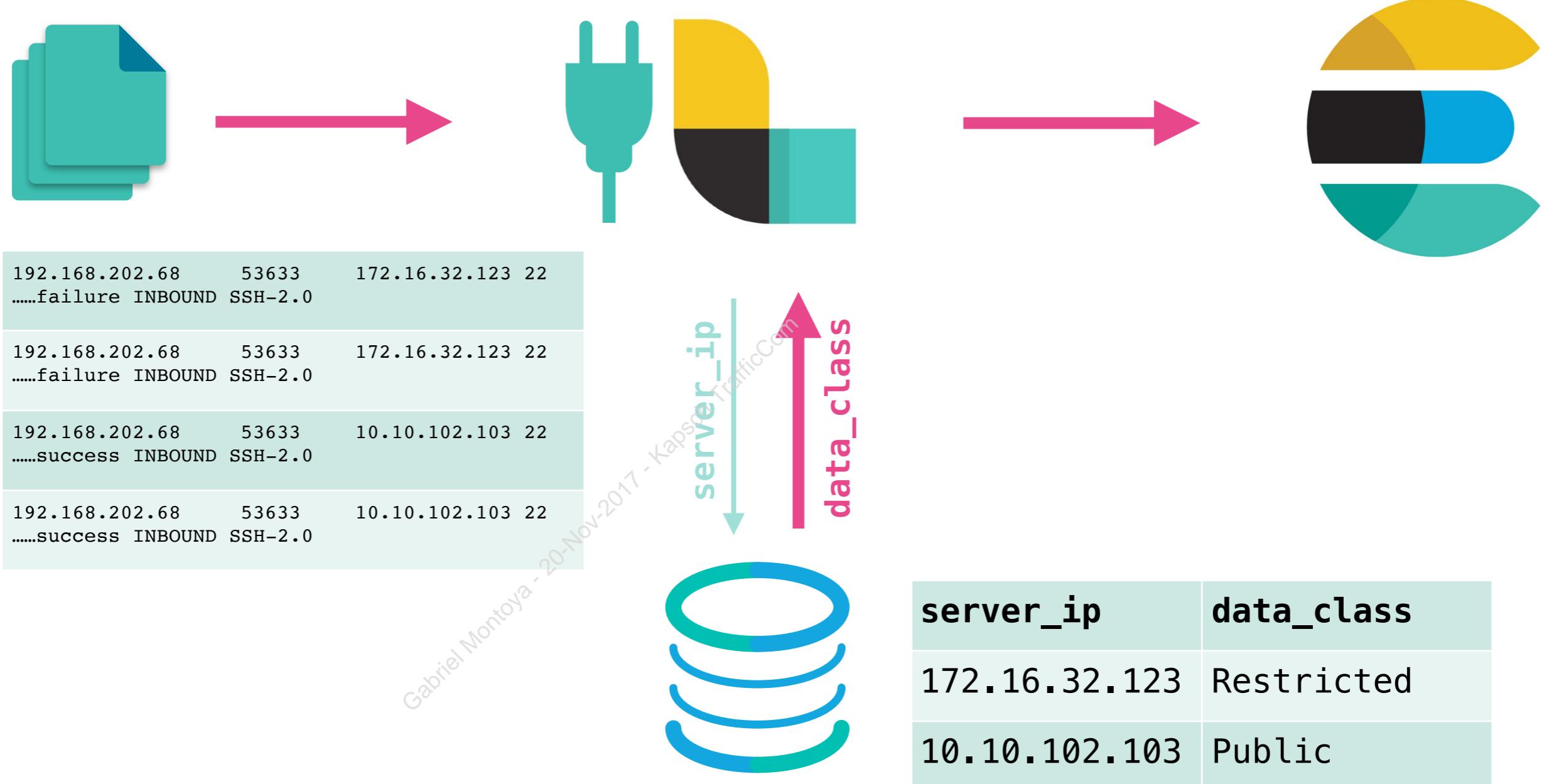
JDBC Streaming Filter Plugin

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



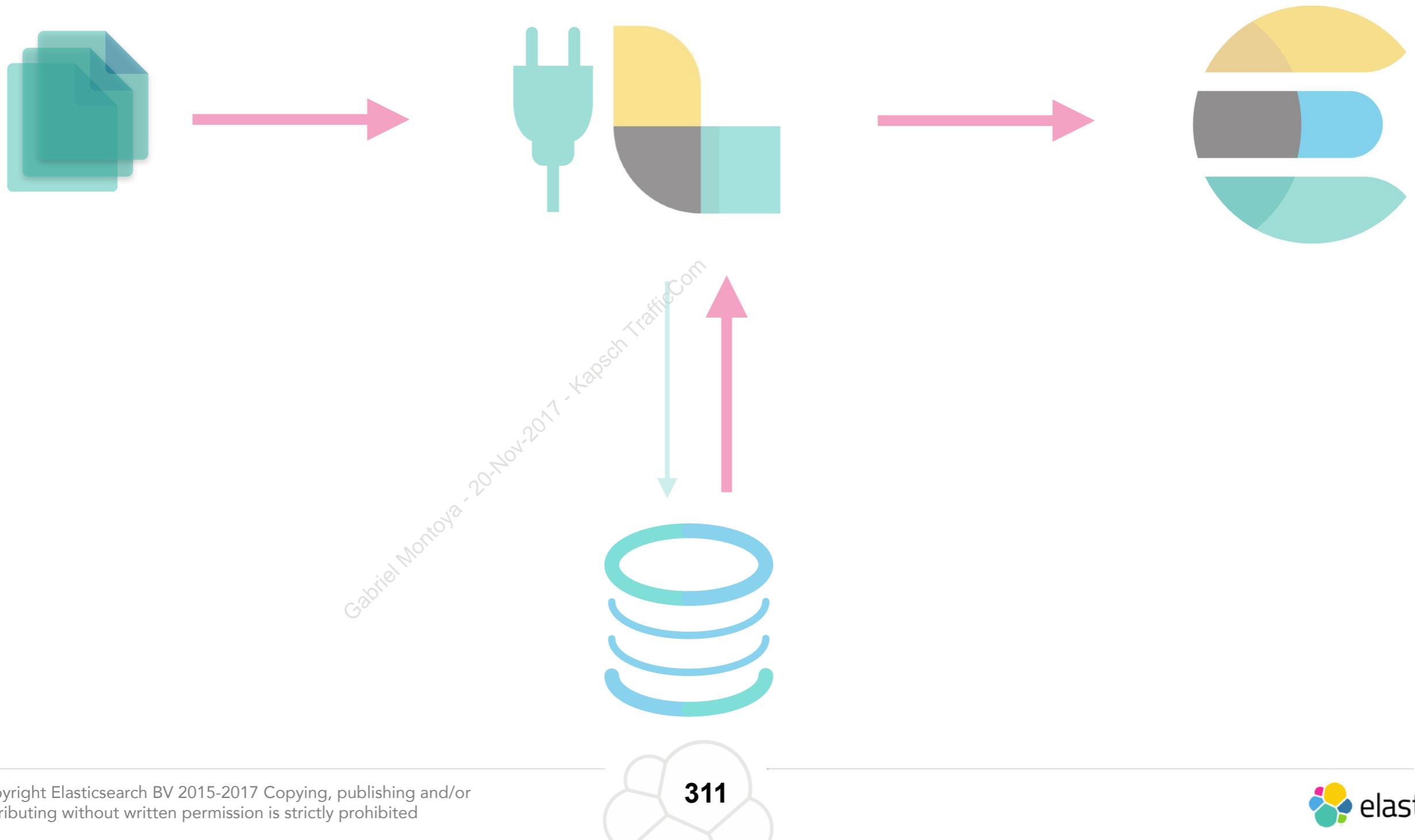
JDBC Streaming Filter Plugin

Lookup and enrich another data source



JDBC Streaming Filter Plugin - Installation

```
bin/logstash-plugin install logstash-filter-jdbc_streaming
```



JDBC Streaming Filter Plugin - Configuration

- Extracted result(s) stored in "target" field
 - field is overwritten if already exists

```
filter {  
    jdbc_streaming {  
        jdbc_driver_library => "/path/to/mysql-connector-java-bin.jar"  
        jdbc_driver_class => "com.mysql.jdbc.Driver"  
        jdbc_connection_string => "jdbc:mysql://localhost:3306/asset_db"  
        jdbc_user => "logstash_service"  
        jdbc_password => "secret"  
        statement => "select * from ASSET.SERVERS WHERE ip = :dest_ip"  
        parameters => { "server_ip" => "dest_ip" }  
        target => "data_class"  
    }  
}
```

Gabriel Montoya - 20-Nov-2017 - KapschTronic.com

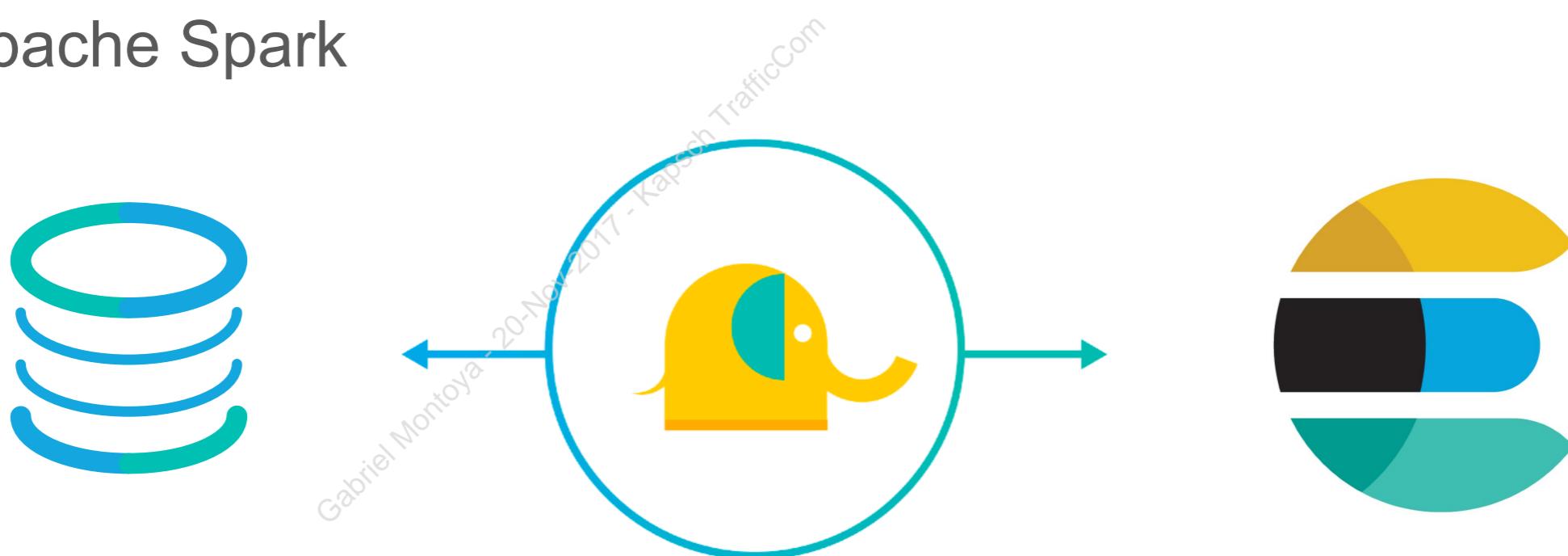
Elasticsearch-Hadoop

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



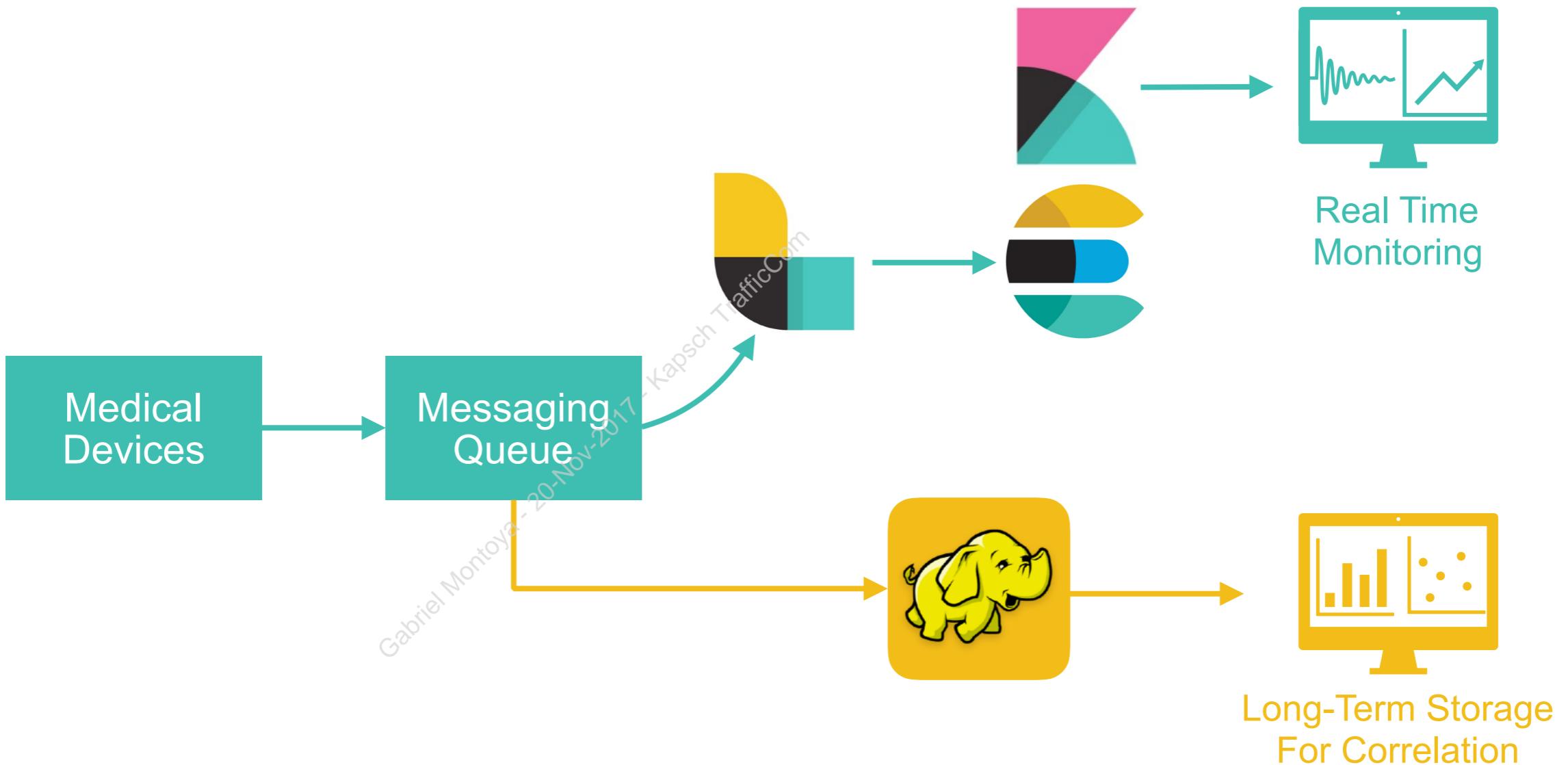
ES-Hadoop - Overview

- Open-source library
- Facilitates bidirectional data flow
- Supports:
 - Map/Reduce
 - Hive, Pig, Cascading
 - Apache Spark



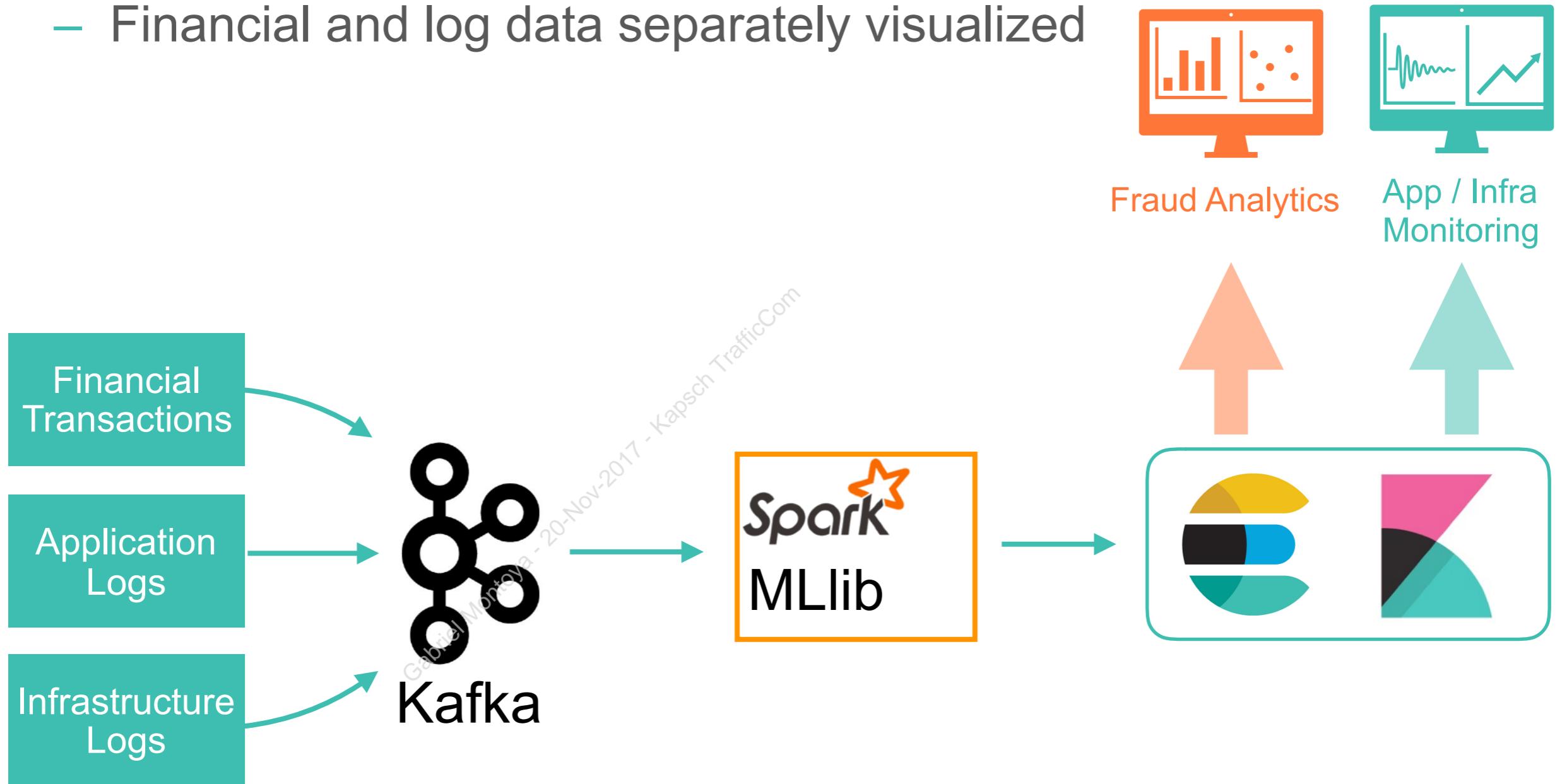
Use Cases

- Medical Device Data
 - Subset of data for real-time monitoring in Elasticsearch
 - Subset of data for long-term computations/correlation in Hadoop



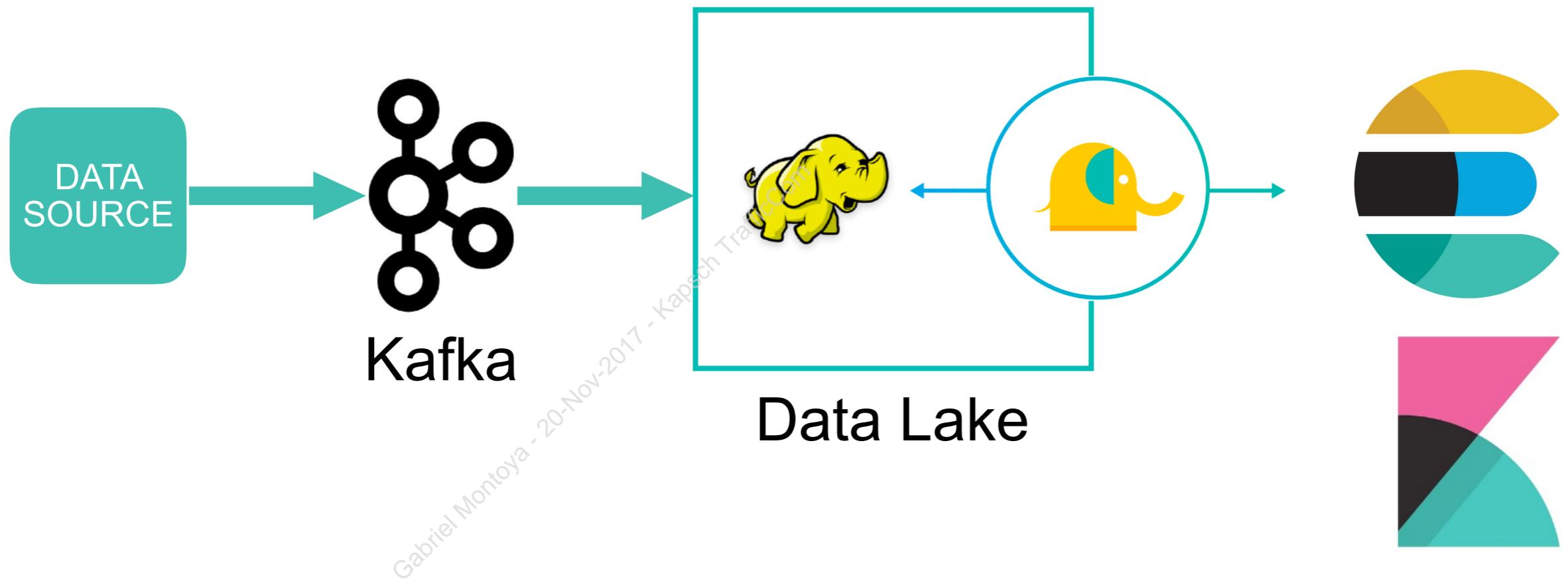
Use Cases

- Financial Services Data
 - Spark machine-learned result set is fed to ES
 - Financial and log data separately visualized



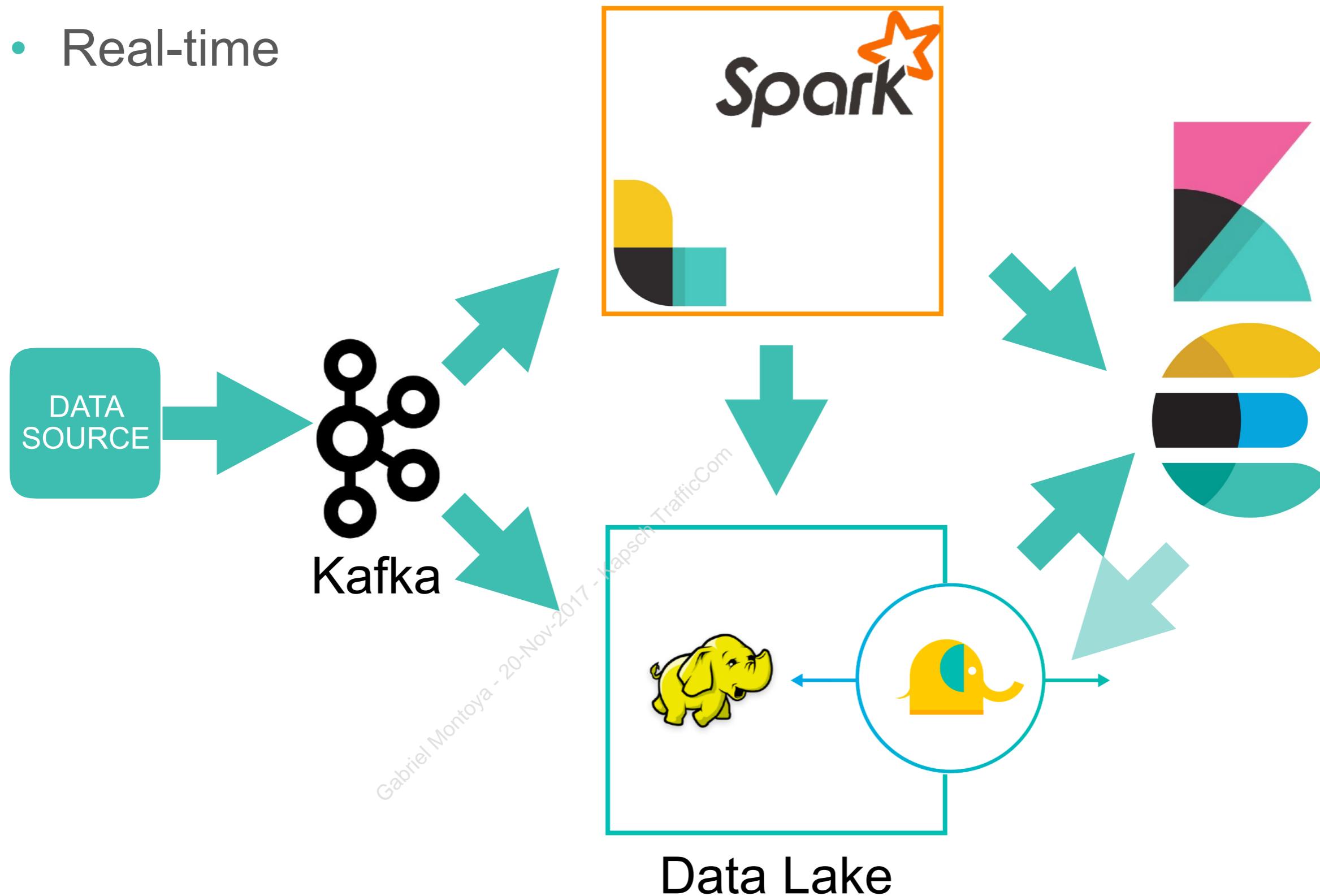
ES-Hadoop Architectures

- Land and forward
 - All data lands in Hadoop data lake, and a subset gets forwarded to Elasticsearch for serving

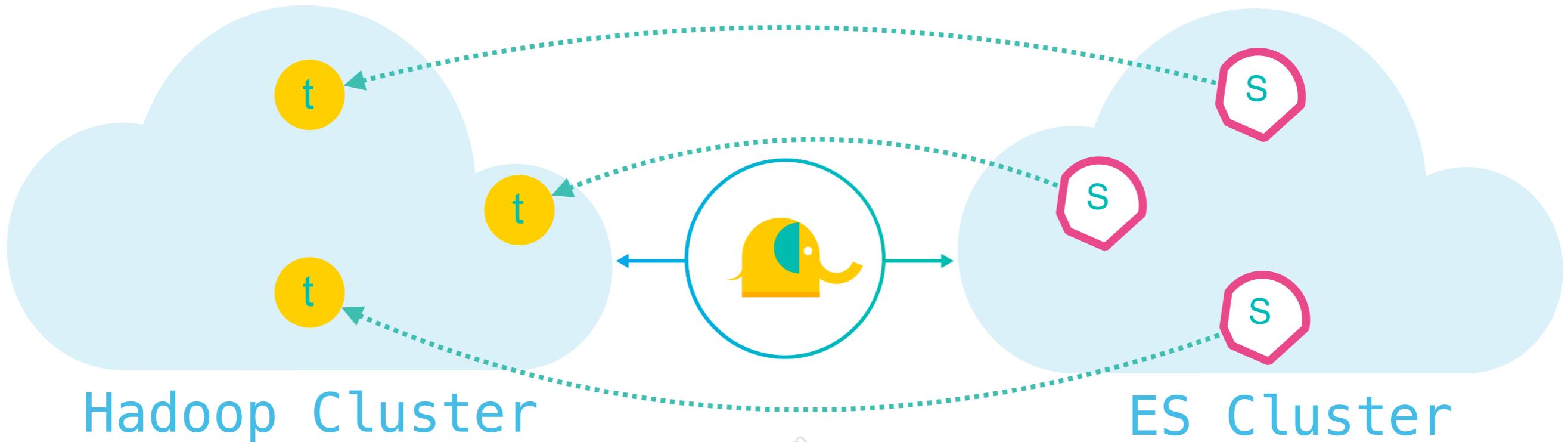


ES-Hadoop Architectures

- Real-time

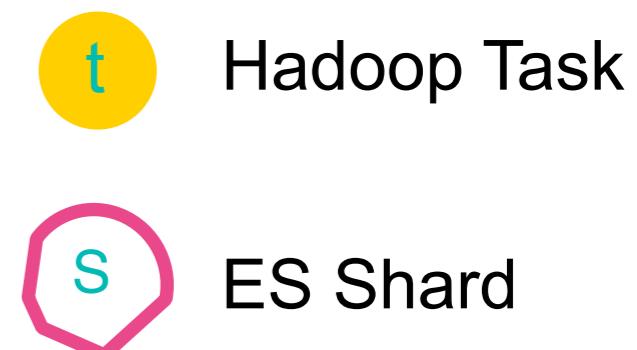


Reading from Elasticsearch

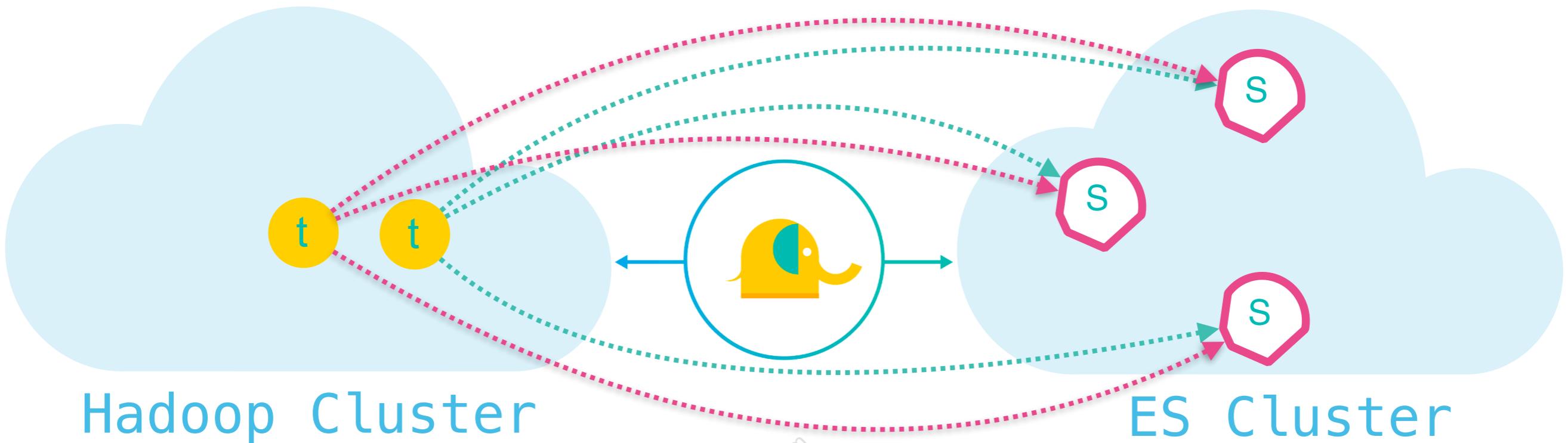


Reading from Elasticsearch

- ES-Hadoop will:
 - detect the number of shards (primary & replica) in query index
 - create one task (Hadoop split / spark partition) per shard



Writing to Elasticsearch



Writing to Elasticsearch

- ES-Hadoop will :
 - detect number of primary shards for write index
 - distribute writes between them
 - more splits = more parallel writes



Installation

- Zip File
- Maven-Compatible Tools

```
<dependency>
  <groupId>org.elasticsearch</groupId>
  <artifactId>elasticsearch-hadoop</artifactId>
  <version>5.6.0</version>
</dependency>
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Installing Minimal Binaries

```
<dependency>
  <groupId>org.elasticsearch</groupId>
  <b><artifactId>elasticsearch-hadoop</artifactId></b>
  <version>5.6.0</version>
</dependency>
```

Map/Reduce	<artifactId>elasticsearch-hadoop-mr</artifactId>
Hive	<artifactId>elasticsearch-hadoop-hive</artifactId>
Pig	<artifactId>elasticsearch-hadoop-pig</artifactId>
Spark	<artifactId>Elasticsearch-hadoop-spark-2.0_2.10</artifactId>



Configuration

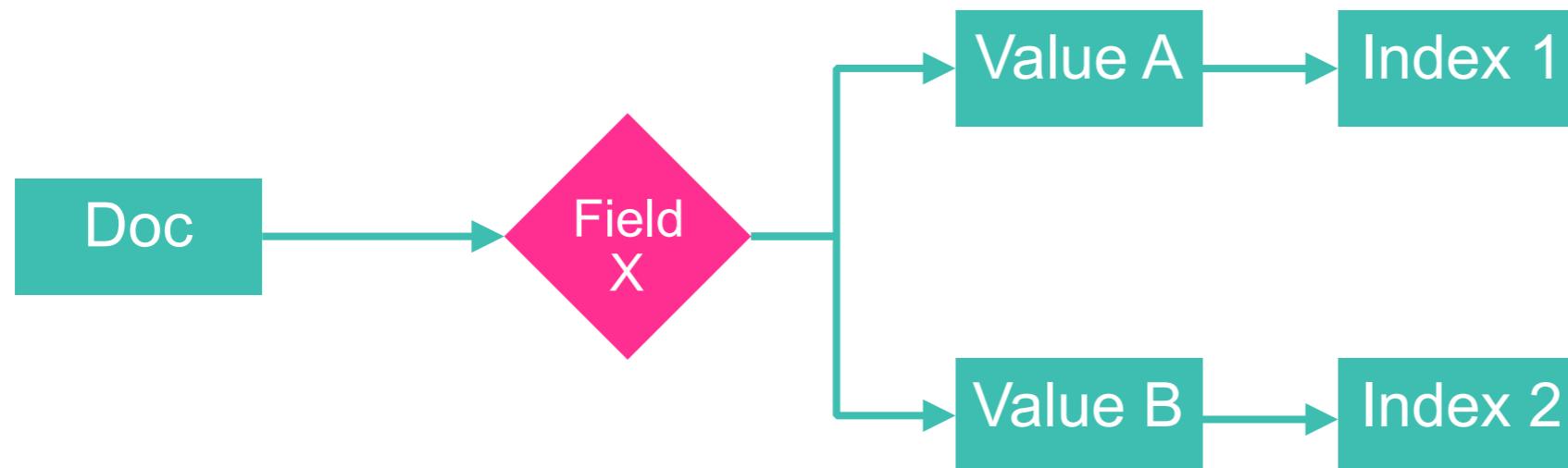
- Required Settings

<i>Setting</i>	<i>Description</i>
es.resource	Elasticsearch Resource Location <index>/<type>
es.resource.read	Resource for reading
es.resource.write	Resource for writing

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Content Based Routing

Based on the content of document (before Indexing) route to a particular index



```
# index the documents based on their type  
es.resource.write = my-collection/{media_type}
```

```
{  
  "media_type": "game",  
  "title": "Final Fantasy VI",  
  "year": "1994"  
}
```



my-collection/game

```
{  
  "media_type": "book",  
  "title": "Harry Potter",  
  "year": "2010"  
}
```



my-collection/book

Data Integrity

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Verifying Data Integrity

- Types of integrity checks
 - **Document existence:** Do all of documents exist in the index?
 - Check one-by-one
 - Batch processing
 - Multi-get API
 - Search
 - Aggregations
 - **Document content:** Were the contents indexed accurately?
 - Scroll through data
 - Take control of versioning

<https://www.elastic.co/blog/elasticsearch-verifying-data-integrity-with-external-data-stores>

Verifying Existence

- Checking one-by-one

```
HEAD /my_index/my_type/my_id1
```

- Found

```
HTTP/1.1 200 OK
Content-Type: text/plain; charset=UTF-8
Content-Length: 0
```

- Not Found

```
HTTP/1.1 404 Not Found
Content-Type: text/plain; charset=UTF-8
Content-Length: 0
```

Verifying Existence

- Batch Processing

- Multi-GET API

```
GET /my_index/my_type/_mget
{
  "ids": [ "my_id1", "my_id2" ]
}
```

- Search

```
GET /my_index/my_type/_search
{
  "size": 2,
  "query": {
    "ids": {
      "values": [ "my_id1", "my_id2" ]
    }
  }
}
```

Verifying Existence using Aggregation

- Least Expensive Method
- Requires planned structuring of data
 - Requires a numeric sequential ID
 - Can be `_id` or a field
 - Reindex if numeric sequential ID doesn't exist

Procedure

1. Setup integer-based key

```
POST /my_index/my_type/my_id1
{
  "id": 1,
  ...
}
```

Gabriel Montaraz - 20-Nov-2017 - Kapsch TrafficCom

Verifying Existence using Aggregation

2. Aggregate and eliminate buckets with data

```
GET /my_index/my_type/_search
{
  "size": 0,
  "aggs": {
    "find_missing_ids": {
      "histogram": {
        "field": "id",
        "interval": 1,
        "min_doc_count": 0
      },
      "aggs": {
        "remove_existing_bucket_selector": {
          "bucket_selector": {
            "buckets_path": {
              "count": "_count"
            },
            "script": {
              "inline": "count == 0",
              "lang": "expression"
            }
          }
        }
      }
    }
  }
}
```

Create Histogram on "id"

Remove buckets with data

Response

```
{
  "took": 4,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 5,
    "max_score": 0,
    "hits": []
  },
  "aggregations": {
    "find_missing_ids": {
      "buckets": [
        {
          "key": 4,
          "doc_count": 0
        }
      ]
    }
  }
}
```



Verifying Document Content

- Easier on Elasticsearch but higher client effort
- Options
 - Scroll: use `_scroll` API to iterate through data 1
 - Control versioning 2
 - Use your own version numbers
 - Implement strict versioning control process
 - Can side-step full document verification

```
GET /my_index/my_type/_search?scroll=1m
{
  "sort": [
    "_doc"
  ]
}
```

```
GET /my_index/my_type/_search
{
  "version": true
}
```

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- Logstash provides myriads of input plugins to ingest data
- ***JDBC input plugin*** is used to ingest data from any database that has a JDBC interface on a scheduled basis
- ***JDBC streaming filter plugin*** is used to lookup values from RDBMS in order to enrich another incoming data source
- ***Elasticsearch-Hadoop*** (ES-Hadoop) provides bidirectional data flow between Hadoop/Spark and Elasticsearch
- ES-Hadoop maximizes parallelism by automatically detecting # of shards and creating Hadoop *splits* or Spark *partitions*
- Elasticsearch lacks transaction support, however data integrity of indexed data can be verified using creative solutions

Quiz

- 1. True or False:** JDBC Input plugin can be installed and used without Logstash.
- 2. True or False:** JDBC Input plugin can be scheduled using CRON style configuration.
- 3. True or False:** JDBC Streaming Filter Plugin should be configured in the input section of the Logstash config file.
- 4. True or False:** Elasticsearch-Hadoop creates one split for every shard found in the read index to maximize parallelism.
- 5. The least expensive method of creating histograms and eliminating buckets with no data requires _____ Numeric ID.**

Gabriel Montoya - 20-Nov-2017 Kapsch TechCom



Lab 7

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 8

Network Monitoring

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Packetbeat Overview
- Protocols
- Examining a Protocol
- Packetbeat Configuration
- Flows
- Data Processing
- Production Settings

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Packetbeat Overview

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

What is Packetbeat?

- Lightweight data shipper that captures data from application layer protocols
- Capture network traffic with minimal to no overhead
- Not in the application request/response path so no risk to breaking running applications
- Can record network traffic at OS or hardware level

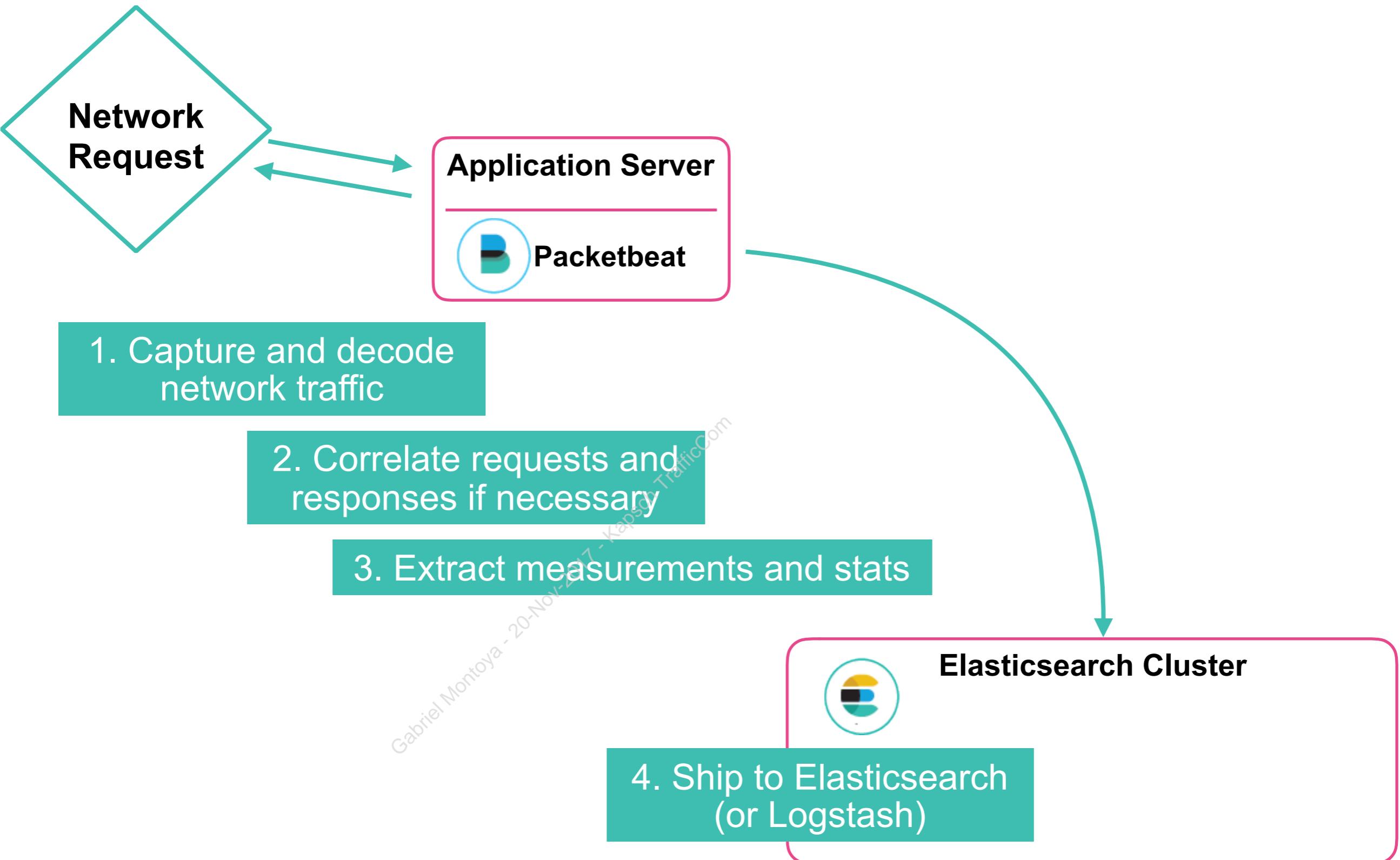
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Packetbeat Use Cases

- Security
 - Intrusion detection
 - Protocol violations
- Troubleshooting
 - Application network communications
 - Network issues
- Performance Analysis
 - Record network performance and detect bottlenecks
 - Understand how much time is spent communicating between networked processes and applications

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

How Packetbeat Works

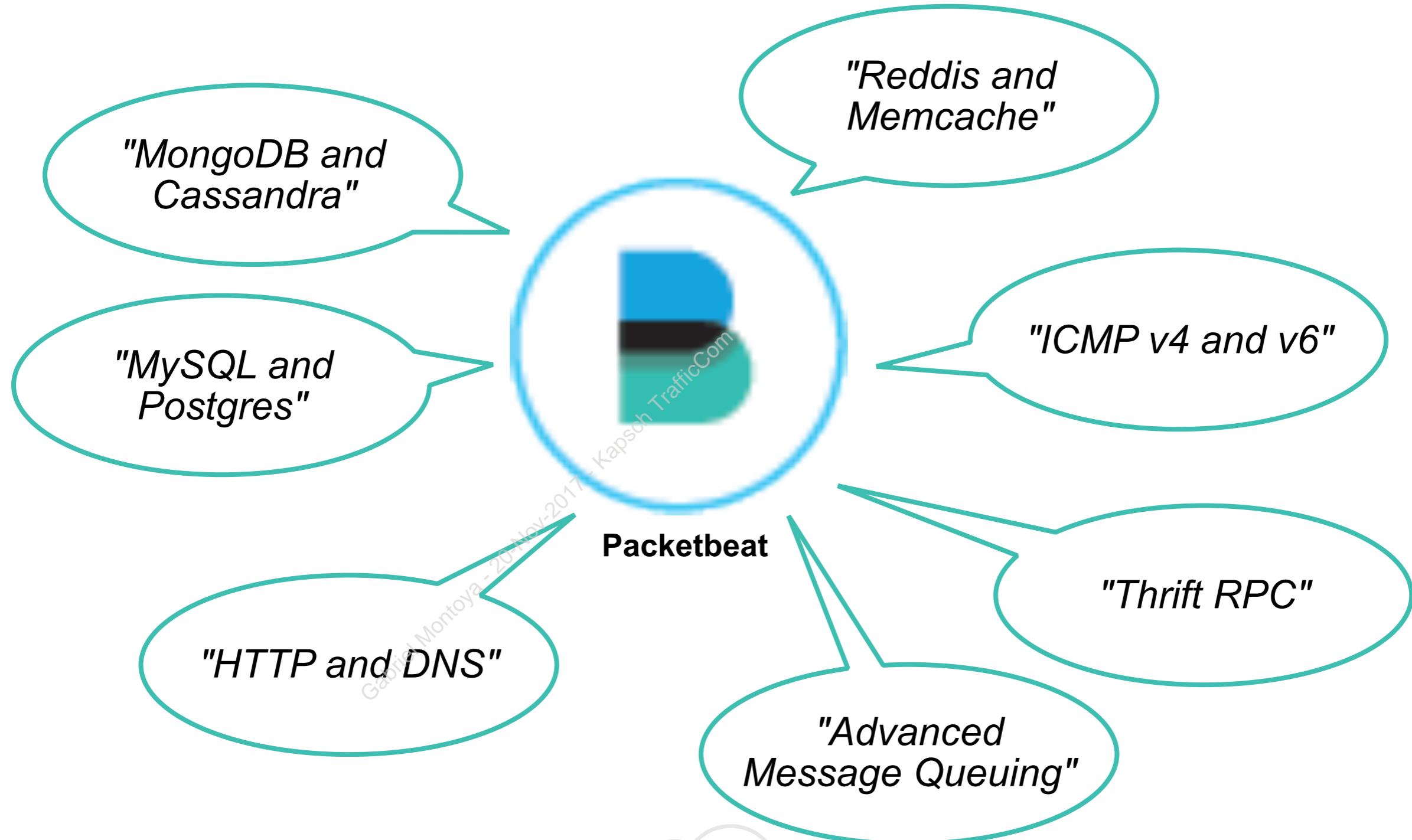


Protocols

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Many Built-in Protocols

- Packetbeat has pre-configured protocols to capture information from specific applications



Protocols

- Each protocol will capture data specific to that protocol implementation

This is a *single* HTTP request captured

They can be *configured* to capture or omit specific data

① @timestamp	⌚⌚⌚⌚⌚ * September 25th 2017, 17:15:00.074
t _id	⌚⌚⌚⌚⌚ * aKTkul4BAvdsnAP2prIE
t _index	⌚⌚⌚⌚⌚ * packetbeat-6.0.0-beta2-2017.09.25
# _score	⌚⌚⌚⌚ * -
t _type	⌚⌚⌚⌚ * doc
t beat.hostname	⌚⌚⌚⌚ * localhost
t beat.name	⌚⌚⌚⌚ * localhost
t beat.version	⌚⌚⌚⌚ * 6.0.0-beta2
# bytes_in	⌚⌚⌚⌚ * 74
# bytes_out	⌚⌚⌚⌚ * 418
t client_ip	⌚⌚⌚⌚ * 2604:2000:1444:3f:e44e:f761:38c4:c2dc
t client_port	⌚⌚⌚⌚ * 61021
t client_proc	⌚⌚⌚⌚ *
t client_server	⌚⌚⌚⌚ *
t direction	⌚⌚⌚⌚ * out
? http.request.headers.content-length	⌚⌚⌚⌚ * ⚠ 0
? http.request.headers.user-agent	⌚⌚⌚⌚ * ⚠ curl/7.43.0
t http.request.params	⌚⌚⌚⌚ *
t http.response.body	⌚⌚⌚⌚ * <html><head><title>301 Moved Permanently</title></head><body bgcolor="white"><center><h1>301 Moved Permanently</h1></center><hr><center>nginx/1.4.6 (Ubuntu)</center></body></html>
t http.response.code	⌚⌚⌚⌚ * 301
? http.response.headers.content-length	⌚⌚⌚⌚ * ⚠ 193

Protocol Built-in Dashboards

Navigation

Packetbeat:

Overview

Flows

Web transactions

MySQL performance

PostgreSQL performance

MongoDB performance

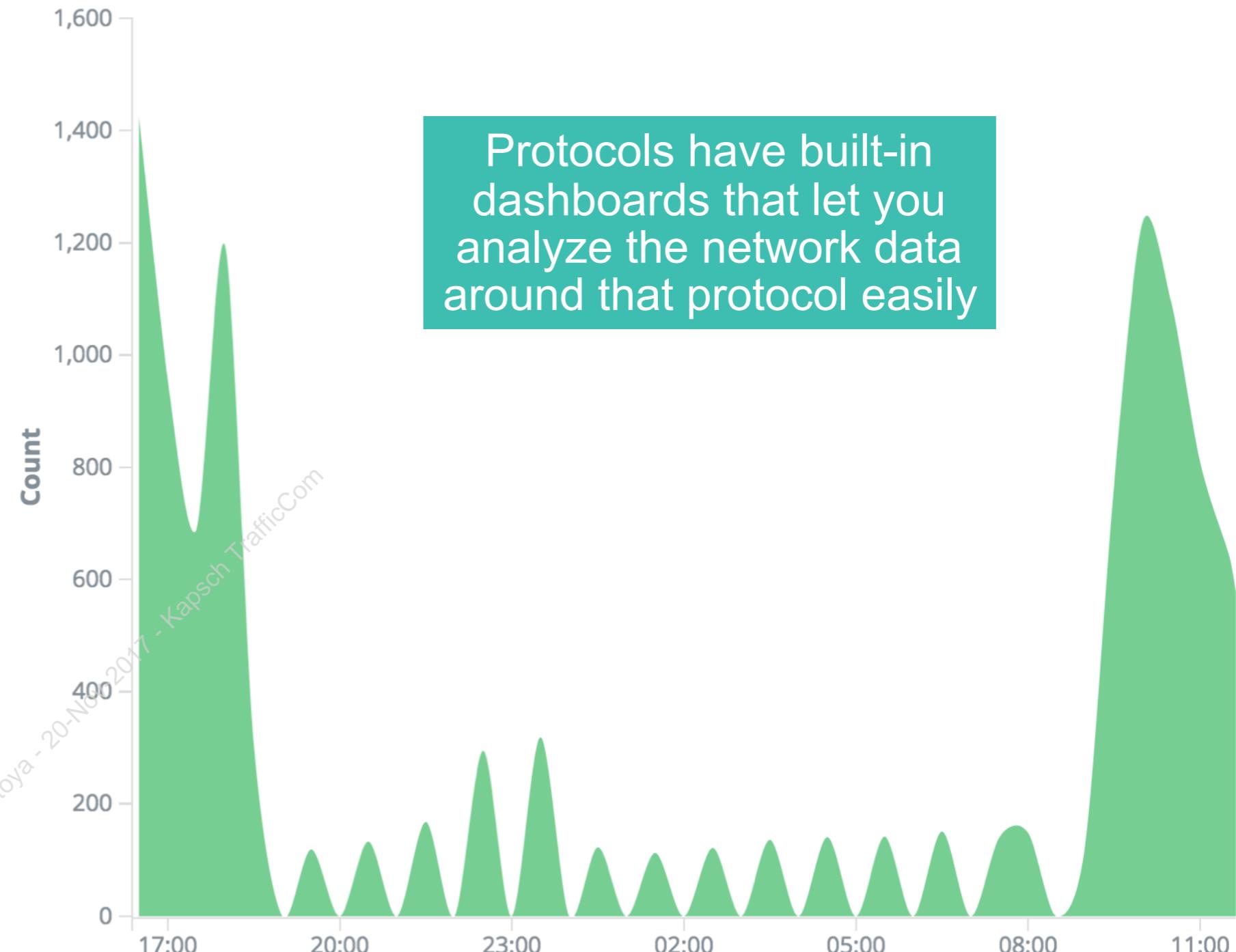
Thrift-RPC performance

NFS transactions

Cassandra performance

HINT: There are
Packetbeat visualizations
that are not used - make
your own dashboards from
them!

Connections over time



Protocols have built-in dashboards that let you analyze the network data around that protocol easily

Examining a Protocol

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

HTTP Protocol

- The **HTTP** protocol is one of many built-in protocols Packetbeat can capture data on
 - Useful when HTTP data is flowing through a system that does not contain the logs
 - Can capture additional information such as the headers and response body
- Configured in the Transaction Protocols part of the config

```
#=====Transaction protocols=====

packetbeat.protocols:

  type: http
  ports: [80, 8080]
  send_response: true
  include_body_for: ["text/html"]
```

HTTP Document

www.elastic.co?pass=abc123

Request parameters captured - can we filter them?

The entire response - include_body_for allowed this

# bytes_in	Q Q D * 90B
# bytes_out	Q Q D * 568B
t client_ip	Q Q D * 2604:2000:1444:3f:29c7:4a48:115f:5759
t client_port	Q Q D * 52544
t client_proc	Q Q D *
t client_server	Q Q D *
t direction	Q Q D * out
# http.request.headers.content-length	Q Q D * 0
t http.request.headers.user-agent	Q Q D * curl/7.43.0
t http.request.params	Q Q D * pass=abc123
t http.response.body	Q Q D * <html><head><title>301 Moved Permanently</title></head><body bgcolor="white"><center><h1>301 Moved Permanently</h1></center><hr><center>CloudFront</center></body></html>
t http.response.code	Q Q D * 301
# http.response.headers.content-length	Q Q D * 183
t http.response.headers.content-type	Q Q D * text/html
t http.response.phrase	Q Q D * Permanently
t ip	Q Q D * 2600:9000:201c:9c00:8:8236:ce00:93a1
t method	Q Q D * GET

Hiding Fields

- We can restrict certain request parameters from being indexed
 - Sensitive data
 - Uninteresting parameters

```
#=====Transaction Protocols=====

packetbeat.protocols:

type: http
ports: [80, 8080]
send_response: true
include_body_for: ["text/html"]
hide_keywords: ["pass", "opts"]
```

↑
hide_keywords blocks
these request parameters



Keywords Hidden

- Now these values are not sent to Elasticsearch

www.elastic.co?pass=abc123

t http.request.params

⊕ Q Q □ * pass=xxxxx

t http.response.body

⊕ Q Q □ * <html>
<head><title>301 Moved Permanently</title></head>
<body bgcolor="white">
<center><h1>301 Moved Permanently</h1></center>
<hr><center>CloudFront</center>
</body>
</html>

t http.response.code

⊕ Q Q □ * 301

>Password is now
blocked out

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Secure URL's

- Trying to access <http://www.elastic.co> resulted in a 301 response code because <elastic.co> uses HTTPS - what if we decided to go to <https://www.elastic.co>?
 - We can't capture the HTTPS traffic because it is encrypted - Packetbeat wouldn't be able to do anything with it
- We can still capture some information about the request but it won't be via HTTP - it will use a "*Flow*" which we will discuss later

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Packetbeat Configuration

Gabriel Montoya - 20-Nov-2017 - KapschTrafficCom

Devices Overview

- Before Packetbeat can listen for network traffic you need to tell Packetbeat which devices to listen on:

```
sudo ./packetbeat devices
```

Packetbeat can tell you which devices are available

```
0: en0 (No description available) (fe80::6e40:8ff:fe8f:19c2 10.0.1.23  
2604:2000:1444:3f:6e40:8ff:fe8f:19c2 2604:2000:1444:3f:e44e:f761:38c4:c2dc)  
1: bridge0 (No description available) (Not assigned ip address)  
2: awdl0 (No description available) (fe80::e8b8:43ff:fe0e:f59f)  
3: en1 (No description available) (Not assigned ip address)  
4: en2 (No description available) (Not assigned ip address)  
5: p2p0 (No description available) (Not assigned ip address)  
6: lo0 (No description available) (::1 127.0.0.1 fe80::1)
```

- We can see there are 7 devices available - **en0** is the *default* network device. **lo0** is the loopback - *localhost*

Devices Configuration

- Network devices are configured in the *packetbeat.yml* file:

```
#=====Network device=====#
# Select the network interface to sniff the data. On Linux, you
# can use the "any" keyword to sniff on all connected interfaces.

packetbeat.interfaces.device: en0
```

- On Linux you can set this value to "any"
 - will listen on *all* active devices/interfaces
- You must ensure that the user that Packetbeat is run as **owns** the configuration file
- Remember, if running locally if you are bound to **en0** then **lo0** traffic won't be captured and vice versa



Traffic Capture Options

- Packetbeat "sniffs" the traffic based on how you configure it:
 - **pcap**: uses a shared library for sniffing network traffic. *libpcap* must be installed as a kernel module and often is by default
 - **af_packet**: *Linux only option* that uses memory mapped sniffing. This is the better option as it has less overhead. Linux memory-maps the packet data into a circular buffer that Packetbeat can read

```
#=====Network device=====#
# Select the network interface to sniff the data. On Linux, you
# can use the "any" keyword to sniff on all connected interfaces.

packetbeat.interfaces.device: en0
packetbeat.interfaces.type: af_packet
```

Protocols

- Each protocol can be configured individually and disabled if required:

Protocols can have additional options set

Ports to listen on

Capture the payload

Hide query parameters

```
#=====Transaction protocols=====#
packetbeat.protocols:
  - type: icmp
    # Enable ICMPv4 and ICMPv6 monitoring. Default: false
    enabled: true

  - type: http
    ports: [80, 8080, 8000, 5000, 8002]
    split_cookie: true
    include_body_for: ["text/html"]
    send_headers: ["User-Agent", "Cookie", "Set-Cookie"]
    real_ip_header: "X-Forwarded-For"
    hide_keywords: ["pass", "password", "pwd"]
```

Outputs

- You can send the data to **Elasticsearch or Logstash**:
 - Kafka, Redis and File are additional outputs

```
#----- Elasticsearch output -----  
  
output.elasticsearch:  
  # Array of hosts to connect to.  
  hosts: ["localhost:9200"]  
  
  # Optional protocol and basic auth credentials.  
  #protocol: "https"  
  #username: "packet_beat"  
  #password: "Q#$*hhaT3%9e!"  
  pipeline: my_pipeline
```

You can transform with an Elasticsearch Pipeline as well if you're using an ingest node

General Configuration Options

- General data you can set for each Packetbeat that is deployed:

```
name: "pb-007"
```

Name each instance - default is host name

```
tags: ["dev-ops", "hardware", "prod"]
```

Add tags to the Beat to group them together

```
fields: {department: "dev", instance-id: "9736438723827287"}
```

Add fields to the documents that Packetbeat generates

Flows

- You can configure flows here or disable them all together:

```
#===== Flows =====#  
  
packetbeat.flows:  
# Set network flow timeout. Flow is killed if no packet is received  
# before being timed out.  
  
timeout: 30s  
  
# Configure reporting period. If set to -1, only killed flows will  
# be reported  
  
period: 10s
```

Lets explore Packetbeat Flows a bit more...

Flows

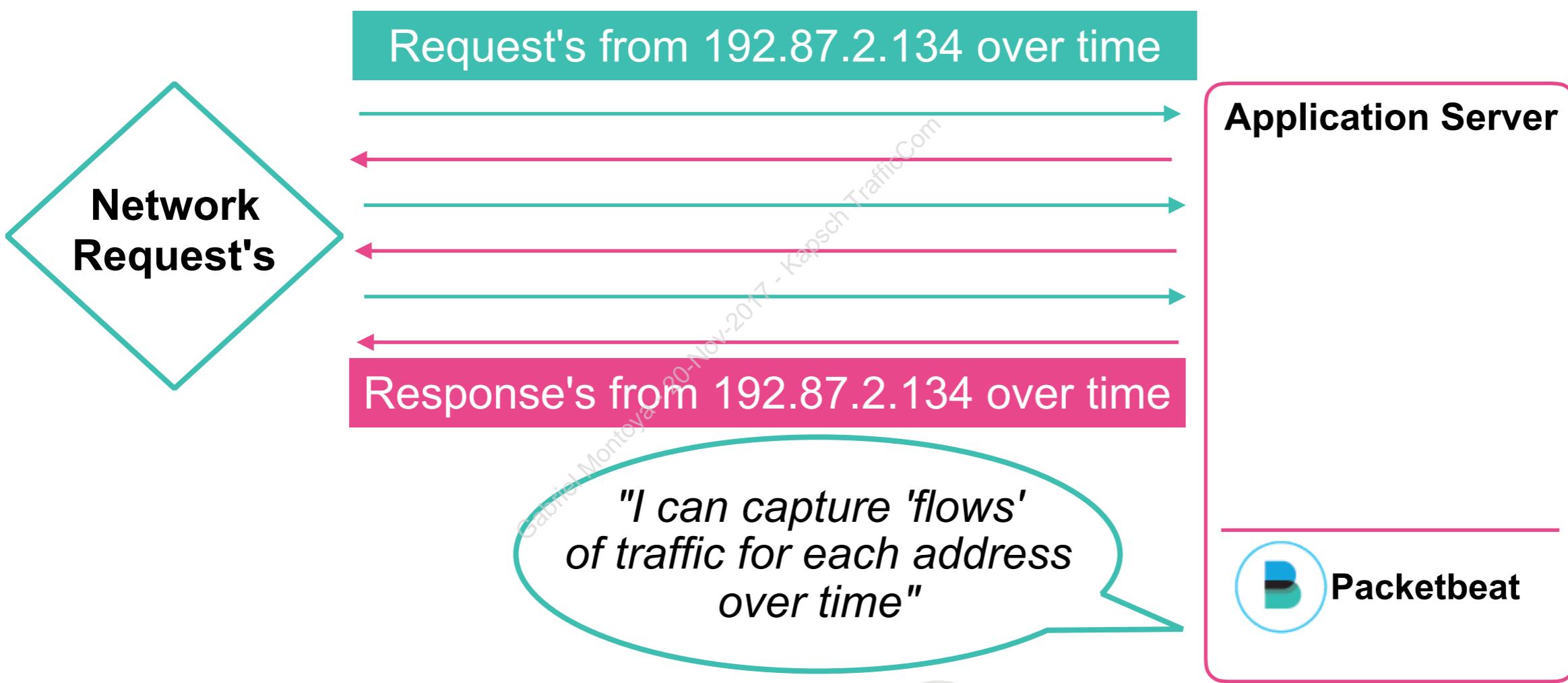
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Packetbeat Flows

- A **flow** is a group of packets sent over the same time period that share common properties, such as the same source and destination address and protocol
- We use flows to capture data about network protocols for which we don't understand the application layer protocol
 - TLS
 - Unsupported protocols
- Get data about IP/TCP/UDP layers
 - Number of packets, source and destination IP, total bytes
 - Retransmissions
 - Detect strange communications

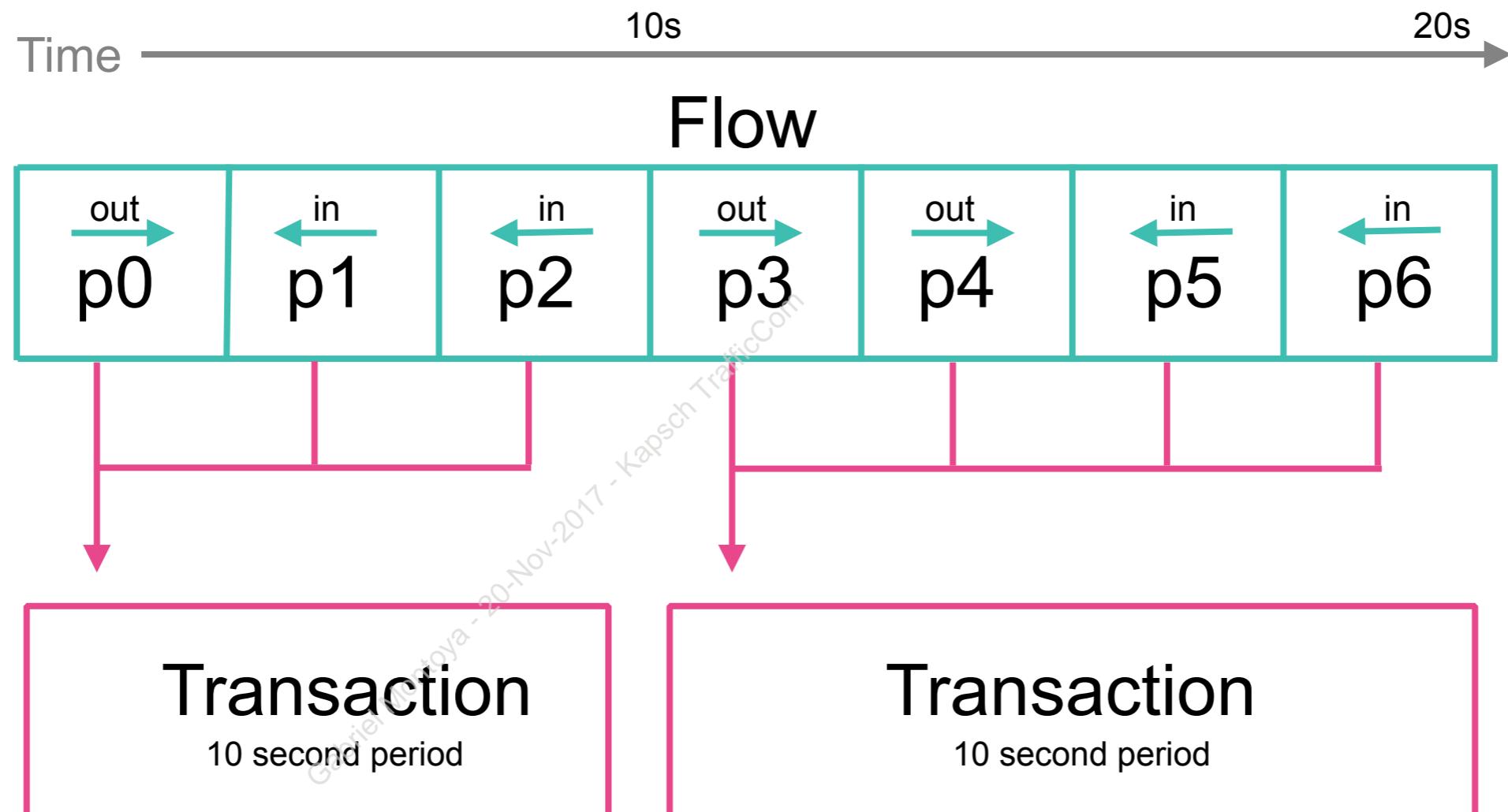
How Flows Work

- Captures "*network flows*" so you have record of the request's and the associated response's for each *address and protocol* pair over a configurable period
 - Ship directly to Elasticsearch or enrich with Logstash - like all Beats

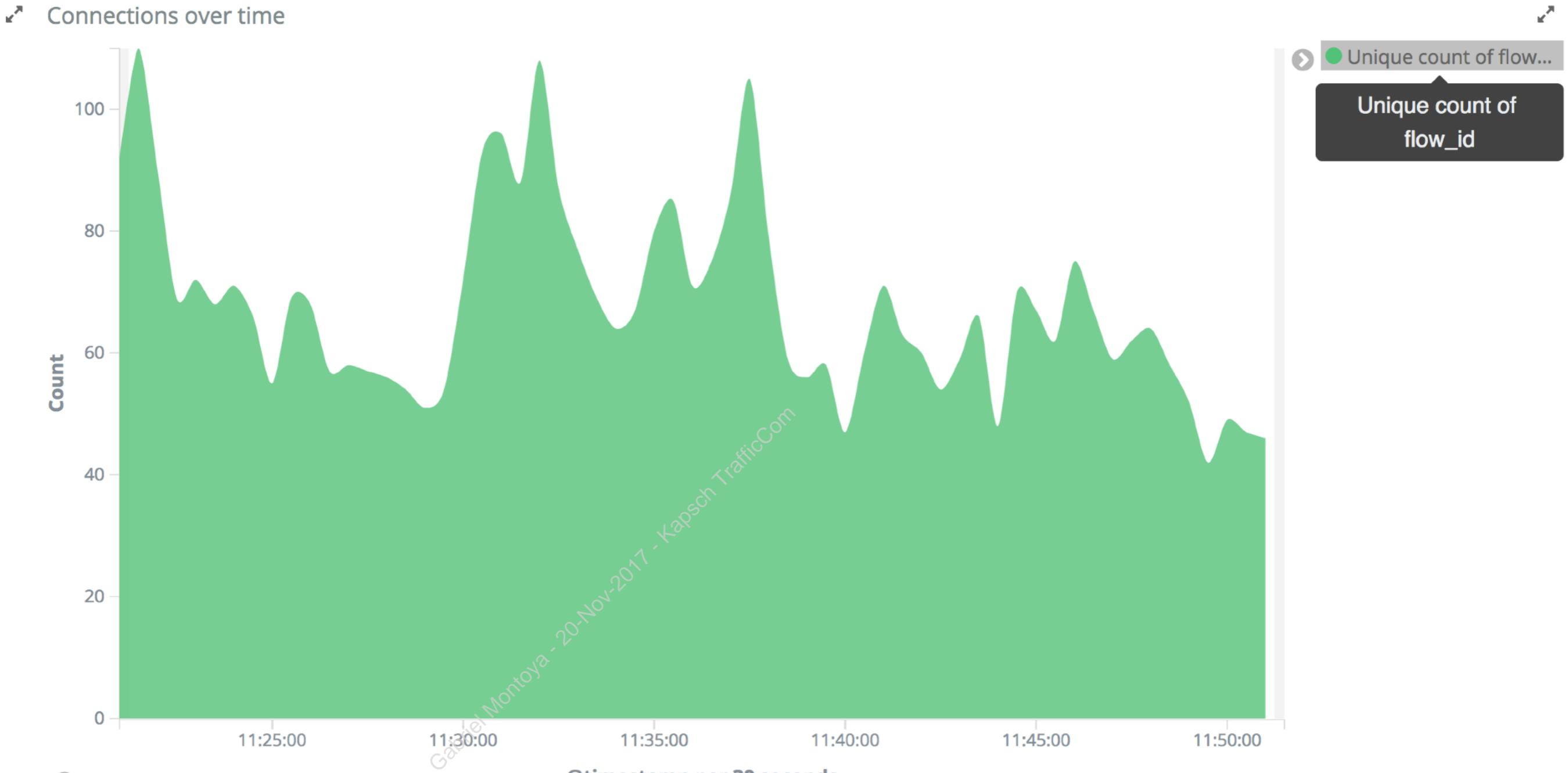


How a Flow is Generated

- Connection endpoint meta data: IP, Port
- Flow-ID based on connection endpoints
- Collects bi-directional metrics/summaries

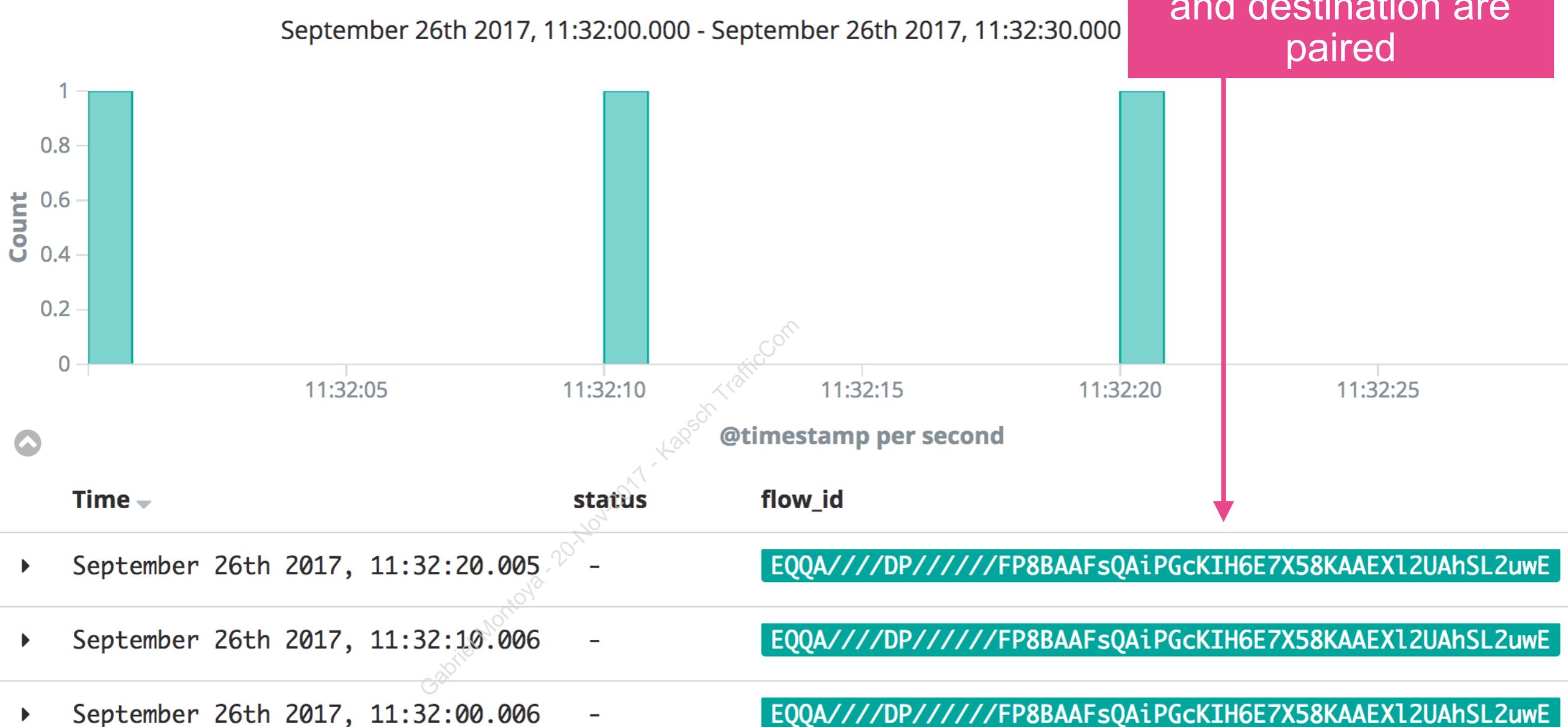


Flows Dashboard UI



UI displays all flows over time among other visualizations

Flows Discover UI



Flows Documents

- Flows can have 2 states: *final* set to **true** or **false**
 - *final* is set to true after the configured *timeout*

t dest.ip	Q Q □ * 151.101.0.133
t dest.mac	Q Q □ * 88:1f:a1:3b:5f:9f
t dest.port	Q Q □ * 443
# dest.stats.net_bytes_total	Q Q □ * 295
# dest.stats.net_packets_total	Q Q □ * 4
t final	Q Q □ * true
t flow_id	Q Q □ * EQQA///DP////FP8BAAFsQAIPGcKIH6E7X58KAAEXL2UAhSL2uwE
⌚ last_time	Q Q □ * September 26th 2017, 11:36:58.651
t source.ip	Q Q □ * 10.0.1.23
t source.mac	Q Q □ * 6c:40:08:8f:19:c2
t source.port	Q Q □ * 63010
⌚ start_time	Q Q □ * September 26th 2017, 11:35:42.185

start_time indicates the first flow for this *flow_id* - all flows with this id from *start_time* to until the *final=true* flow can be thought of as a continuous transmission

Flows Documents

- The series of flows represents the **CUMULATIVE SUM** of packets and bytes sent since the first. The "*final = true*" document is the final state!

First

```
# source.stats.net_bytes_total    * 54
```

```
# source.stats.net_packets_total    * 1
```

Second

...

```
# source.stats.net_bytes_total    * 108
```

```
# source.stats.net_packets_total    * 2
```

Last

```
# source.stats.net_bytes_total    * 282
```

```
# source.stats.net_packets_total    * 5
```

```
+ final                                 * true
```

If you want to aggregate on final flows, query for **final = true**!

Periods

- Each flow could have many documents describing it - but you may only want the final one that describes the entire transaction!

```
#===== Flows =====#  
  
packetbeat.flows:  
# Set network flow timeout. Flow is killed if no packet is received  
# before being timed out.  
  
    timeout: 30s  
  
# Configure reporting period. If set to -1, only killed flows will  
# be reported  
  
    period: -1
```

Set the **period to -1** and then only the final document for the flow will be indexed. This could save considerable space

Data Processing

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Dropping Documents

- If you know things about your data, like the port or source that could be sending and receiving data, you can use *processors* and *drop_event* to drop it
 - This helps filter noise from connections you don't want to monitor
 - especially useful when using Flows

If I know that 151.101.0.133 is my CDN then I can drop it...

```
processors:  
  - drop_event:  
    when:  
      equals:  
        dest.ip: "151.101.0.133"
```

This is the *processor_name*

This is the *condition*

Any flow with this destination IP address will
NOT be sent to Elasticsearch

Dropping http status codes of
200 might be a good idea!

Removing Fields

- Sometimes you want drop certain fields that are not required for your analysis

Keep all the fields except beat.version and http.request.params

```
processors:  
  - drop_fields:  
    fields: [ "beat.version", "http.request.params" ]
```

Can do the inverse where you specify which fields you want to keep and all others are removed. Only query and response will be indexed

```
processors:  
  - include_fields:  
    fields: [ "query", "response" ]
```

You could also use conditionals with these if you wanted

Production Settings

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Dropping Root

- Normally you'd have to start Packetbeat as root - but you can change permissions
 - You have to grant the packetbeat binary: cap_net_raw, cap_net_admin

```
sudo setcap cap_net_raw,cap_net_admin=eip packetbeat
```

Linux ONLY setting

Can run network related operations such as interface configuration and binding to any address

Runs as the restricted user but with a capability to send raw packets

Paths and Directories

- Packetbeat has default path locations which you can change if you'd like:

Default Paths

home: path.home

config: {path.home}

data: {path.home}/data

logs: {path.home}/logs

Config Setting

path.home

path.config

path.data

path.logs

Typical Linux Paths

/usr/share/packetbeat
/etc/packetbeat
/var/lib/packetbeat
/var/log/packetbeat

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- **Packetbeat** can capture network traffic across different devices
- There are various built-in protocols that structure documents in a sensible way
- Packetbeat can output to Elasticsearch and use Ingest Nodes as well as Logstash for further enhancement
- **Flows** can be used to capture network traffic for which a protocol is not available
- Flows are configured using a period and a timeout and the final = true flows contain information about the entire transaction
- You can use Packetbeat **Processors** to modify the data that is sent to Elasticsearch



Quiz

1. **True or False:** Packetbeat must always run as root
2. What field must be set to "true" to indicate that a flow has ended?
3. Which Packetbeat command line options will tell you which network devices are available to bind to?
4. **True or False:** All protocols are disabled by default
5. **True or False:** Packetbeat has a protocol specifically for listening to encrypted HTTP communications

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Lab 8

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 9

Data Ingestion Architectures

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Sizing Elasticsearch
- Sizing the Elastic Stack
- Elastic Stack Architectures
- Scaling
- Load Balancing
- Integrating Distributed Queues

Gabriel Montoya - 20-Nov-2017 - Kapsch TraumCom

Sizing Elasticsearch

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Sizing Elasticsearch

- Mostly out of scope for this training
- The *Elasticsearch Operations I* course goes into great detail
- But there are some important things to keep in mind...

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Remember Event-Based Data?

- Logs, social media streams, time-based events
- Timestamp + Data
- Do not change
- Typically search for recent events
- Older documents become less important
- Hard to predict the data size
- **Time-based Indices is a very good option**
 - create a new index each day, week, month, year, ...
 - search the indices you need in the same request
 - default for logstash and beats



Be careful

- The scaling unit is the shard
- But one should NOT have too many shards
 - specially sub-utilized shards
- Each shard uses resources from the machine
 - if you have too many it will seriously slow things down
- A shard can optimally hold from 10 to 100 gigabytes
 - optimal depends on the use case
- However, a shard with 1GB is sub-utilized!
- And if you have too many shards, Elasticsearch will be slow

Gabriel Montoya - 20-Nov-2017 - Kibana Traffic

Do not Overshard

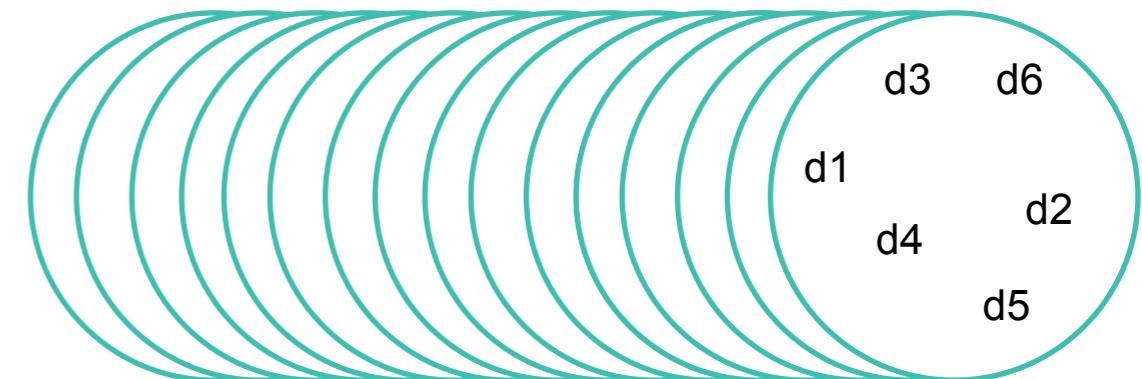
- 3 different logs
- 1 index per day each
- 5 shards (default)
- 6 months retention
- **~900 shards**
- 1GB each
- **~180GB**

too many shards
for no good reason!

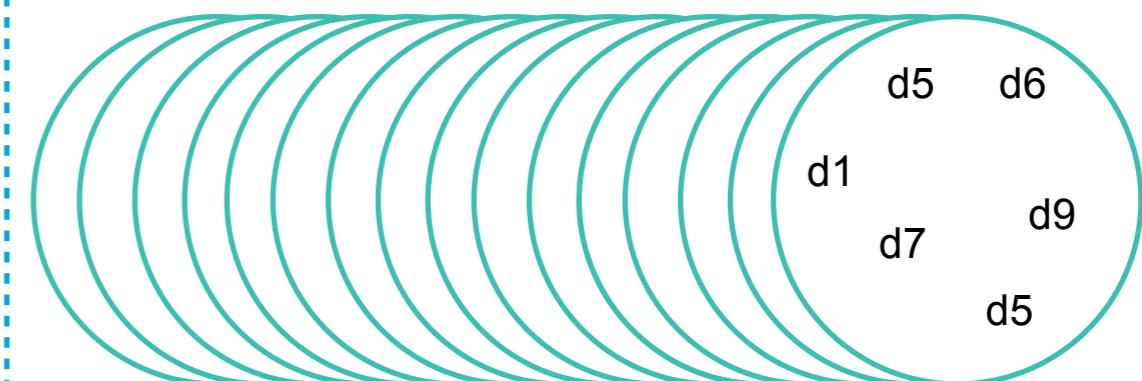
we could easily have
this data in 10 shards

Gabriel Montoya - 20-Nov-2017 - Kapsi - ThaliaCom

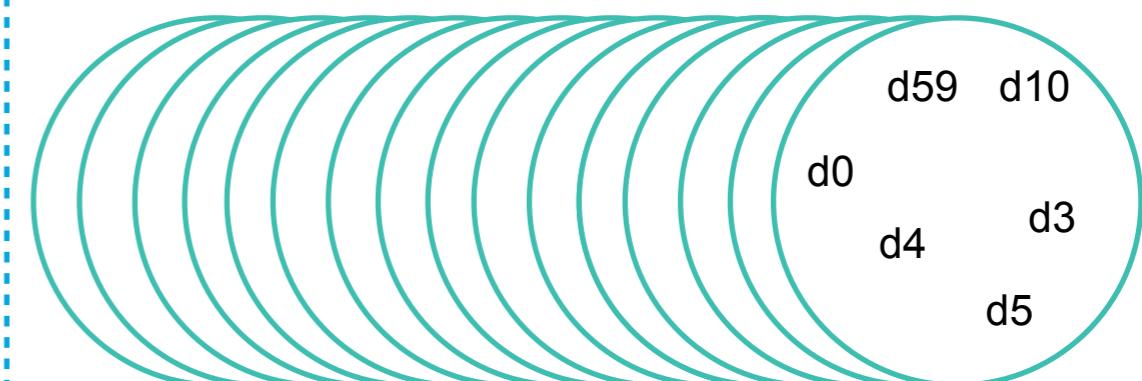
Cluster *my_cluster*



access-...



application-...



mysql-...



Do not overshard (sub-utilized shards)

- What is the problem?
 - # of shards per index * time-based window * retention period
- What can I do?
 - decrease the default number of shards (e.g. 1 instead of 5)
 - Beats (6.0+) uses 1 or 3 as the default depending on the project
 - increase the time-based window (e.g. month instead of days)
 - decrease the retention period
 - not likely, as this is a business requirement
- What if I am not sub-utilizing my shards (e.g. 50GB each)?
 - That is a more complex problem that is not the scope of this course. Probably the solution is to have multiple clusters.

Sizing the Elastic Stack

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom



Sizing the Elastic Stack

- Too many "It Depends!" ...
- Let's look at some of the greatest consumers of resources in the Elastic Stack...

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Sizing the Elastic Stack

- Greatest CPU consumers
 - Heavy indexing loads in ES
 - Complex analysis chains
 - ES scripting
 - ES ingest nodes
 - Logstash filters

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Sizing the Elastic Stack

- Greatest RAM consumers
 - sorting large datasets in Elasticsearch
 - complex aggregations in Elasticsearch
 - watch your garbage collection

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Sizing the Elastic Stack

- Greatest disk I/O consumers
 - Heavy indexing loads in Elasticsearch
 - Logstash file input
 - Logstash file output
 - Lucene merges

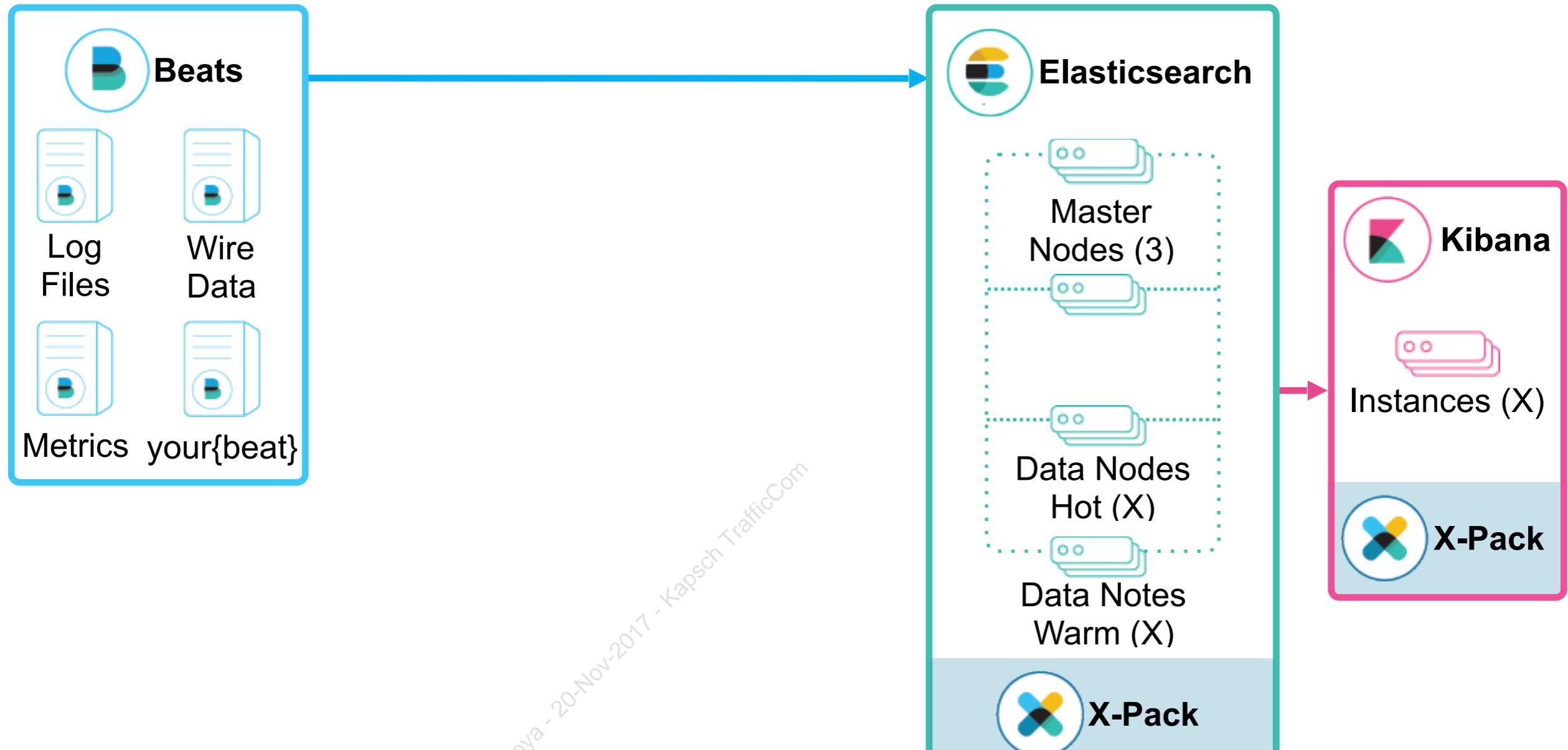
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Elastic Stack Architectures

Gabriel Montoya - 20-Nov-2017 - KapschTrafficCom



Beats -> Elasticsearch

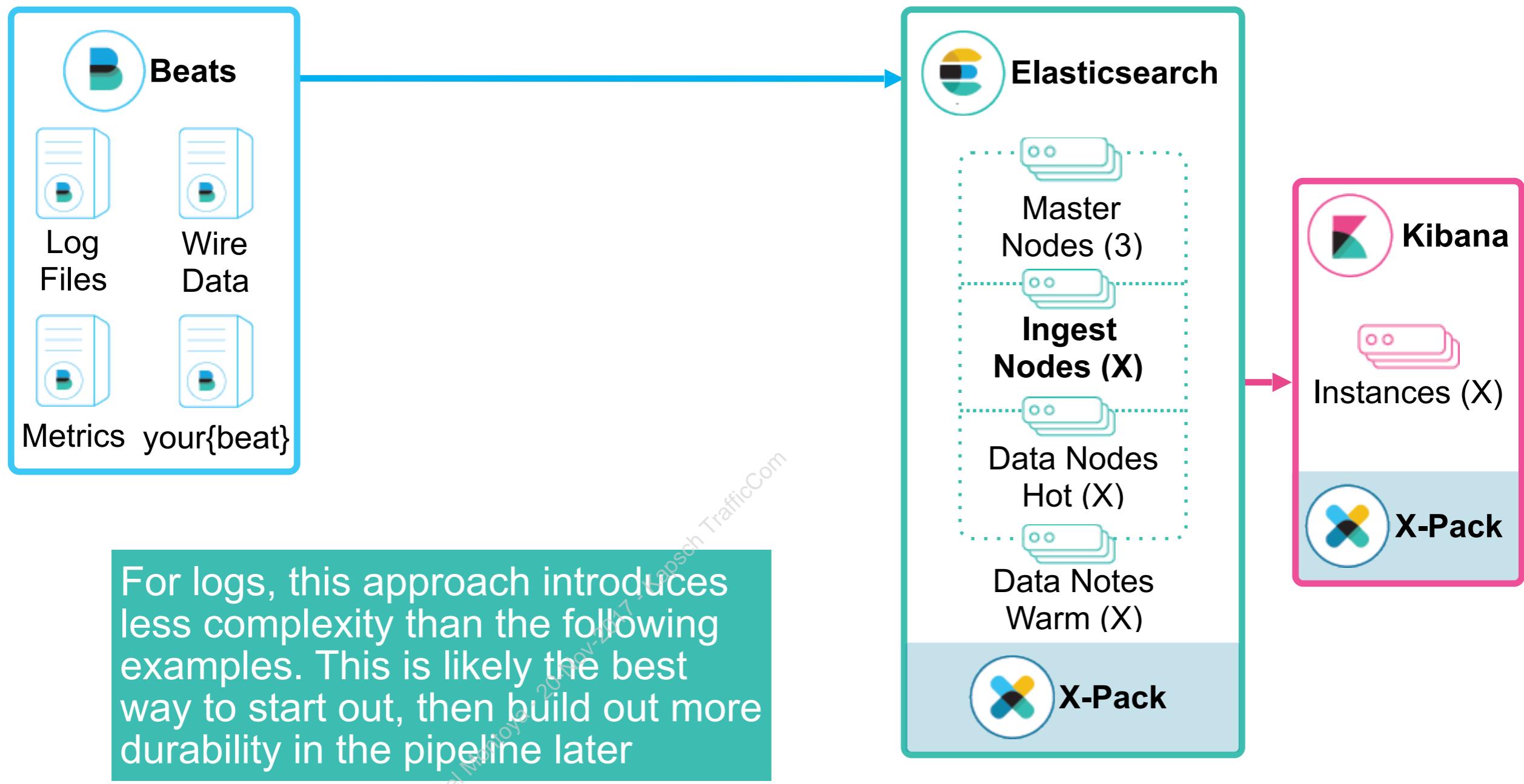


Beats -> Elasticsearch

- Simplest way to ingest data
 - lightweight agent (application-side)
 - lower fault tolerance
 - good for metrics
 - very simple architecture

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats -> [Ingest Node] Elasticsearch

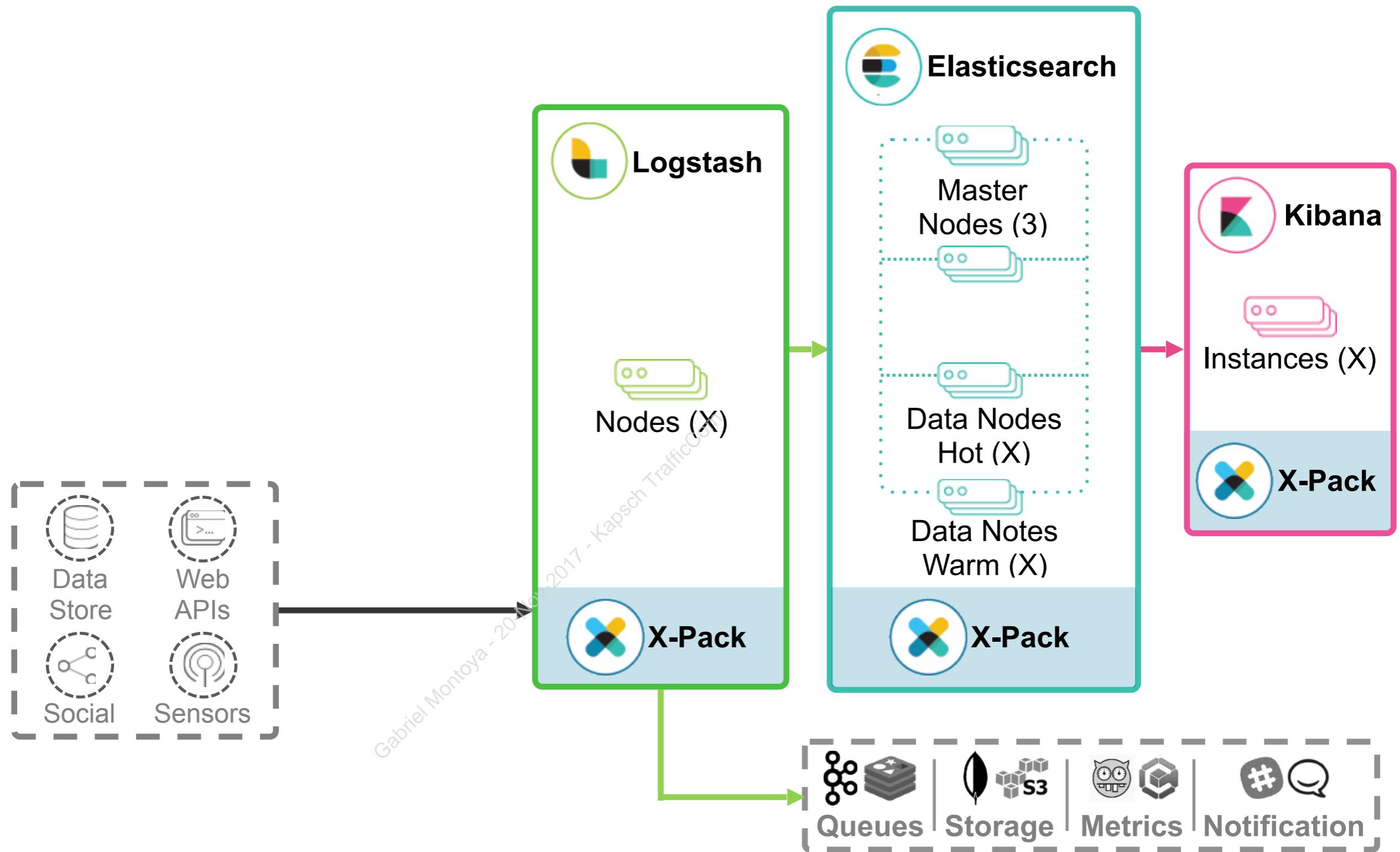


Beats -> [Ingest Node] Elasticsearch

- Simple way to ingest data
 - lightweight agent (application-side)
 - lower fault tolerance
 - limited inputs, outputs and processors
 - good for "simple" log events
 - simple architecture

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logstash -> Elasticsearch

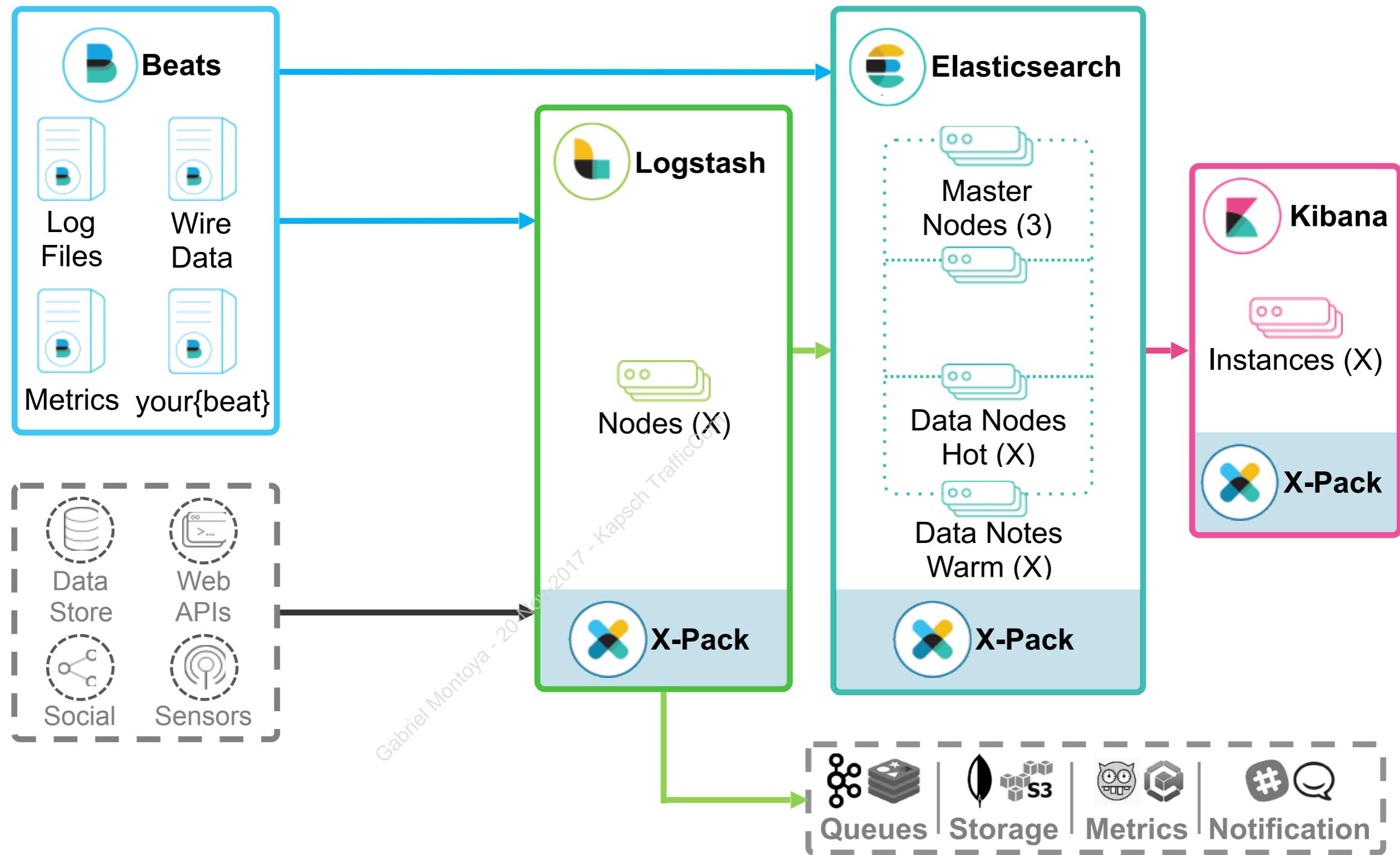


Logstash -> Elasticsearch

- Simple and flexible way to ingest data
 - heavyweight agent (server-side)
 - persistent queue
 - multiple inputs, outputs and processors
 - good for log events (where a heavyweight agent is ok)
 - simple architecture

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats -> Logstash -> Elasticsearch



Beats -> Logstash -> Elasticsearch

- Most flexible way to ingest data
 - lightweight agent
 - persistent queue
 - multiple inputs, outputs and processors
 - good for metrics and log events (simple and complex)
 - complex architecture

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats -> Logstash Configuration

- Use the Beats *logstash* output:

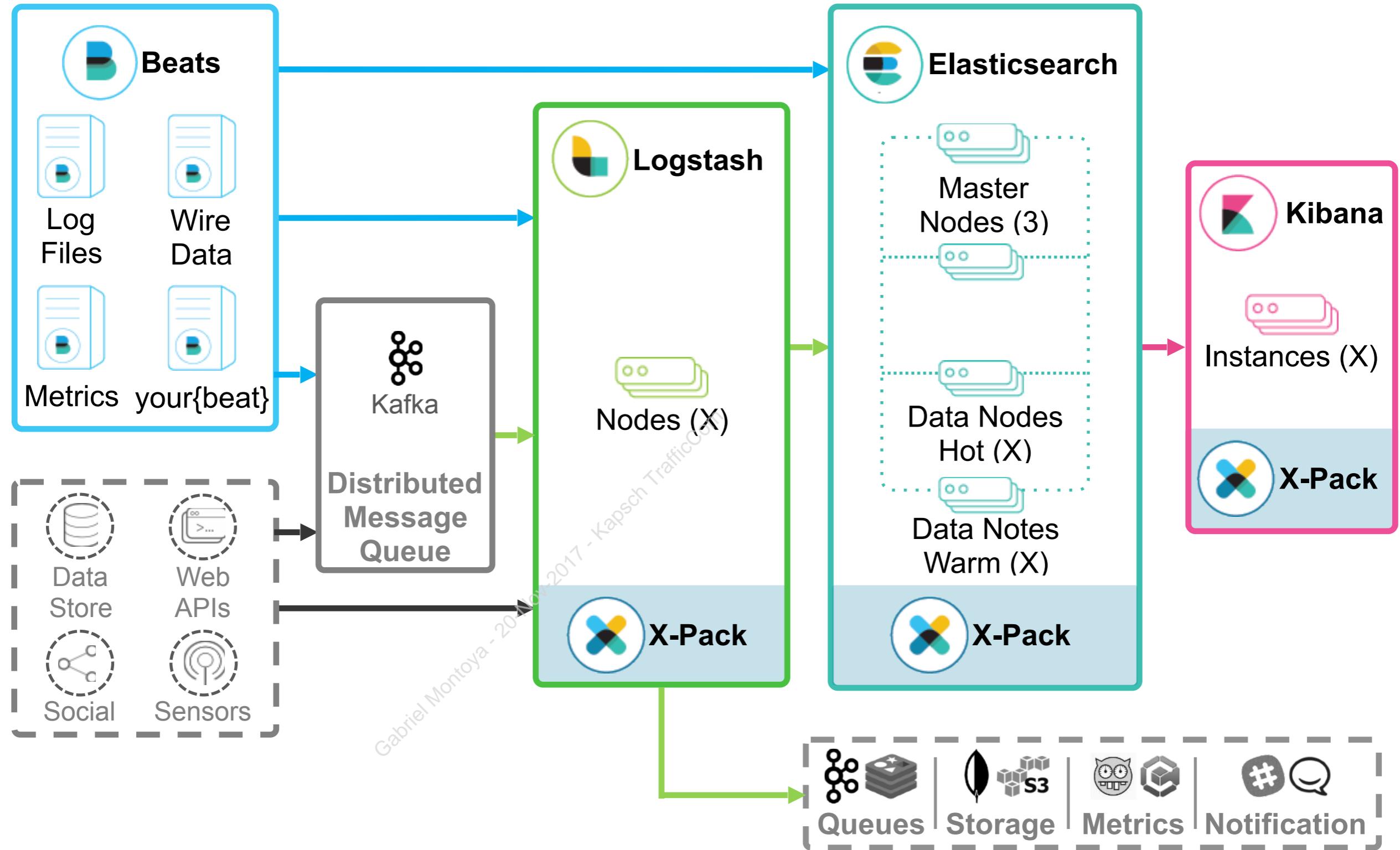
```
output.logstash:  
  hosts: ["localhost:5044"]
```

- Send to the Logstash *beats* input:

```
input {  
  beats {  
    port => 5044  
  }  
}
```

- Can be sent using TLS / certificates for security

Beats → Kafka/Redis → Logstash → Elasticsearch



Beats → Kafka/Redis → Logstash → Elasticsearch

- Most reliable way to ingest data
 - lightweight agent
 - distributed message queue
 - multiple inputs, outputs and processors
 - good for fault tolerant log event ingestion that need to scale
 - very complex architecture

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Scaling

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Scaling Elasticsearch

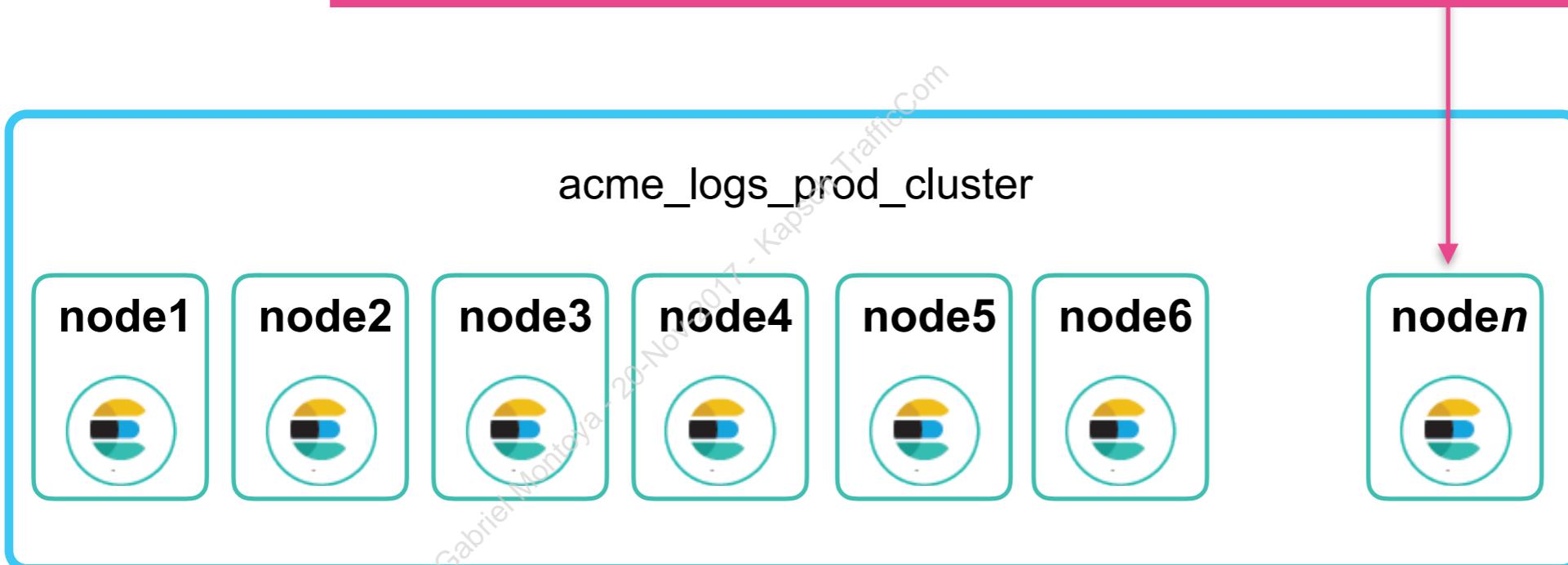
- Elasticsearch clusters are simple to scale
- Just add more nodes
- Typically you would need to scale data nodes (hot/warm nodes) and ingest nodes, but not master-eligible, or coordinating-only nodes.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Scaling Elasticsearch

- Elasticsearch clusters are simple to scale...
- Just add more nodes!

Simply bring a new Elasticsearch data node with the correct configuration, and your cluster will scale up!



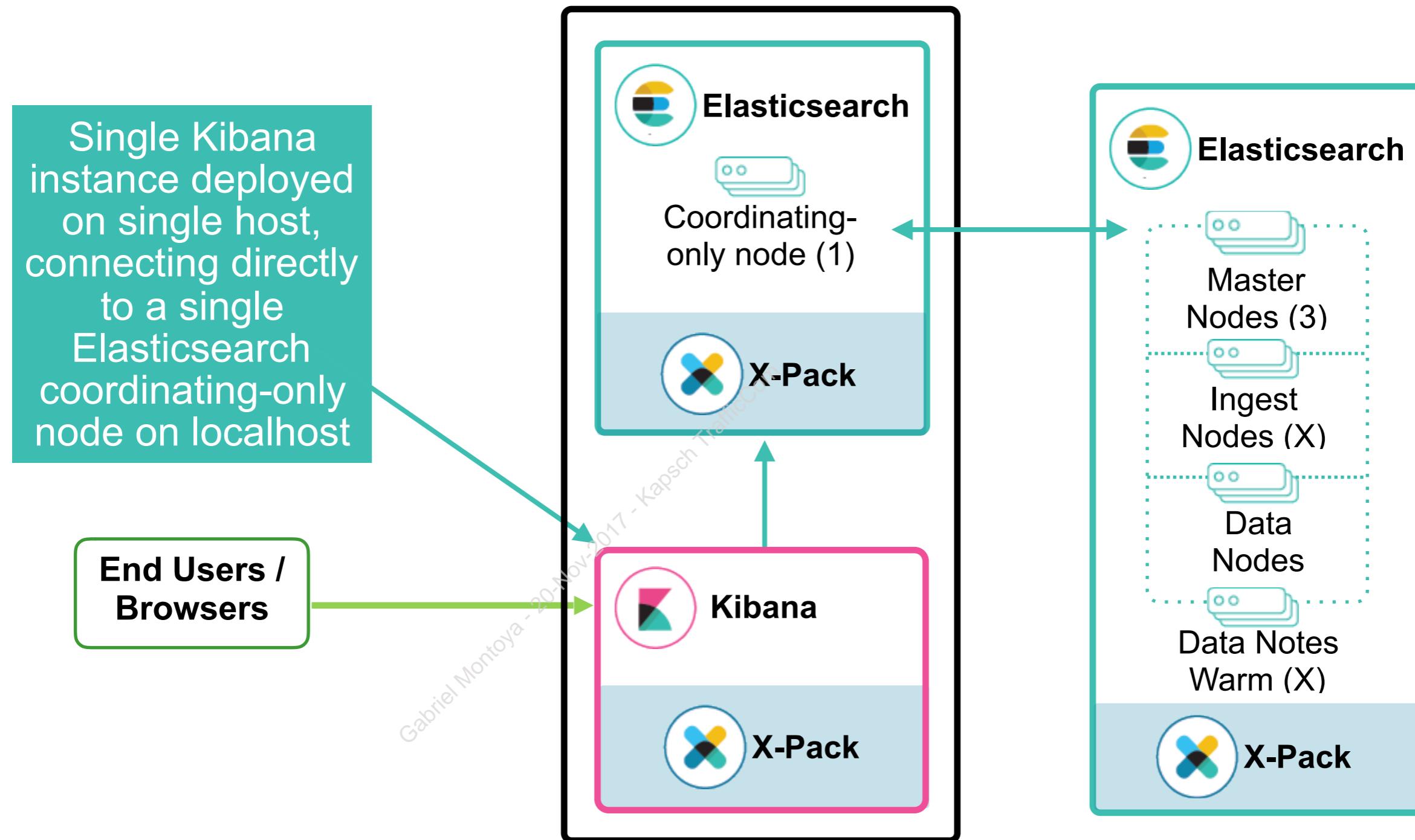
Scaling Kibana

- Kibana instances are stateless (any state information is saved in Elasticsearch indices)
- If a Kibana goes down, it is trivial to spin up another one to take over
- Multiple Kibana instances add fault-tolerance
- We will see how to use Kibana in conjunction with load balancing later in this chapter
- Kibana is often paired with an Elasticsearch coordinating-only node on the same host

Gabriel Montoya - 20-Nov-2017 Elasticsearch Training

Kibana + Coordinating-only node

- Kibana receives HTTP requests from end users via browser and communicates locally with Elasticsearch



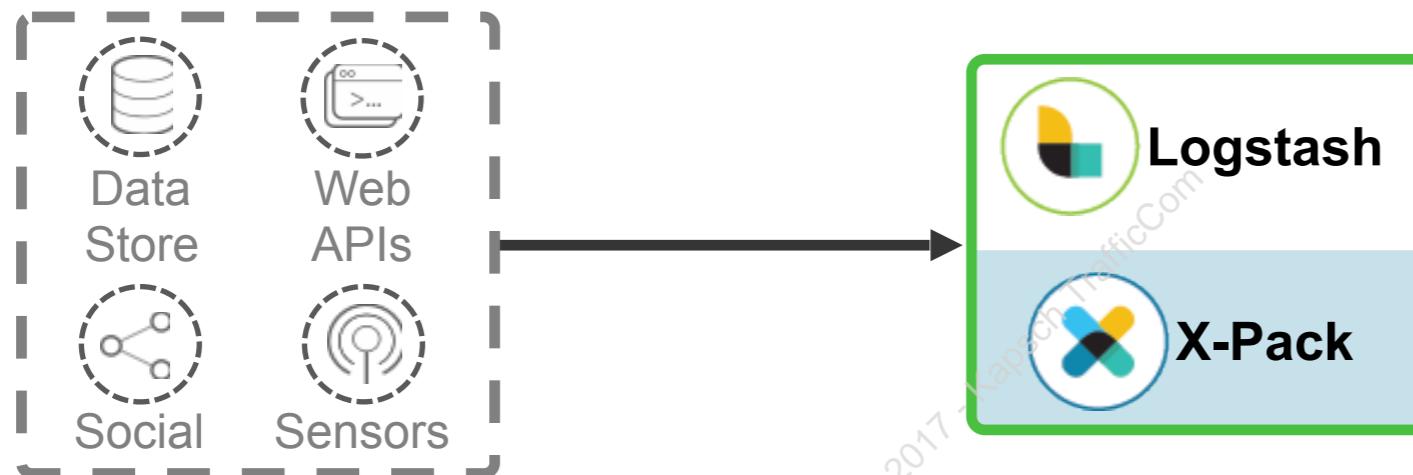
Scaling Logstash

- Logstash is currently not clustered (like Elasticsearch)
- This means you are required to manually add it in to your infrastructure, via configurations
- Loss of a Logstash may mean events are dropped, if clients are configured to send to the failed Logstash
 - We will see how load balancers and queues can help!

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

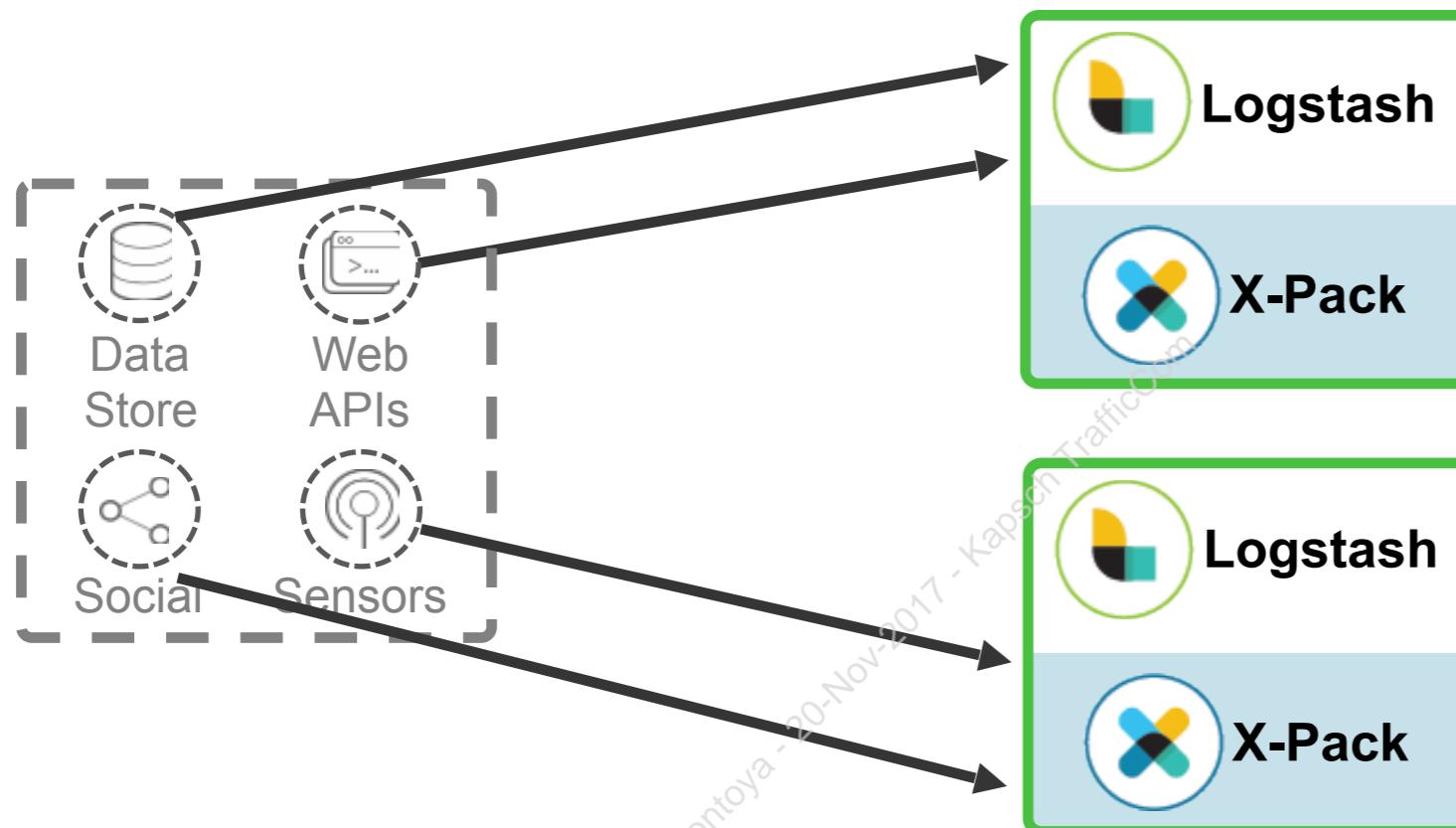
Scaling Logstash

- Before adding capacity, all events might be sent to a single Logstash...



Scaling Logstash

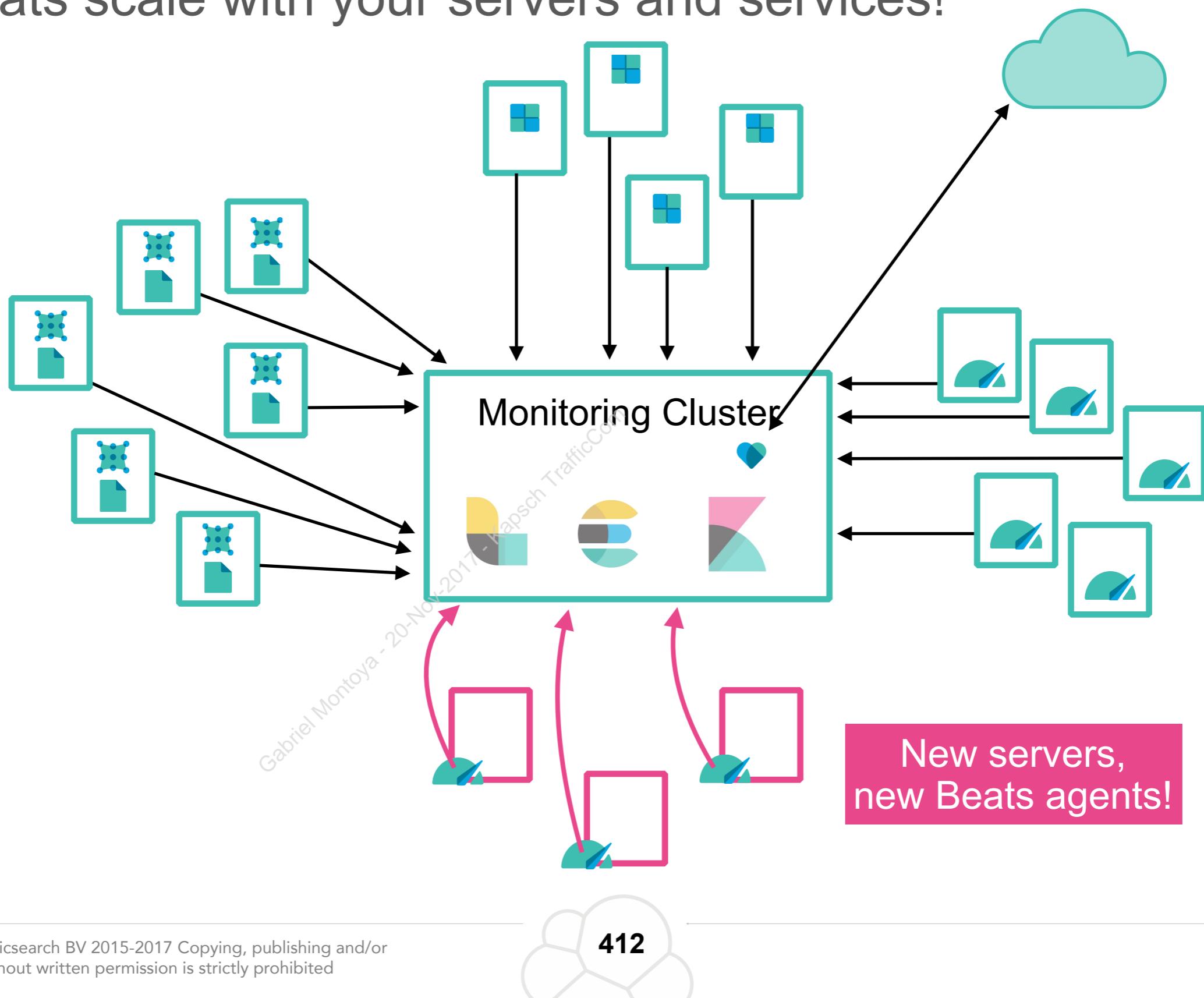
- After adding a Logstash instance, the hosts must be configured to also send to the new Logstash



We will soon see how loadbalancing can help!

Scaling Beats

- Beats scale with your servers and services!



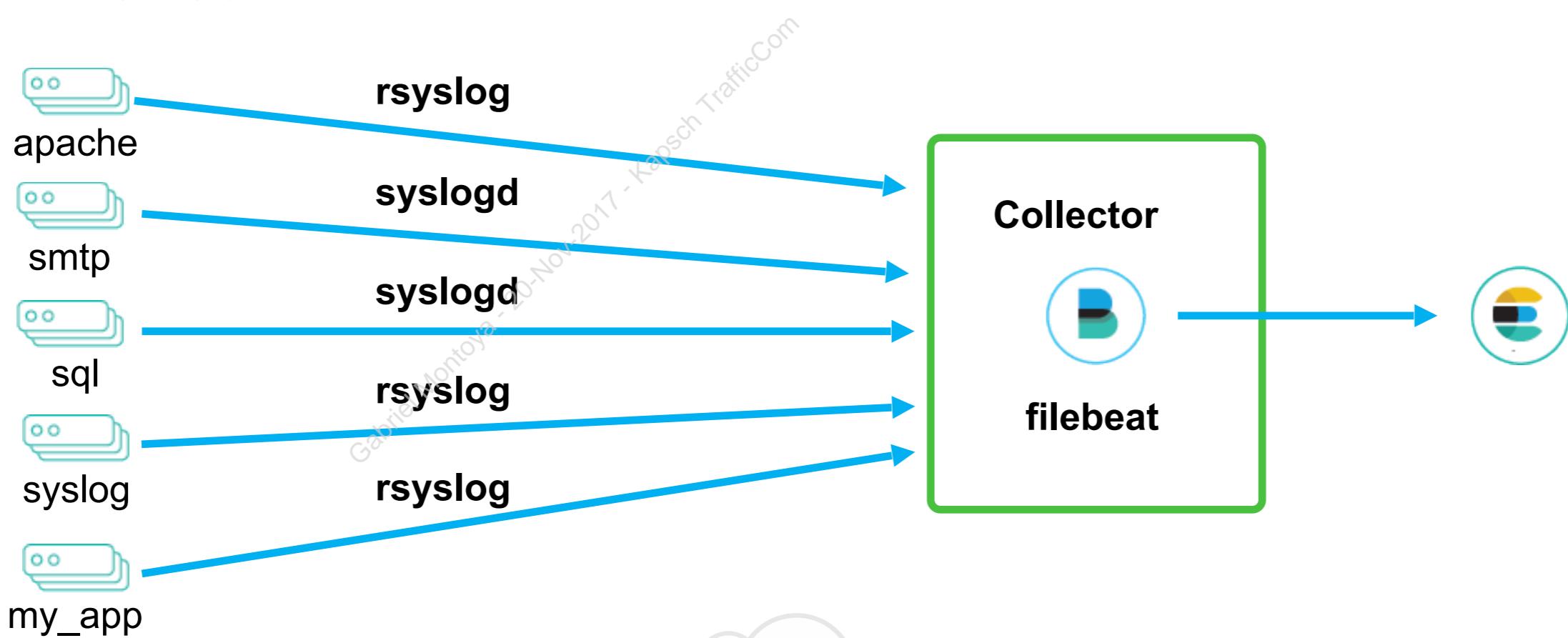
Scaling Beats

- Beats are designed to be easy...
- But they can become too many agents (100s or 1000s)
- The #1 recommendation when deploying Beats?
 - **Use Configuration Management!**
 - Ansible, puppet, chef (use what is already in use in your org)
 - Remember that Beats are written in Go. The binaries are compiled and therefore can be easily deployed to the edge.

Gabriel Montoya - 20-Nov-2017 - KapselTraffic.com

Consuming from a centralized log store

- Beats are not always deployed on all the client hosts
- Often, a tool such as rsyslog / syslogd is used
- The events are sent first to a centralized *collector* which accumulates all the events sent from the edge
- Beats can be used to consume from the collector and send downstream



Load Balancing

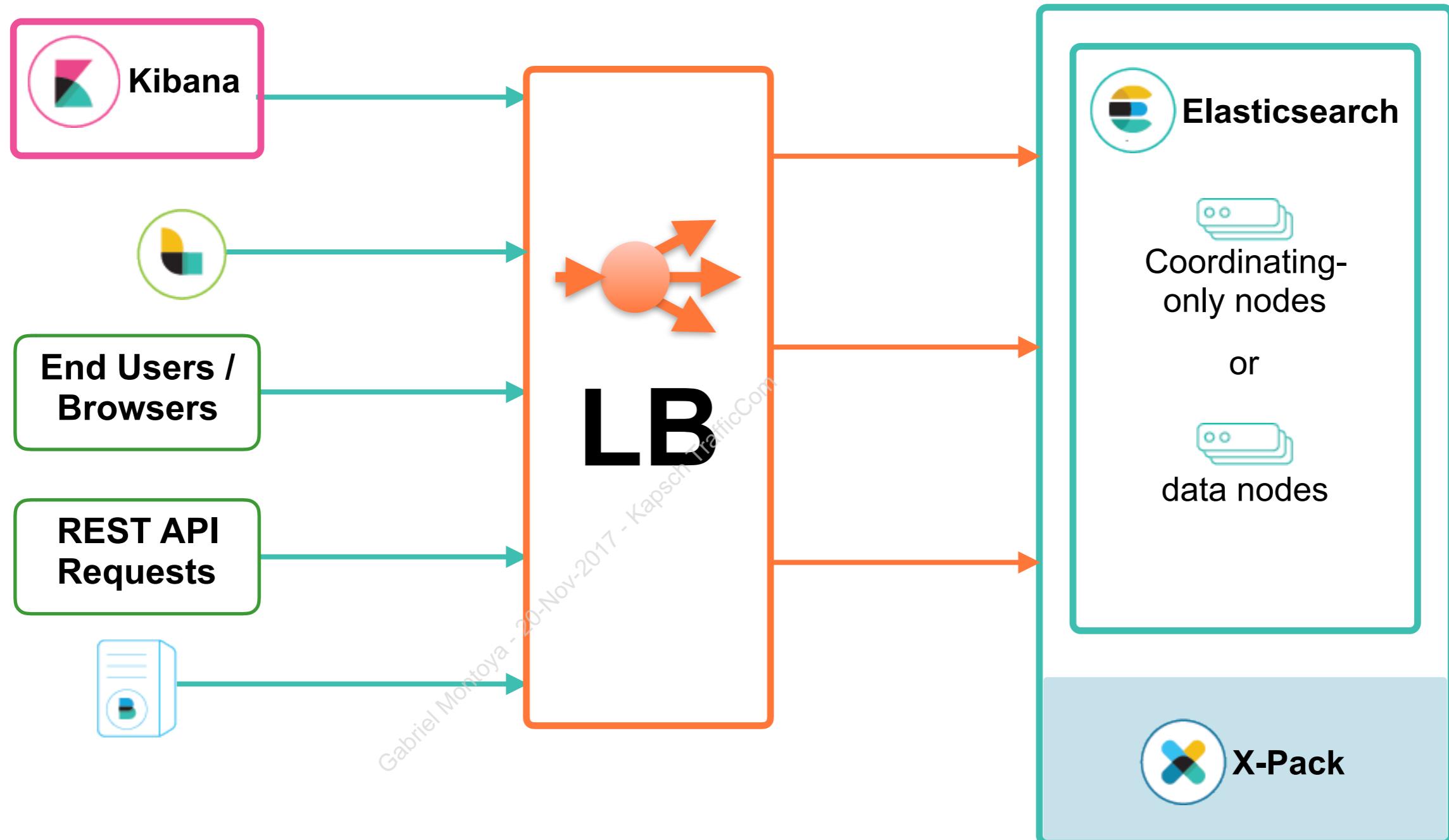
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Loadbalancing Elasticsearch

- Elasticsearch client libraries provide "sniffing"
- Alternatively, use an HTTP loadbalancer to send requests to all data nodes
- Dedicated master nodes must be omitted from the pool!
- Elasticsearch itself behaves as a round-robin loadbalancer
 - Any request sent to a node in the Elasticsearch cluster will be routed to the correct node(s) for handling the request
- Coordinating-only nodes may also be useful here
 - Loadbalancer can send all requests to the cluster through Coordinating-only nodes

Loadbalancing Elasticsearch

- Both reads and writes may be load balanced



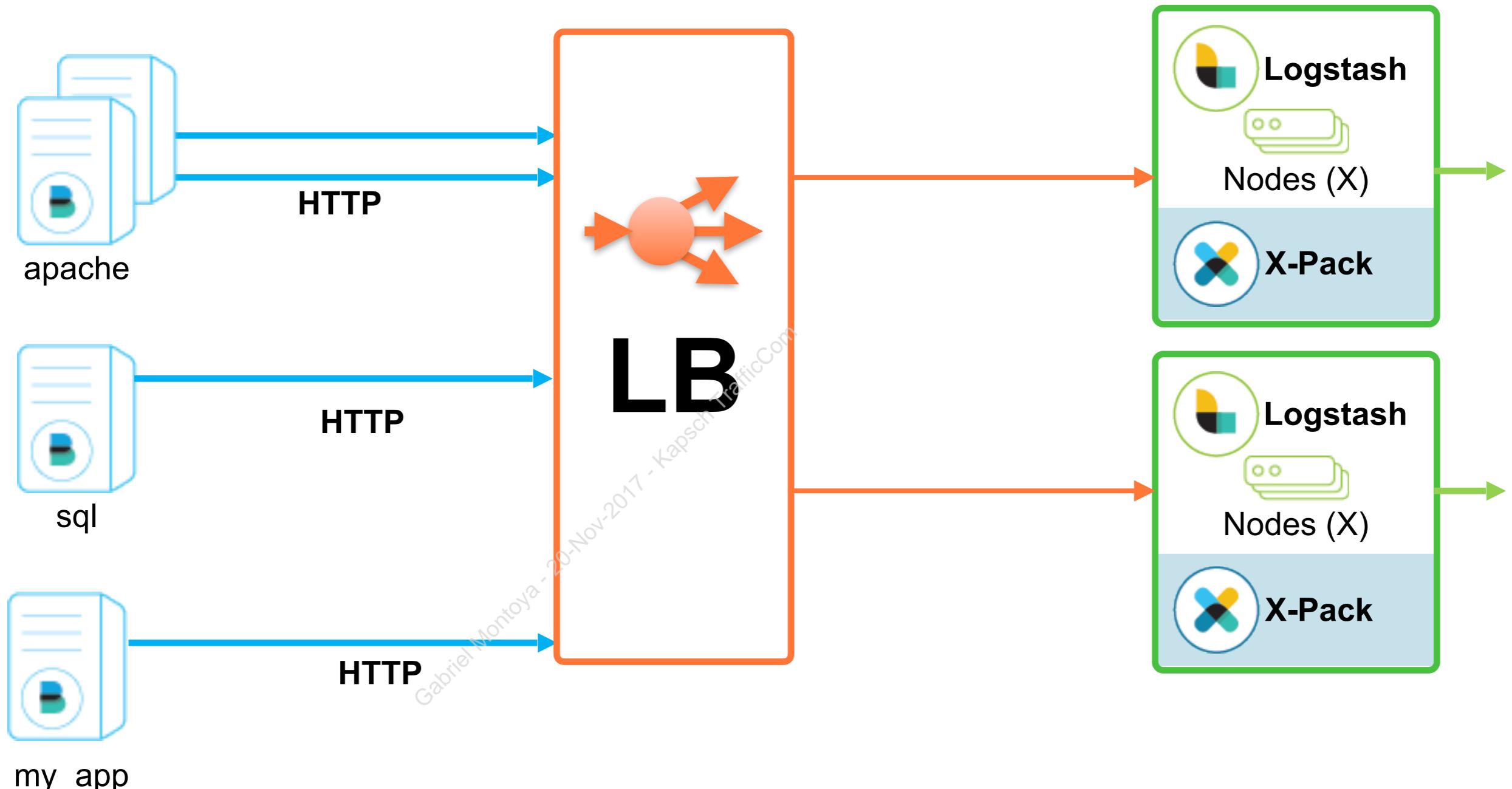
Loadbalancing Logstash

- Useful with multiple Logstash instances
- Can use "sticky" connections
- Will handle failure of Logstash instances
- Easy to scale up or down by adding Logstashes

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

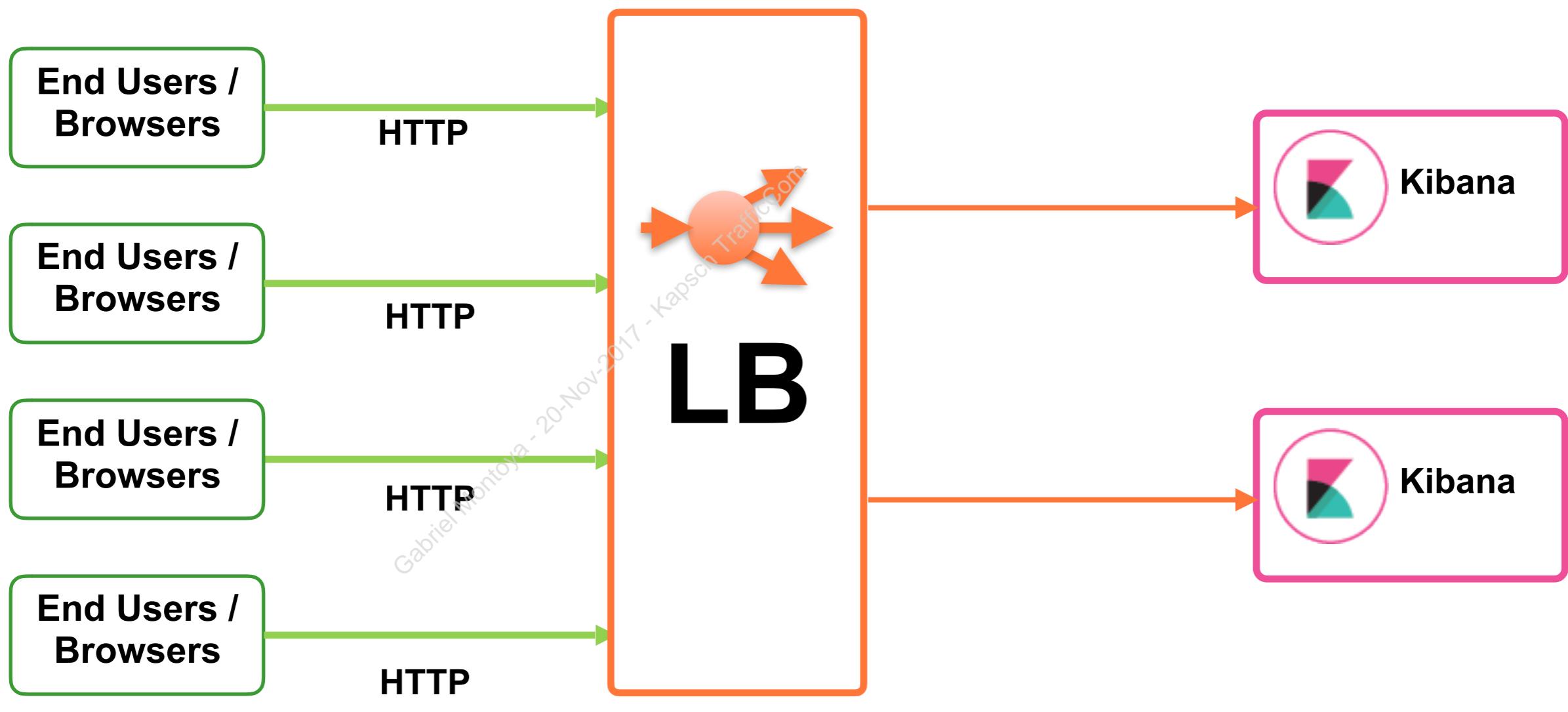
Loadbalancing Logstash

- Adding Logstash instances is easy with loadbalancing; simply add to LB pool



Loadbalancing Kibana

- Use Kibana + Coordinating-only node pattern which we reviewed earlier
- Use loadbalancer in front of Kibana instances
 - Provides fault-tolerance



Integrating Distributed Queues

Gabriel Montoya - 20-Nov-2017 - KapschTrafficCom

Integrating Distributed Queues

- Useful in ingestion architecture pipelines
- Can absorb spikes in event volume
- Allows for maintenance of Elasticsearch clusters (e.g. version upgrades)
- Prevents data loss in case of an outage
- Allows Logstash to perform additional enrichment before indexing into Elasticsearch
- Popular message brokers: Kafka, RabbitMQ, Redis
- Kafka is distributed and very fault-tolerant, but comes with added complexity (zookeeper)
- Logstash Persistent Queues perform a similar task, but are not distributed.

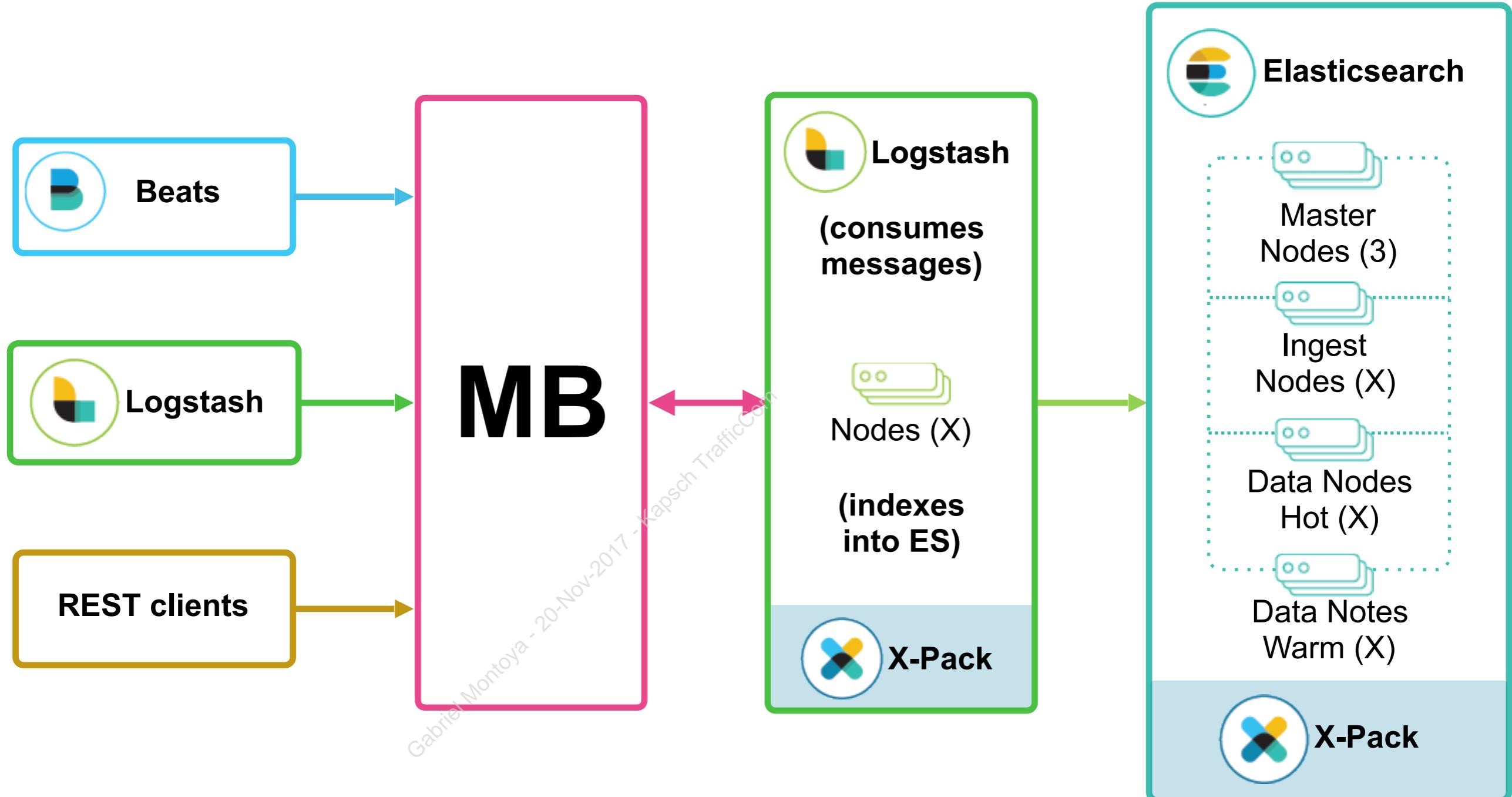


Integrating Distributed Queues

- Basic workflow:
 - Write all messages to message broker
 - These "Producers" can be:
 - Logstash
 - Beats family
 - Elasticsearch client libraries
 - REST API writes
 - etc etc etc
 - Message queue receives messages, buffers
 - Logstash "Consumes" from queue, writes to Elasticsearch

Gatito Montoya - 20-Nov-2017 - Kapsch TrafficCom

Distributed Queues / Message Brokers



Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- Do not overshard!
- Ensure the **sizing** is appropriate at the host level
- Elasticsearch, Logstash and Kibana are **scalable**, using different methods
- Beats deployments benefit from configuration management
- Elasticsearch, Logstash and Kibana can all benefit from the introduction of **loadbalancing**
- Deploying distributed queues in your infrastructure can provide high levels of **durability** and **availability**
- There are many architectural options to deploy the Elastic Stack.
- Start simple and evolve with your business data, the most reliable architecture might not be the best for you.



Quiz

1. How can one avoid oversharding?
2. Name 2 different Elastic Stack architectural options and explain their pros and cons.
3. **True / False:** It is preferred to deploy homogenous hosts in the Elastic Stack, with identical RAM, CPU, and Disk performance.
4. **True / False:** Logstash nodes will cluster with one another when they find other Logstash on the same network.
5. **True / False:** Beats require some form of messaging queue in order to collect events before indexing into Elasticsearch.
6. Name three use cases for deploying a persistent queue.

Gabriel Montoya - Veloy-2017@oschandra.com



Lab 9

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 10

Triage and Maintenance

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

- 1 Elastic Stack Data Administration Concepts
- 2 System Metrics
- 3 Service Metrics
- 4 Ingesting File Data
- 5 Data Processing
- 6 Data Enrichment
- 7 Data Store Integration
- 8 Network Monitoring
- 9 Data Ingestion Architectures
- 10 Triage and Maintenance

Topics covered:

- Monitoring Data Ingestion
- Beats Tuning
- Logstash Tuning
- Upgrading Logstash & Beats

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Monitoring Data Ingestion

Gabriel Montoya - 20-Nov-2017 - Kapsch Training Com

Logstash Monitoring

- Logstash automatically captures runtime metrics
- The metrics collected include:
 - Logstash node info (pipeline settings, OS info, JVM info, etc.)
 - Plugin info, including a list of installed plugins
 - Node stats (JVM, process, event-related, and pipeline runtime)
 - Hot threads.
- Use the **Monitoring APIs** to retrieve these metrics
- Use the **Monitoring UI** to view these metrics (X-Pack Basic)
- Use **Pipeline viewer** to better understand a pipeline (X-Pack Basic)

David Montoya - 20-Nov-2017 - KastchTrafficCom

Logstash Monitoring APIs

Gabriel Montoya - 20-Nov-2017 - KapsiTrain.com



Monitoring APIs

- Node Info API
- Plugins Info API
- Node Stats API
- Hot Threads API

default port, use --http.port
to define another one

```
curl -XGET 'localhost:9600/?pretty'
```

JSON returned will
be pretty formatted

```
{  
  "host": "skywalker",  
  "version": "6.0.0-beta2",  
  "http_address": "127.0.0.1:9600"  
}
```

Node Info API

- Returns static information about the node.

```
curl -XGET 'localhost:9600/_node/<types>'
```

- Types (optional):
 - pipelines
 - os
 - jvm
- Set the info that's returned by using a comma-separated list

Gabriel Montoya - 20-Nov-2017 - Kibana TrafficCom

Node Info API

```
curl -XGET 'localhost:9600/_node/os,jvm?pretty'
```

OS and JVM info

```
{  
  "os": {  
    "name": "Mac OS X",  
    "arch": "x86_64",  
    "version": "10.12.4",  
    "available_processors": 8  
  },  
  "jvm": {  
    "pid": 59616,  
    "version": "1.8.0_65",  
    "vm_name": "Java HotSpot(TM) 64-Bit Server VM",  
    "vm_version": "1.8.0_65",  
    "vm_vendor": "Oracle Corporation",  
    "start_time_in_millis": 1484251185878,  
    "mem": {  
      "heap_init_in_bytes": 268435456,  
      "heap_max_in_bytes": 1037959168,  
      "non_heap_init_in_bytes": 2555904,  
      "non_heap_max_in_bytes": 0  
    },  
    "gc_collectors": [ ... ] } }
```

Gabriel Montoya - 20-Nov-2017 - elasticsearchfordata.science

Node Info API

```
curl -XGET 'localhost:9600/_node/pipelines?pretty'
```

Information on
all pipelines

```
{  
  "pipelines" : {  
    "test" : {  
      "workers" : 1,  
      "batch_size" : 1,  
      "batch_delay" : 5,  
      "config_reload_automatic" : false,  
      "config_reload_interval" : 3  
    },  
    "test2" : {  
      "workers" : 8,  
      "batch_size" : 125,  
      "batch_delay" : 5,  
      "config_reload_automatic" : false,  
      "config_reload_interval" : 3  
    }  
  }  
}
```

Optionally, one can see the info for a specific pipeline by including the pipeline ID

Plugin Info API

- Returns information about all Logstash plugins that are currently installed.

```
curl -XGET 'localhost:9600/_node/plugins?pretty'
```

```
{  
  "total": 93,  
  "plugins": [  
    {  
      "name": "logstash-codec-cef",  
      "version": "4.1.2"  
    },  
    {  
      "name": "logstash-codec-collectd",  
      "version": "3.0.3"  
    },  
    {  
      "name": "logstash-codec-dots",  
      "version": "3.0.2"  
    },  
    ...  
  ]  
}
```

Same output as:
`bin/logstash-plugin list --verbose`

Node Stats API

- Returns runtime stats about the node.

```
curl -XGET 'localhost:9600/_node/stats/<types>'
```

- Types (optional):
 - jvm
 - process
 - events
 - pipelines
 - reloads
 - os
- If no type is provided, all stats are returned.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Node Stats API

```
curl -XGET 'localhost:9600/_node/stats/events,reload?pretty'
```

```
{  
  "events" : { ← event-related statistics for the Logstash instance  
    "in" : 293658,  
    "filtered" : 293658,  
    "out" : 293658,  
    "duration_in_millis" : 2324391,  
    "queue_push_duration_in_millis" : 343816  
  },  
  " reloads" : { ← config reload successes and failures  
    "successes" : 0,  
    "failures" : 0  
  }  
}
```

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Node Stats API

```
curl -XGET 'localhost:9600/_node/stats/pipelines?pretty'
```

```
{  
  "pipelines" : {  
    "test" : {  
      "events" : {  
        "duration_in_millis" : 365495,  
        "in" : 216485,  
        "filtered" : 216485,  
        "out" : 216485,  
        "queue_push_duration_in_millis" : 342466  
      },  
      "plugins" : {  
        "inputs" : [ { "name" : "beats", ... }. {...} ],  
        "filters" : [ { "name" : "grok", ... }, {...} ],  
        "outputs" : [ { "name" : "elasticsearch", ... } ]  
      },  
      " reloads" : { ... },  
      "queue" : {  
        "type" : "memory"  
      }  
    },  
    "test2" : { ... }  
  }  
}
```

Information on all pipelines

the number of events that were input, filtered, or output by each pipeline

stats for each configured plugin

config reload successes and failures (when enabled)

info about the persistent queue (when enabled)

Gabriel Mottola - 20 Nov 2017 - KapstoneTrafficCom

Hot Threads API

```
curl -XGET 'localhost:9600/_node/hot_threads?pretty'
```

```
{  
  "hot_threads" : {  
    "time" : "2017-06-06T18:25:28-07:00",  
    "busiest_threads" : 3,  
    "threads" : [  
      {  
        "name" : "Ruby-0-Thread-7",  
        "percent_of_cpu_time" : 0.0,  
        "state" : "timed_waiting",  
        "path" : "/path/to/logstash-6.0.0-beta2/vendor/  
bundle/jruby/1.9/gems/puma-2.16.0-java/lib/puma/  
thread_pool.rb:187",  
        "traces" : [ "java.lang.Object.wait(Native Method)",  
"org.jruby.RubyThread.sleep(RubyThread.java:1002)",  
"org.jruby.RubyKernel.sleep(RubyKernel.java:803) " ]  
      },  
      ...  
    ]  
  }  
}
```

gets the current hot threads for Logstash

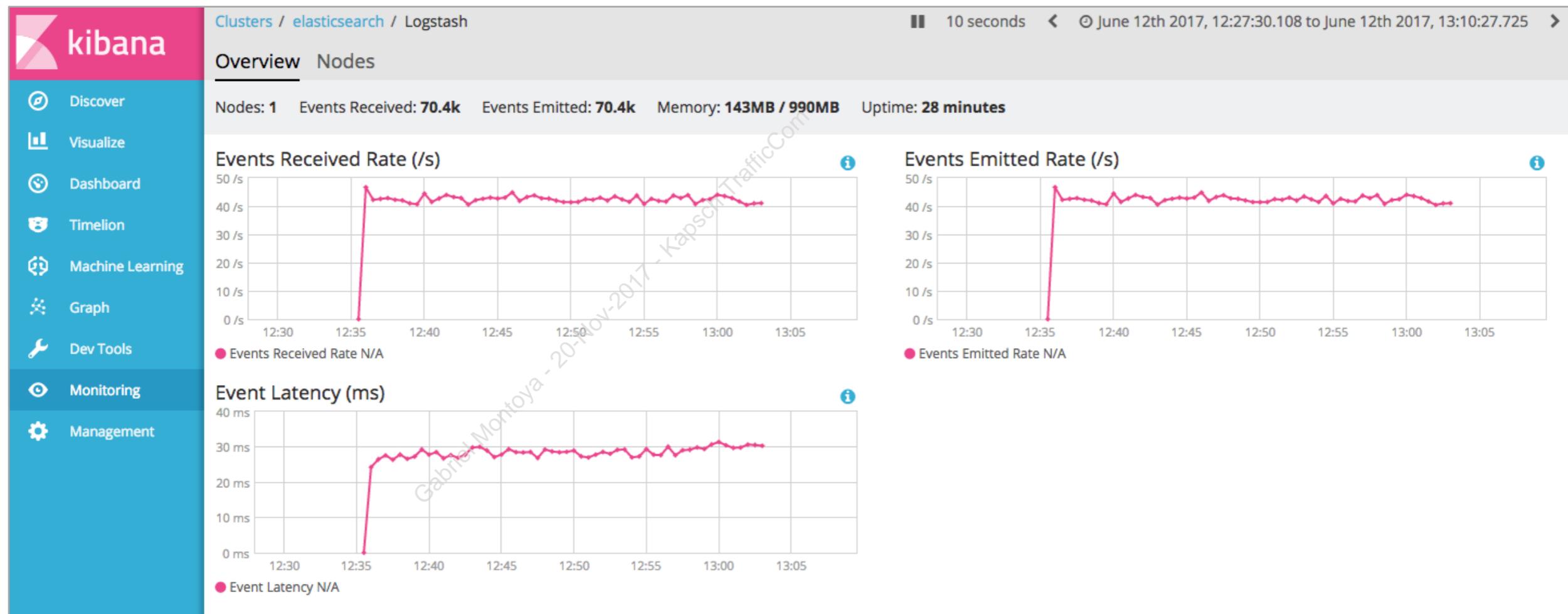


Logstash Monitoring UI

Gabriel Montoya - 20-Nov-2017 - KapsuleTrainCom

Monitoring UI

- X-Pack feature under the Basic License (**free**).
- Deep visibility into your Logstash deployment.
- Subset of the Monitoring API data as Visualizations
- Logstash Overview:



Monitoring UI

- Logstash Node Stats:



Monitoring UI

- Installation:

```
bin/logstash-plugin install x-pack
```

- Configuration (logstash.yml):

```
xpack.monitoring.elasticsearch.url: ["http://es-prod-node-1:9200"]  
xpack.monitoring.elasticsearch.username: "logstash_system"  
xpack.monitoring.elasticsearch.password: "changeme"
```

Logstash always points
to the production cluster

the logstash_system password
defined with setup-passwords

Logstash Monitoring

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

The screenshot shows the Kibana interface with the following sections:

- Clusters:** elasticsearch
- Top Cluster Alerts:** Elasticsearch cluster status is yellow. **Allocate missing replica shards.** Last checked August 9, 2017 5:05:13 PM (since 2 hrs 36 min ago)
- Elasticsearch Overview:** Version: 6.0.0-beta1, Uptime: 3 hours, Jobs: 0. Metrics: Nodes: 1, Disk Available: 5GB / 8GB (68.68%), JVM Heap: 66.36% (669MB / 1007MB). Indices: 13, Documents: 35,090, Primary Shards: 13, Replica Shards: 0. Health: Yellow.
- Kibana Overview:** Requests: 1276, Max. Response Time: 3380 ms. Metrics: Instances: 1, Connections: 2,827, Memory Usage: 10.58% (152MB / 1GB). Health: Green.
- Logstash Overview:** Events Received: 6.1k, Events Emitted: 6k. Metrics: Nodes: 1, Uptime: 2 hours, JVM Heap: 18.47% (186MB / 1007MB). Pipelines: 1. A teal bar at the bottom right says "Logstash is now here!" with a green arrow pointing from the Logstash section towards it.

Logstash Pipeline Viewer

Gabriel Montoya - 20-Nov-2017 - KapschTraining.com



Logstash Pipeline Viewer

- X-Pack feature under the Basic license (**free**).
- Deep visibility into your pipelines.
- Directed acyclic graph (DAG) representation of the overall pipeline topology, data flow, and branching logic
- Important metrics, like events per second and time spent in milliseconds, for each plugin in the view.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logstash Pipeline Viewer

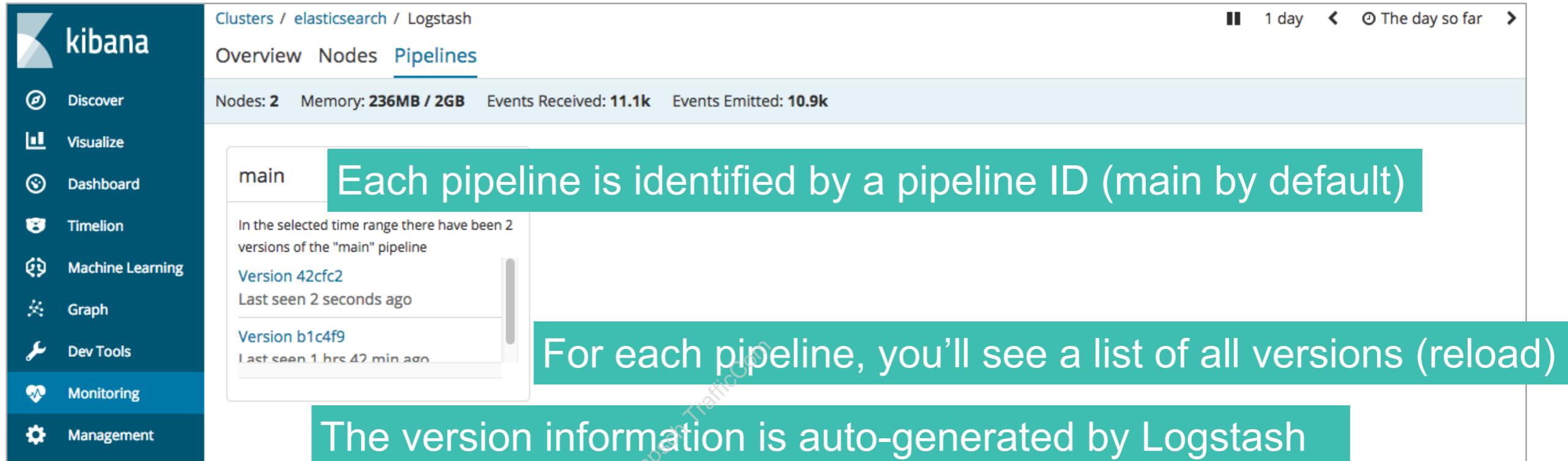
Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

The screenshot shows the Kibana interface with the following sections:

- Clusters**:
 - elasticsearch**: Status: Yellow (Allocate missing replica shards). Last checked August 9, 2017 5:05:13 PM (since 2 hrs 36 min ago).
 - Elasticsearch**: Overview (Version: 6.0.0-beta1, Uptime: 3 hours, Jobs: 0), Nodes: 1 (Disk Available: 5GB / 8GB (68.68%), JVM Heap: 66.36% (669MB / 1007MB)), Indices: 13 (Documents: 35,090, Disk Usage: 10MB, Primary Shards: 13, Replica Shards: 0). Health: Yellow.
 - Kibana**: Overview (Requests: 1276, Max. Response Time: 3380 ms). Health: Green.
 - Logstash**: Overview (Events Received: 6.1k, Events Emitted: 6k). Pipelines: 1.

A teal callout box with the text "Check your pipelines!" has an arrow pointing down towards the Logstash Pipelines section.

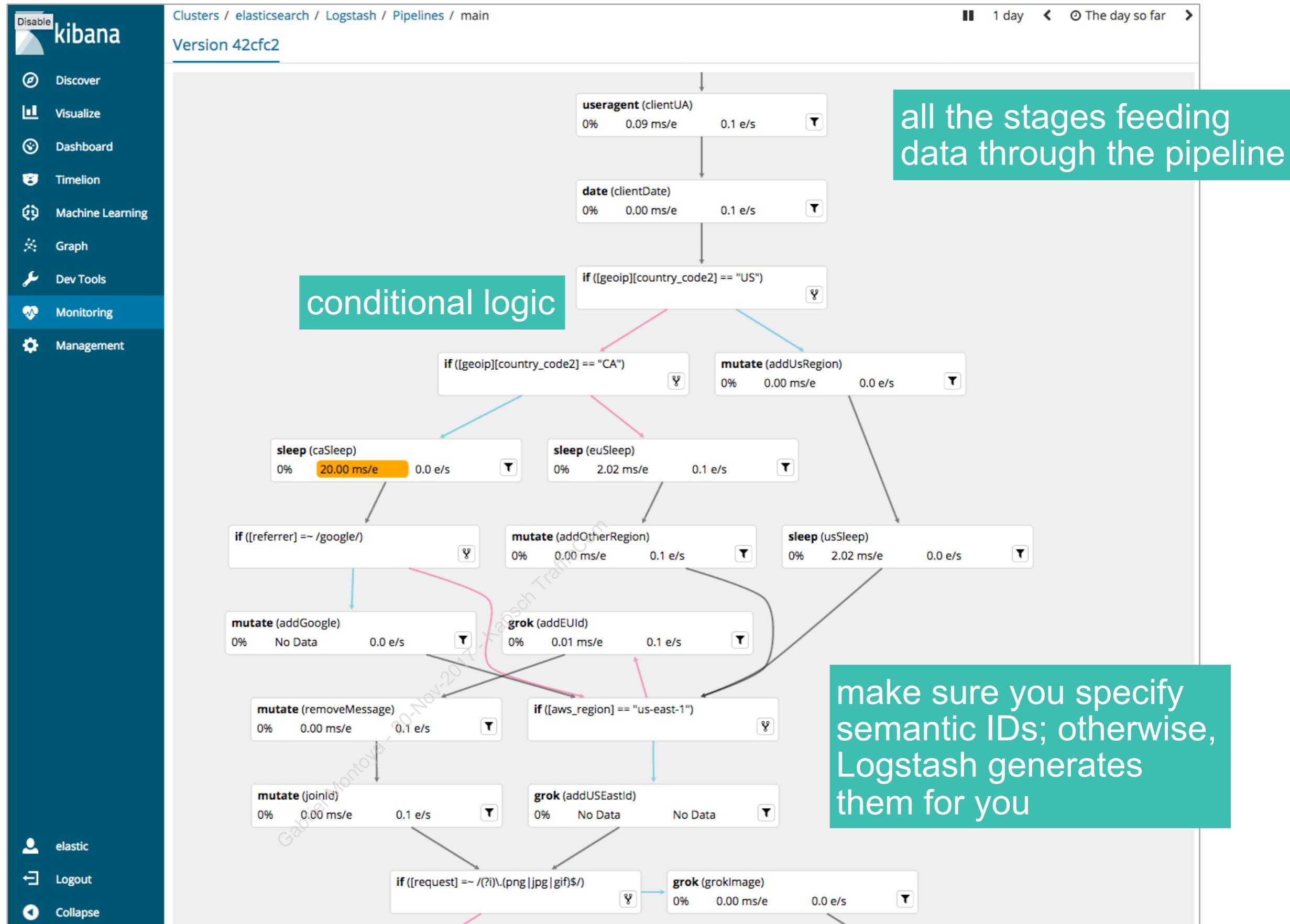
Logstash Pipeline Viewer



The screenshot shows the Kibana interface with the 'Clusters / elasticsearch / Logstash' path selected. The 'Pipelines' tab is active. The top bar shows 'Nodes: 2 Memory: 236MB / 2GB Events Received: 11.1k Events Emitted: 10.9k'. A large teal box highlights the 'main' pipeline section, which displays the message: 'Each pipeline is identified by a pipeline ID (main by default)'. Below this, it says 'In the selected time range there have been 2 versions of the "main" pipeline' and lists two versions: 'Version 42fcf2 Last seen 2 seconds ago' and 'Version b1c4f9 Last seen 1 hrs 42 min ago'. Another teal box highlights the message: 'For each pipeline, you'll see a list of all versions (reload)'. A third teal box highlights the message: 'The version information is auto-generated by Logstash'.

Each time you modify a pipeline, Logstash generates a new version hash.

Note that Logstash stores the pipeline stats; it does not store the pipeline configurations themselves.



Beats Tuning

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats Tuning

- Input
 - depends on the Beats
- Output
 - Elasticsearch
 - Logstash

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Beats Tuning

- Settings you can try to optimize to your use case.

	Elasticsearch	Logstash
bulk_max_size	50	2048
compression_level	0	3
loadbalance	FALSE	TRUE
worker	1	1
flush_interval	1s	X
pipelining	X	0

default values

Gabriel Montoya
20-Nov-2017 - Kapsch TrafficCom

Logstash Tuning

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Performance Troubleshooting Guide

- Check input and output source performance.
 - Logstash is only as fast as the services it connects to.
- Check system statistics
 - CPU
 - memory
 - I/O
- Tune Logstash worker settings

Gabriel Montoya - 20-Nov-2017 - KappaTrafficCom

CPU

- Is CPU being heavily used?
- top -Hto shows detailed process statistics (Linux/Unix)
- Is the heap size too low? (constant garbage collection)
 - try to double the heap size and see if performance improves, but within Memory recommendations
- Scale up the number of pipeline workers (-w flag)
 - scale this up to a multiple of CPU cores, if need be, as the threads can become idle on I/O.

Gabriel Montoya - 20-Nov-2017 XapsusTrafficCom

Logstash Heap

- JVM
 - Set min (Xms) and max (Xmx) heap size to the same value
- Size
 - It depends on your pipeline
 - Never set more than the amount of physical memory
 - Leave at least 1GB free for the OS and other processes
- Is Logstash swapping?
 - other applications that use large amounts of memory

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Logstash I/O

- Disk saturation
 - plugins like file output may saturate your storage
 - too many errors that force Logstash to generate large error logs
 - use iostat, dstat, or something similar to monitor disk I/O (Unix)
 - persistent queues are I/O intensive
- Network saturation
 - plugins that perform a lot of network operations
 - use a tool like dstat or iftop to monitor your network (Unix)

Gabriel Montoya - 20-Nov-2017 - KapehTrafficCom

Tune Logstash worker settings

- Scale up the number of pipeline workers (-w flag)
 - increases the number of threads available for filters and outputs.
- By default, a single pipeline worker thread per output
 - increase the workers setting per output
 - Never make it larger than the number of pipeline workers
- Tune the output batch size
 - try different values and check which one has better results

Gabriel Montoya - 20-Nov-2017 - Kibana TrafficCom

Upgrading Logstash & Beats

Gabriel Montoya - 20-Nov-2017 - KapshticTraining.Com



Upgrading the Elastic Stack

1. Check your current version and choose the new version.
 - we will cover upgrades from 5.x to 6.0
 - upgrade steps might change between versions, check the documentation for the correct version.
2. Read the breaking changes and see how it is going to impact your current deployment!
3. Fully upgrade Elasticsearch and Kibana to version 6.0 before upgrading Logstash or Beats.

Gabriel Montoya - 20-Nov-2017 - KappaTraffic.com

Upgrading Beats

1. Read the breaking changes!
2. Minor Version
 - A. install the new version
 - B. restart the Beat process
3. Major Version (5.x -> 6.0)
 - A. upgrade to 5.6 (use step 2)
 - B. manually load new template (for every beat type)

```
curl -XPUT -H'Content-Type: application/json' \
http://localhost:9200/_template/metricbeat -d @metricbeat.template.json
```
 - C. migrate configuration files there were a lot of changes
 - D. re-import Kibana dashboards
 - E. install the new version and restart the Beat process

Upgrading Logstash

1. Read the breaking changes!
2. Shut down your Logstash pipeline, including any inputs that send events to Logstash.
3. Download the Logstash installation file that matches your host environment.
4. Unpack the installation file into your Logstash directory.
5. Test your configuration file with:
`--config.test_and_exit -f <configuration-file>`
6. Restart your Logstash pipeline after updating your configuration file.

Chapter Review

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Summary

- Logstash automatically captures runtime metrics
- Monitoring UI and the Pipeline Viewer are free and should be used to monitor logstash performance.
- The pipeline viewer gives you a fine-grained view of each pipeline being executed.
- The number of pipeline workers (-w flag), controls the number of threads available for filters and outputs.
- There are a few settings you can change to try to optimize Beats output.
- Upgrade Elasticsearch and Kibana before upgrading Beats or Logstash.
- There are a lot of breaking changes from 5.x to 6.0 so make sure you read the documentation before doing an upgrade.



Quiz

1. What are the three monitoring options that Logstash offers?
2. **True or False:** The Monitoring UI and the Pipeline Viewer are part of the X-Pack and you need a payed license to use them.
3. In Beats, the _____ setting defines the maximum number of events in a single request to Elasticsearch or Logstash.
4. **True or False:** The JVM Heap settings -Xms -Xmx should have different values.
5. When upgrading the Elastic Stack, first upgrade _____ and _____, then upgrade _____ and _____.

Gabriel Montoya - 20-Nov-2017 Kapsch TrafficCom

Lab 10

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Conclusions

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Documentation and Help

- Discussion forums: <http://discuss.elastic.co/>
- Meetups: <http://elasticsearch.meetup.com>
- Docs: <https://elastic.co/docs>
- Community: <https://elastic.co/community>
- More resources: <https://elastic.co/learn>

Gabriel Montoya - 20-Nov-2017 - https://TrafficCom

Thank you!

Please complete the online survey

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Quiz Answers

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 1 Quiz Answers

1. Apache server logs that you want to analyze. E-commerce search.
2. Heartbeat - monitors if services are available. Filebeat - tails files. Metricbeat - collect metrics from OSs and Services.
3. False, it also supports HTTP.
4. True
5. mode: all
6. A visualization is a chart. A dashboard is a collection of visualizations together in the same page.
7. True

Gabriel Montalvo
26Nov-2017 - Kapsch TrafficCom

Chapter 2 Quiz Answers

1. ./metricbeat test config
2. False. Use setup command
3. True
4. Metricbeat
5. False. The system module is enabled by default, by having a file called system.yml in the module.d folder
6. False. Add a drop_event processor that tests the user name of the process
7. False. You have only one period per module for all the metricsets of that module

Gabriel Montoya - 20-Nov-2017 - Kapsel Traffic Control

Chapter 3 Quiz Answers

1. True.
2. True, but need to enable `reload.enable:true` (which is false by default)
3. True
4. Shards
5. True

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 4 Quiz Answers

1. Input, Filter, Output, Codec
2. True
3. False
4. Registry file keeps track of which lines were read. At-least-once guarantee
5. False! You should use it to debug.
6. Any line that does not match the specified pattern belongs to the previous line. Lines starting with [yyyy-mm-dd] are the start of a multiline event.

Gabriel Montoya - 2017-07-20 Kappa Traffic



Chapter 5 Quiz Answers

1. False. you can hardly answer questions with unstructured data
2. The field reference is wrong. %{ should be inside the string and not in the conditional.
3. True.
4. translate
5. drop

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 6 Quiz Answers

1. Easier to write regular expression by re-using patterns
2. False
3. True
4. True
5. Flexibility is a pro as you can execute anything, but performance wise it might have a negative impact
6. False, it could be running anywhere. Actually as it is a lookup and in general the data is not HUGE, it would be better to have an Elasticsearch node with a replica of the lookup data allocated to the same machine.



Chapter 7 Quiz Answers

1. False
2. True
3. False, it is a Filter
4. True
5. Sequential

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 8 Quiz Answers

1. False - you can use setcap to allow running as a normal user.
2. final
3. ./packetbeat devices
4. false
5. false - a flow is needed to gather some information about secure communications

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Chapter 9 Quiz Answers

1. In many cases, oversharding can be avoided by decreasing the default number of shards per index or increasing the index time window.
2. Beats->Elasticsearch, for metrics.
Beats->logstash->elasticsearch for metrics and logs.
3. False
4. False
5. False
6. Several choices:
 - absorb traffic spikes
 - perform upgrades of Elasticsearch
 - unplanned Elasticsearch outages
 - additional enrichment of event data



Chapter 10 Quiz Answers

1. Monitoring APIs and UI and Pipeline Viewer
2. False. It is free.
3. **bulk_max_size**
4. False. They should have the same value
5. Elasticsearch, Kibana, Logstash, Beats

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom

Course: Elastic Stack Data Administration

Version 6.0.1

© 2015-2017 Elasticsearch BV. All rights reserved. Decompiling, copying, publishing and/or distribution without written consent of Elasticsearch BV is strictly prohibited.

Gabriel Montoya - 20-Nov-2017 - Kapsch TrafficCom