

**Centralized**

# **Logging Patterns**

Philipp Krenn

@xeraa



@xeraa



[philipp@~/Documents/GitHub/java-logging(git\*master)\*> gradle run 14:47:14]

> Task :run

[2018-05-31 14:47:22.185] TRACE net.xeraa.logging.LogMe [main] - session=29, loop=1 - Iteration '1' and session '29'

[2018-05-31 14:47:22.196] DEBUG net.xeraa.logging.LogMe [main] - session=29, loop=1 - Collect in development

[2018-05-31 14:47:22.200] TRACE net.xeraa.logging.LogMe [main] - session=49, loop=2 - Iteration '2' and session '49'

[2018-05-31 14:47:22.201] DEBUG net.xeraa.logging.LogMe [main] - session=49, loop=2 - Collect in development

[2018-05-31 14:47:22.202] TRACE net.xeraa.logging.LogMe [main] - session=85, loop=3 - Iteration '3' and session '85'

[2018-05-31 14:47:22.203] INFO net.xeraa.logging.LogMe [main] - session=85, loop=3 - Collect in production

[2018-05-31 14:47:22.204] TRACE net.xeraa.logging.LogMe [main] - session=55, loop=4 - Iteration '4' and session '55'

[2018-05-31 14:47:22.204] DEBUG net.xeraa.logging.LogMe [main] - session=55, loop=4 - Collect in development

[2018-05-31 14:47:22.205] TRACE net.xeraa.logging.LogMe [main] - session=83, loop=5 - Iteration '5' and session '83'

[2018-05-31 14:47:22.205] WARN net.xeraa.logging.LogMe [main] - session=83, loop=5 - Investigate tomorrow

[2018-05-31 14:47:22.206] TRACE net.xeraa.logging.LogMe [main] - session=36, loop=6 - Iteration '6' and session '36'

[2018-05-31 14:47:22.206] INFO net.xeraa.logging.LogMe [main] - session=36, loop=6 - Collect in producti

```
java-logging — fish /Users/philipp/Documents/GitHub/java-logging — -fish — 105x26
[philipp@~/Documents/GitHub/java-logging(git*master)✓] cat logs/java-logging.log 14:47:23]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and session '46'
[2018-05-31 14:42:58.961] DEBUG net.xeraa.logging.LogMe [main] - session=46, loop=1 - Collect in development
[2018-05-31 14:42:58.963] TRACE net.xeraa.logging.LogMe [main] - session=13, loop=2 - Iteration '2' and session '13'
[2018-05-31 14:42:58.963] DEBUG net.xeraa.logging.LogMe [main] - session=13, loop=2 - Collect in development
[2018-05-31 14:42:58.964] TRACE net.xeraa.logging.LogMe [main] - session=70, loop=3 - Iteration '3' and session '70'
[2018-05-31 14:42:58.964] INFO net.xeraa.logging.LogMe [main] - session=70, loop=3 - Collect in production
[2018-05-31 14:42:58.965] TRACE net.xeraa.logging.LogMe [main] - session=68, loop=4 - Iteration '4' and session '68'
[2018-05-31 14:42:58.966] DEBUG net.xeraa.logging.LogMe [main] - session=68, loop=4 - Collect in development
[2018-05-31 14:42:58.966] TRACE net.xeraa.logging.LogMe [main] - session=84, loop=5 - Iteration '5' and session '84'
[2018-05-31 14:42:58.966] WARN net.xeraa.logging.LogMe [main] - session=84, loop=5 - Investigate tomorrow
[2018-05-31 14:42:58.967] TRACE net.xeraa.logging.LogMe [main] - session=82, loop=6 - Iteration '6' and session '82'
[2018-05-31 14:42:58.969] INFO net.xeraa.logging.LogMe [main] - session=82, loop=6 - Collect in production
[2018-05-31 14:42:58.969] TRACE net.xeraa.logging.LogMe [main] - session=7, loop=7 - Iteration '7' and se
```



java-logging — tail /Users/philipp/Documents/GitHub/java-logging — tail -f logs/java-loggi...

```
[philipp@~/Documents/GitHub/java-logging(git*master)✓] tail -f logs/java-logging.log 18:39:45]
[2018-05-31 17:20:22.874] TRACE net.xeraa.logging.LogMe [main] - session=61, loop=16 - Iteration '16' and
  session '61'
[2018-05-31 17:20:22.874] DEBUG net.xeraa.logging.LogMe [main] - session=61, loop=16 - Collect in develop
  ment
[2018-05-31 17:20:22.881] TRACE net.xeraa.logging.LogMe [main] - session=2, loop=17 - Iteration '17' and
  session '2'
[2018-05-31 17:20:22.882] DEBUG net.xeraa.logging.LogMe [main] - session=2, loop=17 - Collect in developm
  ent
[2018-05-31 17:20:22.883] TRACE net.xeraa.logging.LogMe [main] - session=35, loop=18 - Iteration '18' and
  session '35'
[2018-05-31 17:20:22.884] INFO net.xeraa.logging.LogMe [main] - session=35, loop=18 - Collect in product
  ion
[2018-05-31 17:20:22.886] TRACE net.xeraa.logging.LogMe [main] - session=86, loop=19 - Iteration '19' and
  session '86'
[2018-05-31 17:20:22.889] DEBUG net.xeraa.logging.LogMe [main] - session=86, loop=19 - Collect in develop
  ment
[2018-05-31 17:20:22.890] TRACE net.xeraa.logging.LogMe [main] - session=92, loop=20 - Iteration '20' and
  session '92'
[2018-05-31 17:20:22.891] WARN net.xeraa.logging.LogMe [main] - session=92, loop=20 - Investigate tomorr
  OW
[2018-05-31 18:40:05.399] TRACE net.xeraa.logging.LogMe [main] - session=40, loop=1 - Iteration '1' and s
  ession '40'
[2018-05-31 18:40:05.417] DEBUG net.xeraa.logging.LogMe [main] - session=40, loop=1 - Collect in developm
  ent
[2018-05-31 18:40:05.420] TRACE net.xeraa.logging.LogMe [main] - session=51, loop=2 - Iteration '2' and s
```



```
java-logging — fish /Users/philipp/Documents/GitHub/java-logging — -fish — 105x26
philipp@~/Documents/GitHub/java-logging(git*master) ✓> less +F logs/java-logging.log 18:42:08
```





```
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and session '46'
[2018-05-31 14:42:58.961] DEBUG net.xeraa.logging.LogMe [main] - session=46, loop=1 - Collect in development
[2018-05-31 14:42:58.963] TRACE net.xeraa.logging.LogMe [main] - session=13, loop=2 - Iteration '2' and s
```

```
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and session '46'
[2018-05-31 14:42:58.961] DEBUG net.xeraa.logging.LogMe [main] - session=46, loop=1 - Collect in development
[2018-05-31 14:42:58.963] TRACE net.xeraa.logging.LogMe [main] - session=13, loop=2 - Iteration '2' and s
```

```
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and session '46'
[2018-05-31 14:42:58.961] DEBUG net.xeraa.logging.LogMe [main] - session=46, loop=1 - Collect in development
[2018-05-31 14:42:58.963] TRACE net.xeraa.logging.LogMe [main] - session=13, loop=2 - Iteration '2' and session '13'
[2018-05-31 14:42:58.963] DEBUG net.xeraa.logging.LogMe [main] - session=13, loop=2 - Collect in development
[2018-05-31 14:42:58.964] TRACE net.xeraa.logging.LogMe [main] - session=70, loop=3 - Iteration '3' and session '70'
[2018-05-31 14:42:58.964] INFO net.xeraa.logging.LogMe [main] - session=70, loop=3 - Collect in producti
```

```
java-logging — fish /Users/philipp/Documents/GitHub/java-logging — -fish — 105x1
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
session '46'
[cat logs/java-logging.log]
[2018-05-31 14:42:58.951] TRACE net.xeraa.logging.LogMe [main] - session=46, loop=1 - Iteration '1' and s
[cat logs/java-logging.log]
```







A large, rusted metal structure, possibly a ship's hull, is shown against a dark background. A bright light source, likely a searchlight, is visible in the upper right, casting a beam of light. The structure is heavily corroded and has a complex, layered appearance.

me looking  
for the bug

**7.2 GB**  
**of log file**



ALL THE THINGS!



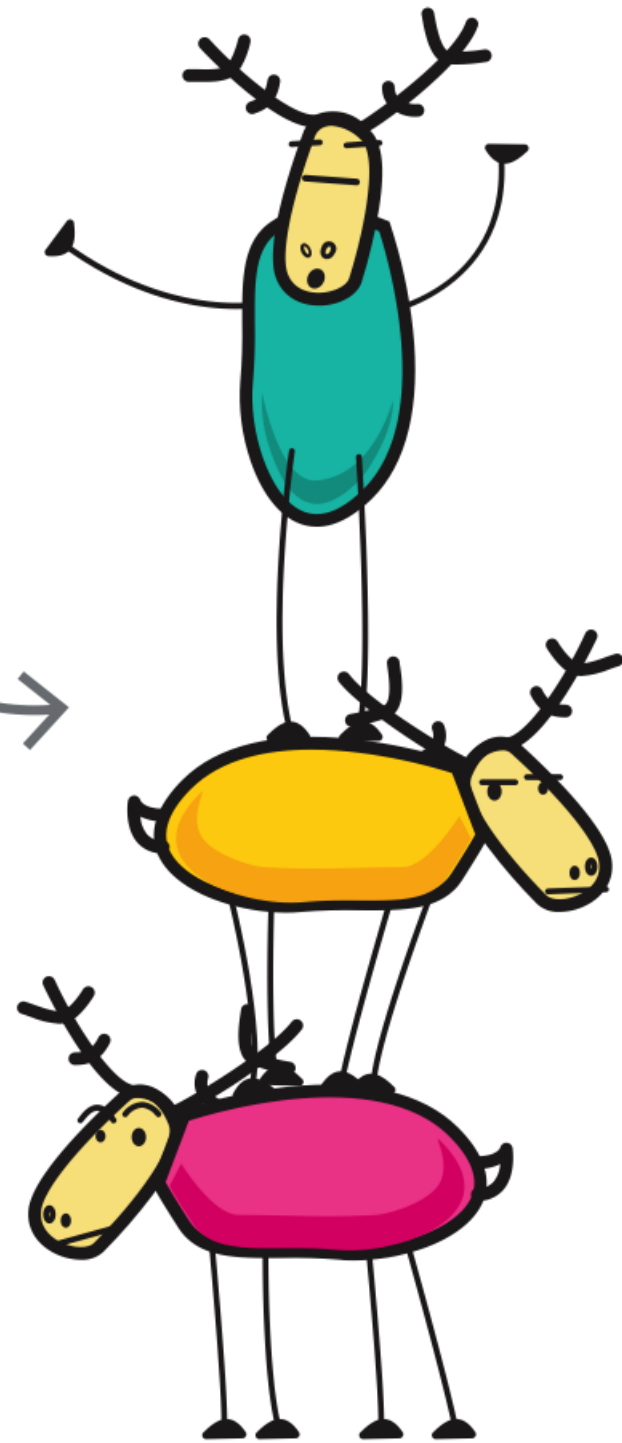


# elastic

Developer 🥑



ELK Stack!  
Get it?



**E** Elasticsearch

**L** Logstash

**K** Kibana

lyft



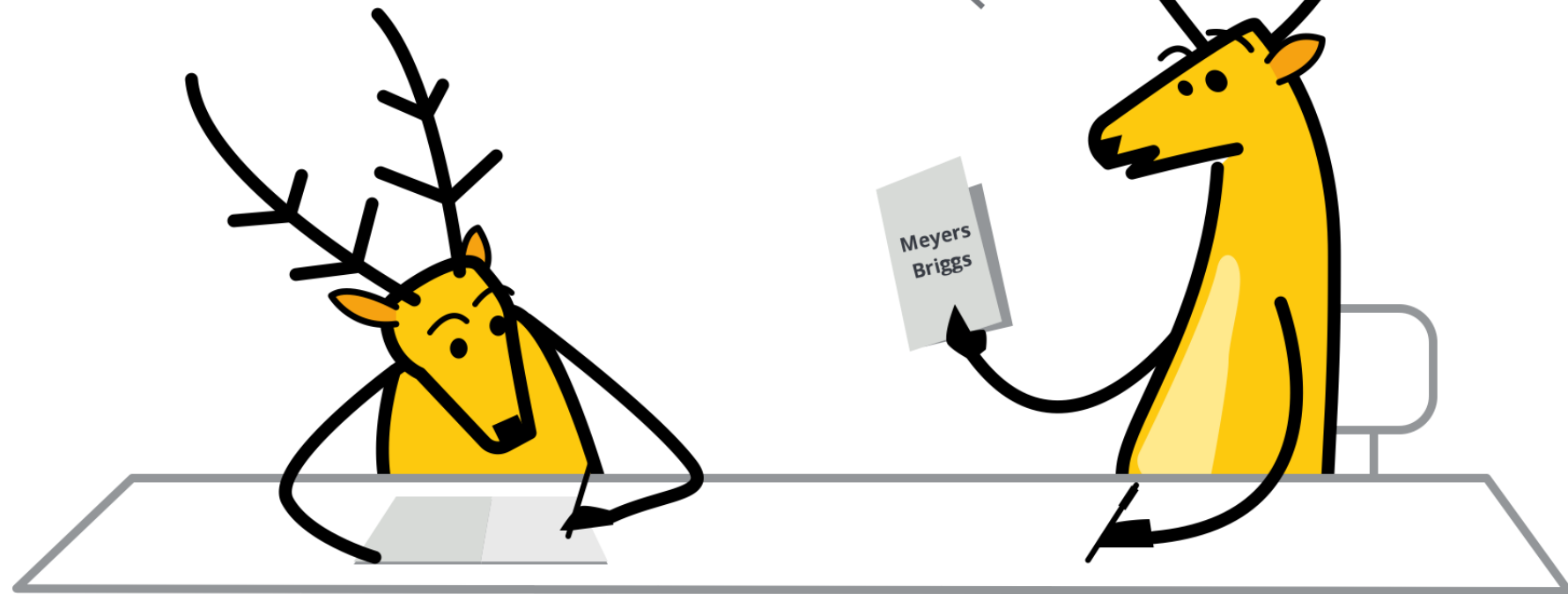
slack

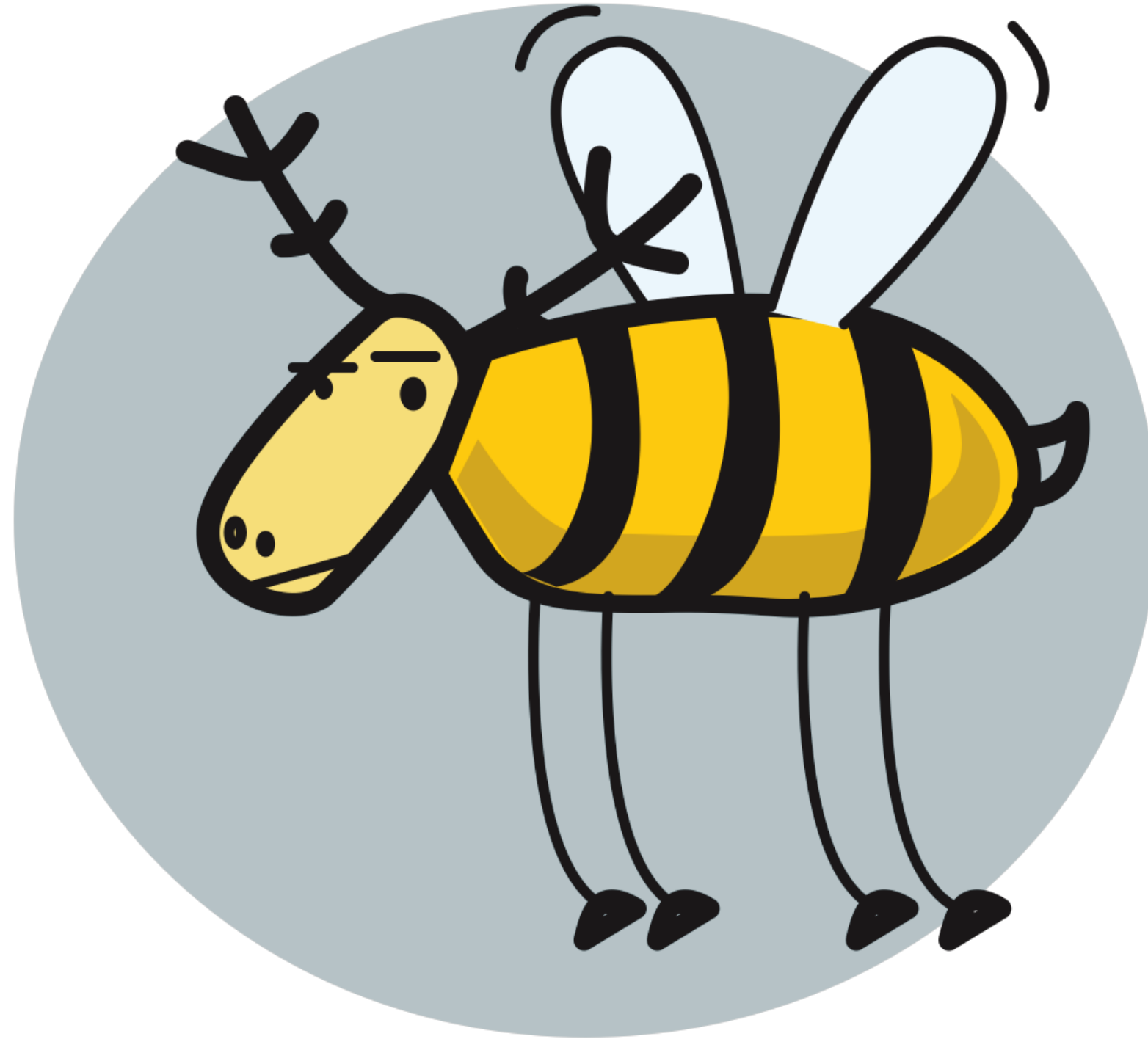


fitbit



*Apparently, I'm an  
ELKB personality.*

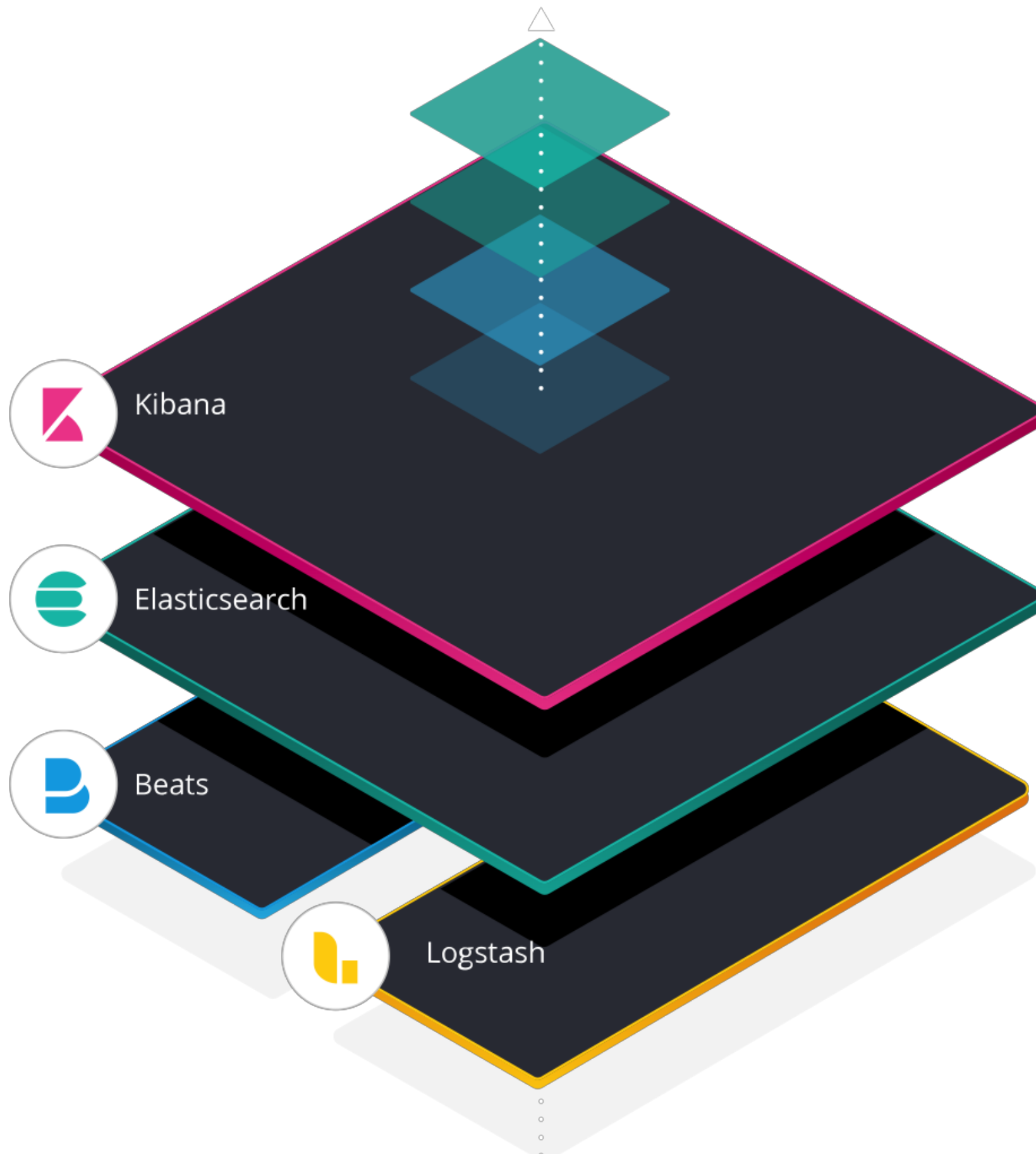








# elastic stack





# Disclaimer

I build **highly** monitored Hello World  
apps

# Setup (optional)

```
$ git clone https://github.com/xeraa/java-logging.git
```

```
$ docker load -i <image>.tar (optional)
```

```
$ docker-compose up
```



# Example: Java

SLF4J, Logback, MDC  
with logstash-logback-encoder

Alternative <https://github.com/vy/log4j2-logstash-layout>

# And Everywhere Else

.NET: NLog

JavaScript: Winston

Python: structlog

PHP: Monolog



# Anti-Pattern: `print`

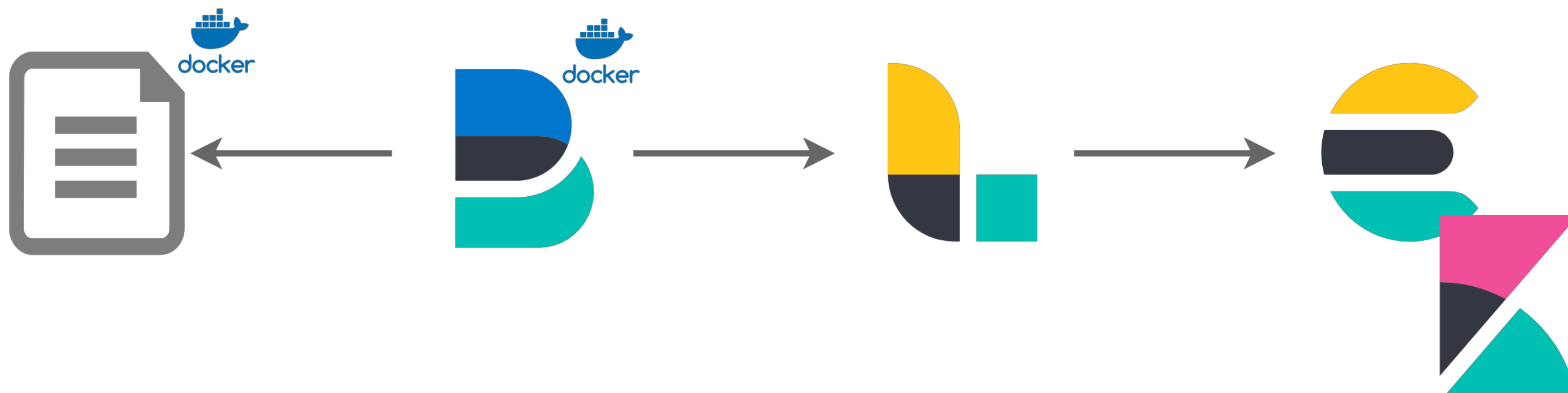
```
System.out.println("Oops");
```

# Anti-Pattern: Coupling



# Parse





# Bind Mount Logs

```
java_app:  
  volumes:  
    - './logs-docker/:/logs/'  
  ...  
  
filebeat_for_logstash:  
  volumes:  
    - './logs-docker/:/mnt/logs/:ro'  
  ...
```



# Collect Log Lines

```
filebeat.inputs:
```

```
- type: log
```

```
  paths:
```

```
    - /mnt/logs/*.log
```

# Metadata

processors:

- add\_host\_metadata: ~

Setting for negate	Setting for match	Result	Example pattern: ^b
false	after	Consecutive lines that match the pattern are appended to the previous line that doesn't match.	<pre> a b b c b b   } abb   } cbb </pre>
false	before	Consecutive lines that match the pattern are prepended to the next line that doesn't match.	<pre> b b a b b c   } bba   } bbc </pre>
true	after	Consecutive lines that don't match the pattern are appended to the previous line that does match.	<pre> b a c b d e   } bac   } bde </pre>
true	before	Consecutive lines that don't match the pattern are prepended to the next line that does match.	<pre> a c b b d e b   } acb   } deb </pre>



# Test Multiline Pattern

[https://www.elastic.co/guide/en/beats/filebeat/current/\\_test\\_your\\_regexp\\_pattern\\_for\\_multiline.html](https://www.elastic.co/guide/en/beats/filebeat/current/_test_your_regexp_pattern_for_multiline.html)

# Grok

<https://github.com/logstash-plugins/logstash-patterns-core/blob/master/patterns/grok-patterns>

# Dev Tools

# Grok Debugger

## Sample Data

```
1 [2018-11-16 01:16:59.983] ERROR net.xeraa.logging.LogMe [main] - user_experience=👎, ses
```

## Grok Pattern

```
1 \[%{TIMESTAMP_ISO8601:timestamp}\] %{LOGLEVEL:loglevel}
```

> Custom Patterns

[Simulate](#)

## Structured Data

```
1 {  
2   "loglevel": "ERROR",  
3   "timestamp": "2018-11-16 01:16:59.983"  
4 }
```



```
[2018-09-28 10:30:38.516] ERROR net.xeraa.logging.LogMe [main] -
    user_experience=🤔, session=46, loop=15 -
    Wake me up at night
java.lang.RuntimeException: Bad runtime...
    at net.xeraa.logging.LogMe.main(LogMe.java:30)
```

```
^\[%{TIMESTAMP_ISO8601:@timestamp}\] %SPACE%{LOGLEVEL:log.level}
%SPACE%{USERNAME:log.package}%SPACE% \[%{WORD:log.method}\]
%SPACE%- %SPACE%{GREEDYDATA:log.labels}%SPACE%- %SPACE%
%{GREEDYDATA:message} (?: \n+ (?<stacktrace> (?: . | \r | \n)+) )?
```

# Elastic Common Schema

<https://github.com/elastic/ecs>

## Event fields

The event fields are used for context information about the data itself.

Field	Description	Level	Type	Example
event.id	Unique ID to describe the event.	core	keyword	8a4f500d
event.category	Event category. This can be a user defined category.	core	keyword	metrics
event.type	A type given to this kind of event which can be used for grouping. This is normally defined by the user.	core	keyword	nginx-stats-metrics
event.action	The action captured by the event. The type of action will vary from system to system but is likely to include actions by security services, such as blocking or quarantining; as well as more generic actions such as login	core	keyword	reject

# Machine Learning Data Visualizer

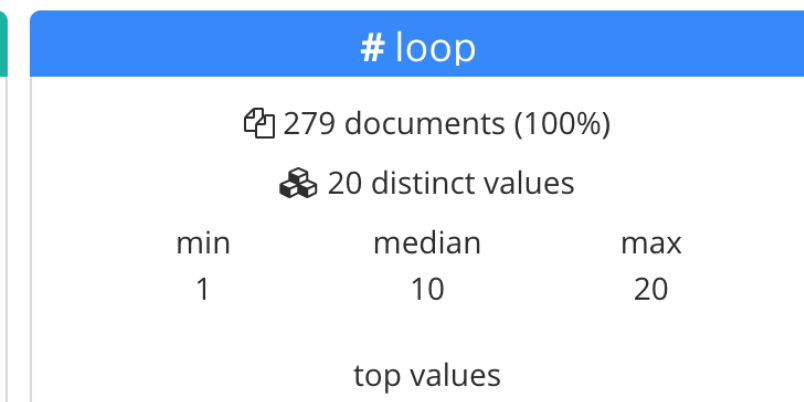
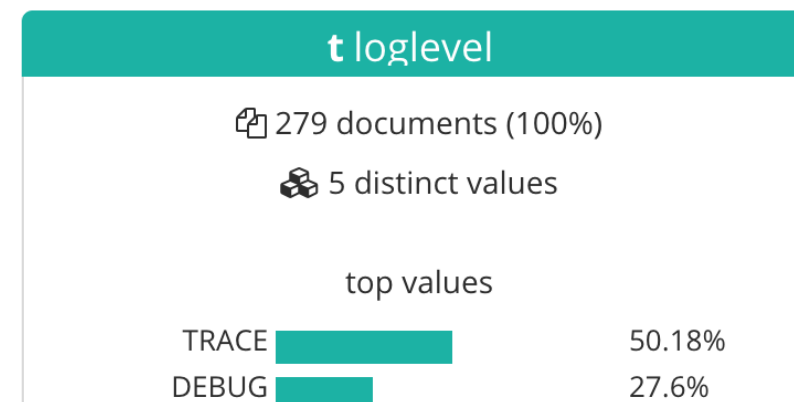
```
28 [2018-11-16 01:16:59.976] DEBUG net.xeraa.logging.LogMe [main] - session=94, loop=14 - Collect
29 [2018-11-16 01:16:59.977] TRACE net.xeraa.logging.LogMe [main] - session=43, loop=15 - Iteration
30 [2018-11-16 01:16:59.983] ERROR net.xeraa.logging.LogMe [main] - user_experience=🤔, session=43
31 java.lang.RuntimeException: Bad runtime...
```

## Summary

Number of lines analyzed	293
Format	semi_structured_text
Grok pattern	\[%{TIMESTAMP_ISO8601:timestamp}\] %{LOGLEVEL:loglevel}.*? .*?\[.*?\].*? .*?\bsessi
Time field	timestamp
Time format	YYYY-MM-dd HH:mm:ss.SSS

[Override settings](#)

## File stats



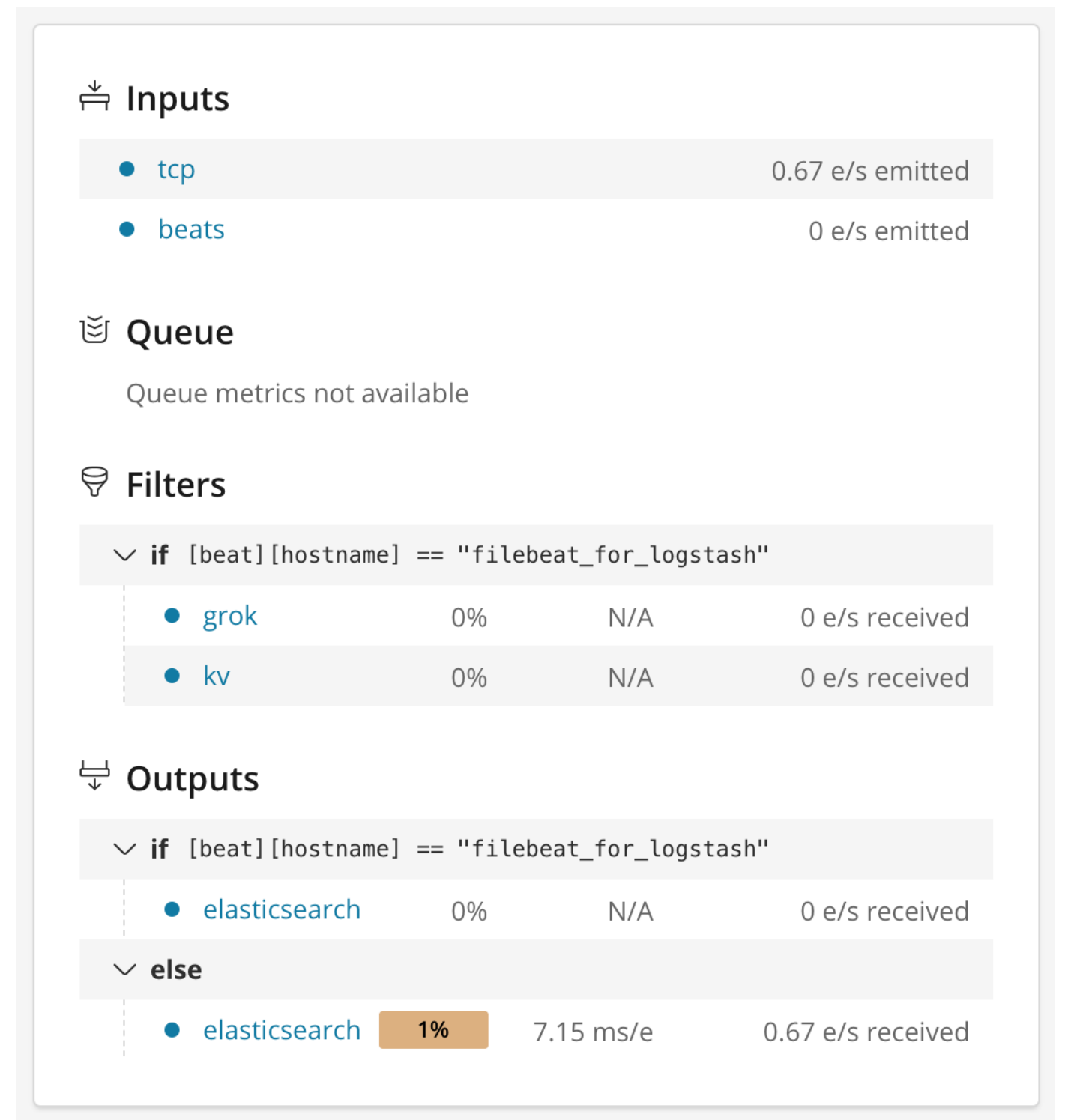
# Logstash Key Value Filter for MDC

```
kv {  
  source => "labels"  
  field_split => ","  
  trim_key => " "  
}
```



# Monitoring: Logstash Pipeline

## Plus other components

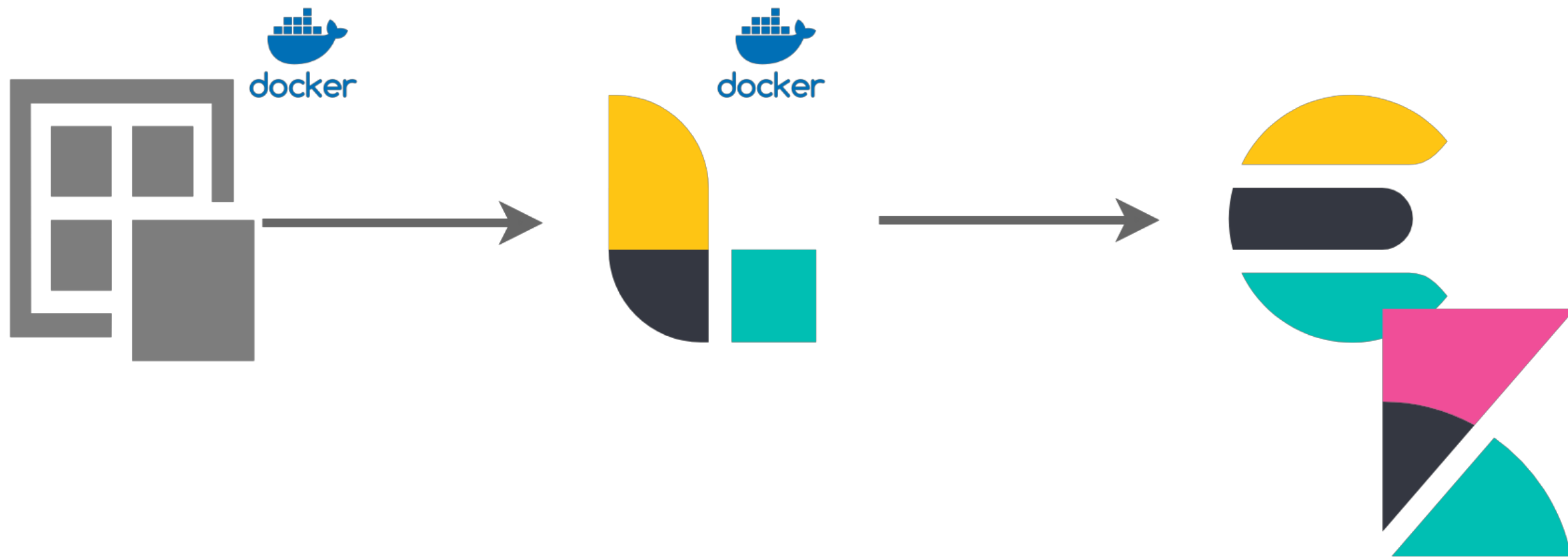


Pro: No change

Con: Regular expression, multiline,  
format changes

# Send







# logback.xml

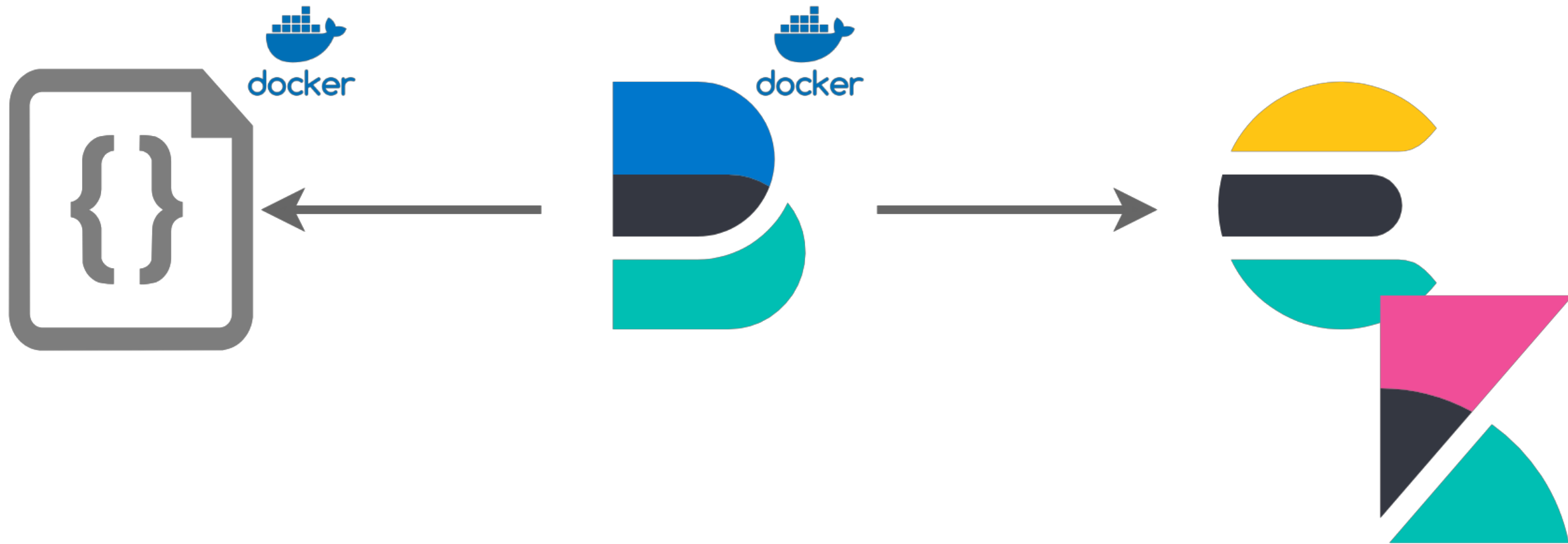
```
<appender name="logstash" class="net.logstash.logback.appender.LogstashAccessTcpSocketAppender">  
  <destination>logstash:4560</destination>  
  <encoder class="net.logstash.logback.encoder.LogstashEncoder"/>  
</appender>
```

Pro: No files

Con: Outages & coupling

# Structure







# Collect JSON

```
filebeat.input:  
- type: log  
  paths:  
    - /mnt/logs/*.json  
  json:  
    message_key: message  
    keys_under_root: true
```

# Stack(trace) Hash

# Bonues: Multi-Index

```
output.elasticsearch:
```

```
  hosts: ["http://localhost:9200"]
```

```
  indices:
```

```
    - index: "warning-%{[agent.version]}-%{+yyyy.MM.dd}"
```

```
      when.contains:
```

```
        message: "WARN"
```

```
    - index: "error-%{[agent.version]}-%{+yyyy.MM.dd}"
```

```
      when.contains:
```

```
        message: "ERR"
```

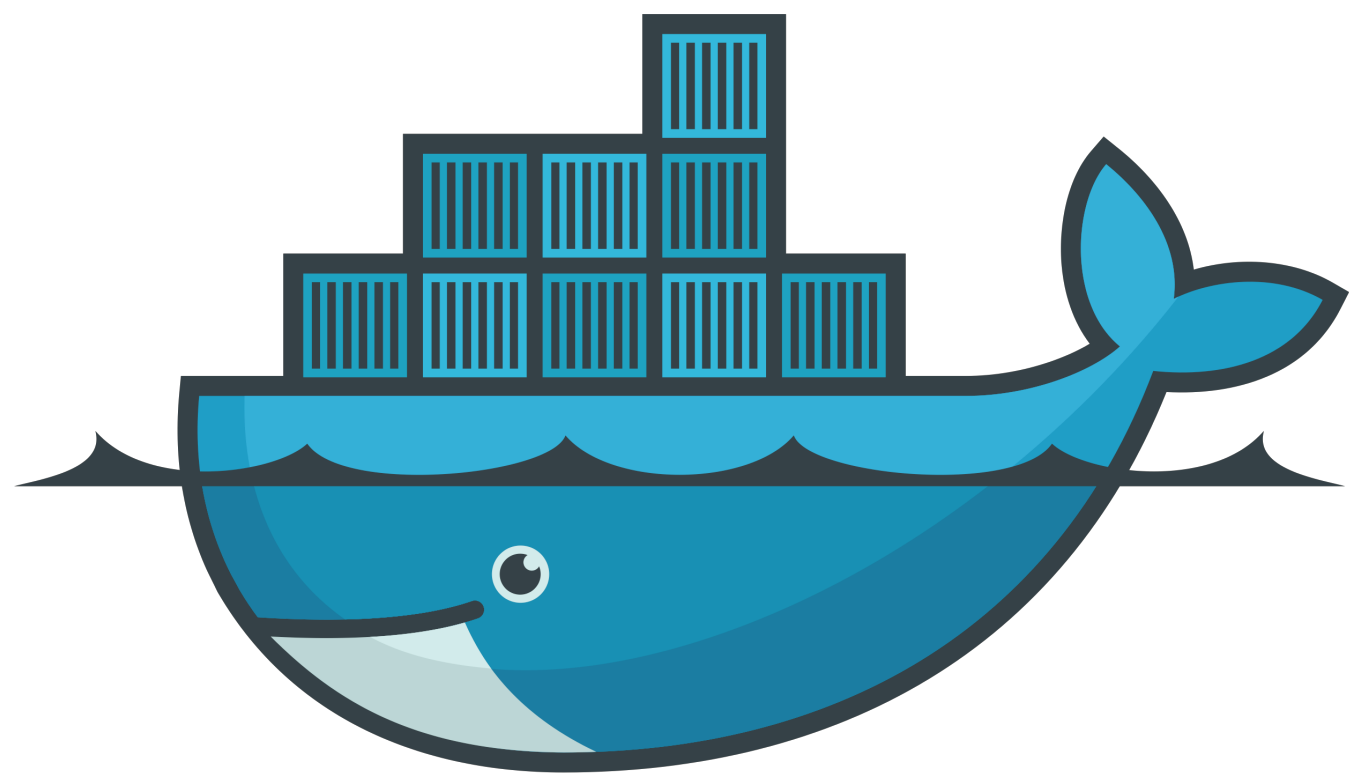
Pro: Right format

Con: JSON serialization overhead

# Containerize



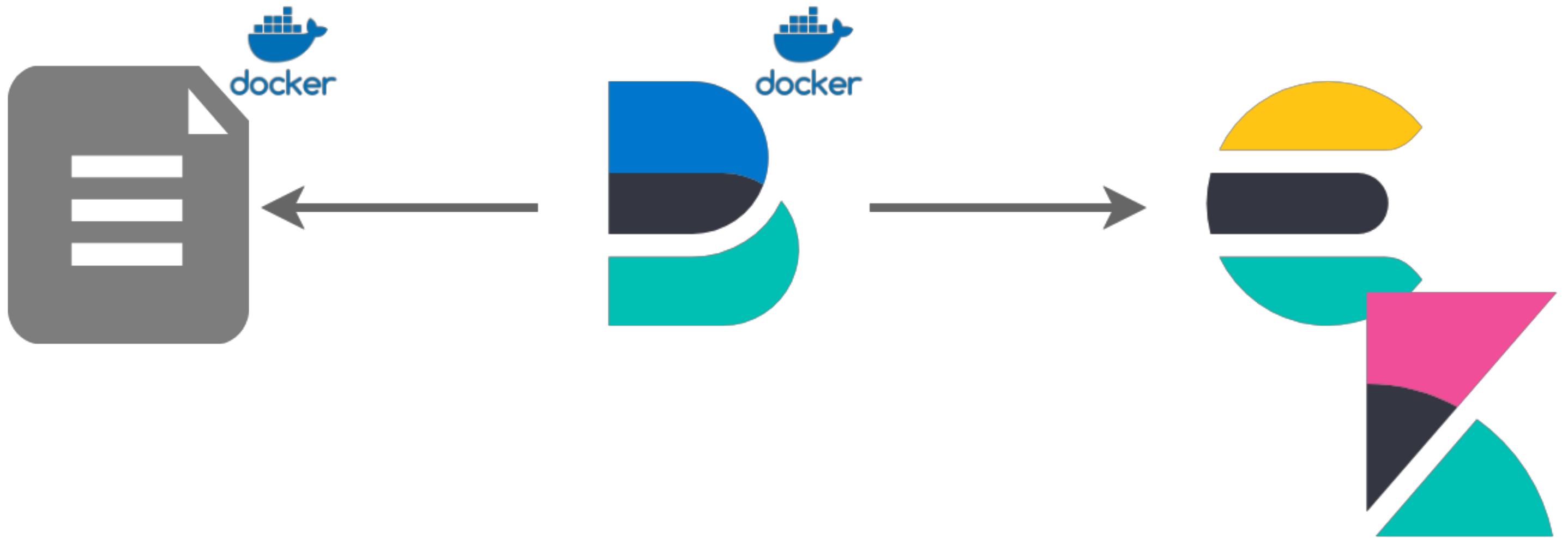




docker

# Where to put Filebeat?

## Sidecar



[https://github.com/elastic/beats/tree/  
master/deploy/docker](https://github.com/elastic/beats/tree/master/deploy/docker)

# Docker Logs

```
filebeat.autodiscover:
```

```
  providers:
```

```
    - type: docker
```

```
      hints.enabled: true
```

```
processors:
```

```
  - add_docker_metadata: ~
```



# Metadata

No Docker metadata with the other methods

```
"docker": {
  "container": {
    "labels": {
      "app": "fizzbuzz",
      "co_elastic_logs/multiline_match": "after",
      "com_docker_compose_config-hash": "41520c6cf2b6a1f3dae4f16d0a6fd76760cdfc38fbfe43a3a3be2e09bdd1b8b5",
      "environment": "production",
      "co_elastic_logs/multiline_pattern": "^\\\[",
      "co_elastic_logs/multiline_negate": "true",
      "com_docker_compose_oneoff": "False",
      "com_docker_compose_project": "java-logging",
      "com_docker_compose_service": "java_app",
      "com_docker_compose_container-number": "1",
      "com_docker_compose_version": "1.23.2"
    }
  }
}
```

# Missing the Last Line

Waiting for the newline

# Hints

## labels:

- "app=fizzbuzz"
- "co.elastic.logs/multiline.pattern=^\\\\"
- "co.elastic.logs/multiline.negate=true"
- "co.elastic.logs/multiline.match=after"

# Registry File

```
filebeat.registry.path: /usr/share/filebeat/data/registry
```

# Ingest Pipeline

```
output.elasticsearch:  
  hosts: ["http://elasticsearch:9200"]  
  index: "docker"  
  pipelines:  
    - pipeline: "parse_java"  
      when.contains:  
        container.name: "java_app"
```



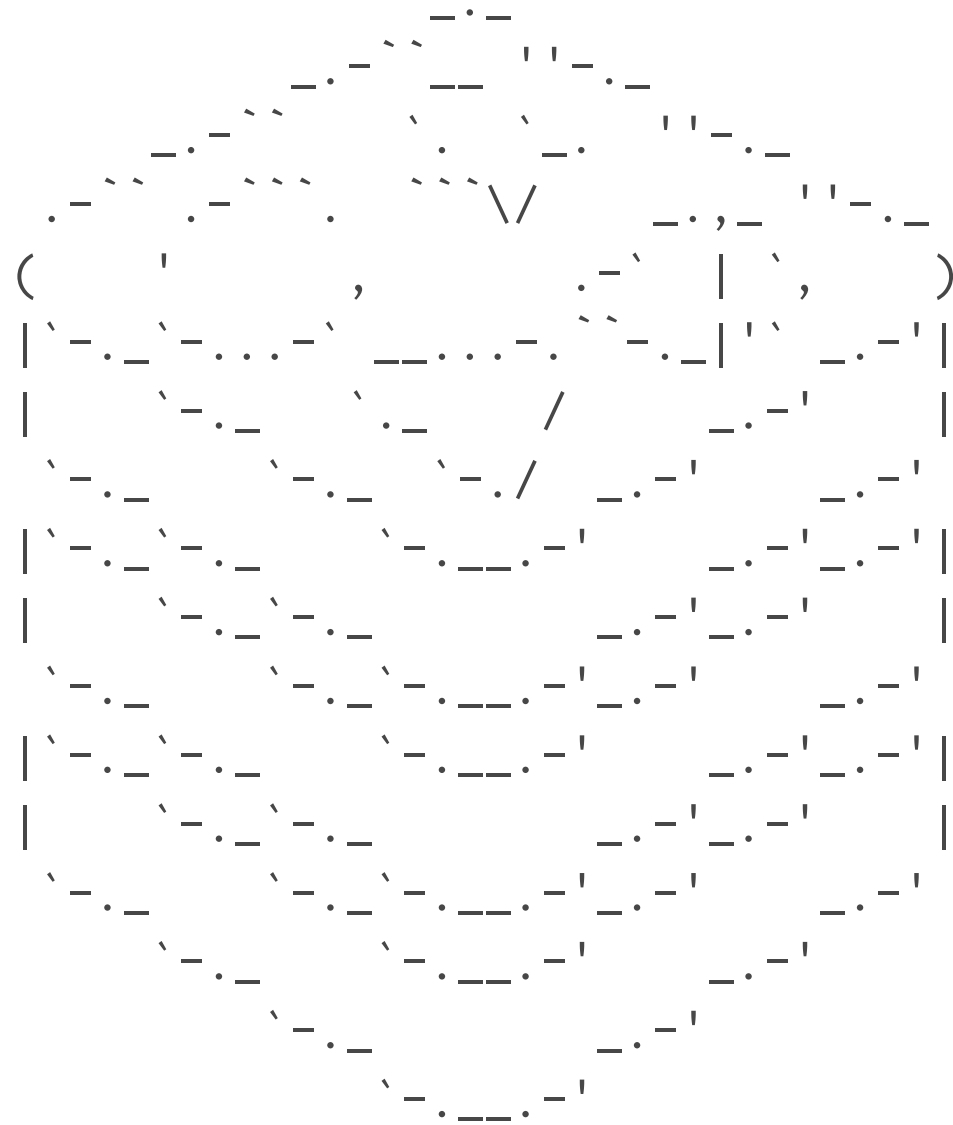
# Ingest Pipeline

```
{
  "description" : "Parse Java log lines",
  "processors": [
    {
      "grok": {
        "field": "message",
        "patterns": [
          "^\\[\\%{TIMESTAMP_ISO8601:timestamp}\\]\\%{SPACE}\\%{LOGLEVEL:log.level}\\%{SPACE}\\%{USERNAME:log.package}\\%{SPACE}\\[\\%{WORD:log.method}\\]\\%{SPACE}-\\%{SPACE}\\%{GREEDYDATA:labels}\\%{SPACE}-\\%{SPACE}\\%{GREEDYDATA:message_parsed}\\%{SPACE}(?:\\n+(?<stacktrace>(?:.|\\r|\\n)+))?"
        ],
        "ignore_failure": true
      }
    }
  ]
}
```

# Unknown Fields

```
? log.labels          ▲ session=69, loop=20
t log.level           WARN
? log.method          ▲ main
# log.offset          19,744
? log.package         ▲ net.xeraa.logging.LogMe
t message             [2019-05-21 05:02:07.458] WARN net.xeraa.logging.LogMe
                        [main] - session=69, loop=20 - Investigate tomorrow
? message_parsed     ▲ Investigate tomorrow
t stream              stdout
☉ suricata.eve.timestamp May 21, 2019 @ 07:02:07.459
? timestamp          ▲ 2019-05-21 05:02:07.458
```

# ASCII Art



Redis 4.0.9 (00000000/0) 64 bit

Running in stand alone mode

Port: 6379

PID: 55757

<http://redis.io>

# Configuration Templates

```
filebeat.autodiscover:  
  providers:  
    - type: docker  
  templates:  
    - condition:  
      equals:  
        docker.container.image: redis  
  config:  
    - type: docker  
      containers.ids:  
        - "${data.docker.container.id}"  
  exclude_lines: ["^\s+[\-\`('.|_)]"]
```

# Who Logs the Logger

Avoid loops

Process without `-e`

```
filebeat.yml: logging.to_files: true
```

Pro: Hot 🍌

Con: Complexity

# Orchestrator





# kubernetes



# Where to put Filebeat?

# DaemonSet

[https://github.com/elastic/beats/tree/  
master/deploy/kubernetes](https://github.com/elastic/beats/tree/master/deploy/kubernetes)

# Metadata

Either in cluster or outside

processors:

- add\_kubernetes\_metadata:  
  in\_cluster: true
- add\_kubernetes\_metadata:  
  in\_cluster: false  
  host: <hostname>  
  kube\_config: \${HOME}/.kube/config

```
{
  "host": "172.17.0.21",
  "port": 9090,
  "kubernetes": {
    "container": {
      "id": "382184ecdb385cfd5d1f1a65f78911054c8511ae009635300ac28b4fc357ce51",
      "image": "my-java:1.0.0",
      "name": "my-java"
    },
    "labels": {
      "app": "java",
    },
    "namespace": "default",
    "node": {
      "name": "minikube"
    },
    "pod": {
      "name": "java-2657348378-k1pnh"
    }
  },
}
```

# More Metadata

Add: Cloud, local timezone, process

Drop: Events, fields

Rename: Fields

Dissect, DNS reverse lookup

# Configuration Templates

```
filebeat.autodiscover:  
  providers:  
    - type: kubernetes  
  templates:  
    - condition:  
      equals:  
        kubernetes.namespace: redis  
  config:  
    - type: docker  
      containers.ids:  
        - "${data.kubernetes.container.id}"  
  exclude_lines: ["^\s+[\-\`('.|_)]"]
```

# Customize Indices

```
output.elasticsearch:
```

```
  index: "%{[kubernetes.namespace]:filebeat}-%{[beat.version]}-%{+yyyy.MM.dd}"
```

Pro: Hot 🍌 🍌 🍌

Con: Complexity++



# Moar





D



# Docker Containers ⚙️ Default

Showing the last 1 minute of data from the time period



Hosts



Kubernetes



Docker

🔍 Search for infrastructure data... (e.g. h

Metric: Outbound Traffic ▾

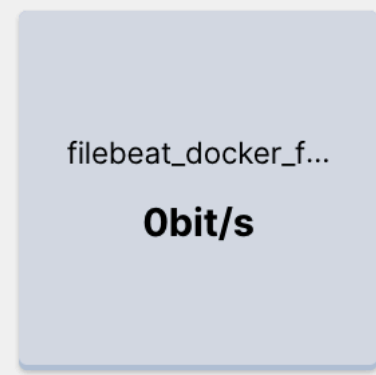
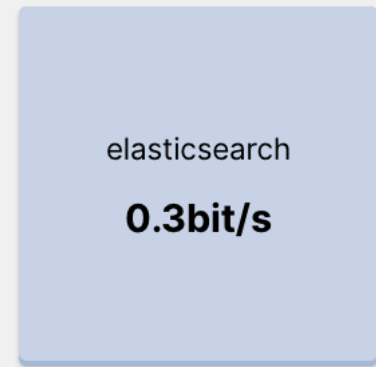
Group By: All ▾

📅 05/21/2019 7:46:35 AM

▶ Auto-refresh

🗺 Map View 📄 Table View

All | 7



0bit/s

5.2bit/s





D

Logs



container.name : "java\_app"

Default Customize

streaming...

Stop streaming



2019-05-21 07:02:07.356

[ERROR] [2019-05-21 05:02:07.354] ERROR net.xeraa.logging.LogMe [main] - user\_experience=🤪, session=53, loop=15 - Wake me up at night  
java.lang.RuntimeException: Bad runtime...  
at net.xeraa.logging.LogMe.main(LogMe.java:30)

09 PM

2019-05-21 07:02:07.377

[TRACE] [2019-05-21 05:02:07.376] TRACE net.xeraa.logging.LogMe [main] - session=39, loop=16 - Iterati on '16' and session '39'

Tue 21

2019-05-21 07:02:07.380

[DEBUG] [2019-05-21 05:02:07.379] DEBUG net.xeraa.logging.LogMe [main] - session=39, loop=16 - Collect in development

03 AM

2019-05-21 07:02:07.395

[TRACE] [2019-05-21 05:02:07.394] TRACE net.xeraa.logging.LogMe [main] - session=27, loop=17 - Iterati on '17' and session '27'

2019-05-21 07:02:07.400

[DEBUG] [2019-05-21 05:02:07.400] DEBUG net.xeraa.logging.LogMe [main] - session=27, loop=17 - Collect in development

06 AM

2019-05-21 07:02:07.407

[TRACE] [2019-05-21 05:02:07.404] TRACE net.xeraa.logging.LogMe [main] - session=94, loop=18 - Iterati on '18' and session '94'

2019-05-21 07:02:07.412

[INFO] [2019-05-21 05:02:07.411] INFO net.xeraa.logging.LogMe [main] - session=94, loop=18 - Collect in production

09 AM

2019-05-21 07:02:07.423

[TRACE] [2019-05-21 05:02:07.419] TRACE net.xeraa.logging.LogMe [main] - session=12, loop=19 - Iterati on '19' and session '12'

12 PM

2019-05-21 07:02:07.443

[DEBUG] [2019-05-21 05:02:07.443] DEBUG net.xeraa.logging.LogMe [main] - session=12, loop=19 - Collect in development

2019-05-21 07:02:07.457

[TRACE] [2019-05-21 05:02:07.456] TRACE net.xeraa.logging.LogMe [main] - session=69, loop=20 - Iterati on '20' and session '69'

03 PM

2019-05-21 07:02:07.459

[WARN] [2019-05-21 05:02:07.458] WARN net.xeraa.logging.LogMe [main] - session=69, loop=20 - Investigate tomorrow

06 PM

Streaming new entries last updated 0s ago

# Index Patterns

Time based (default: daily)

Versioned

# Sizing

Daily volume \* Retention \* Replication

Number of shards

# Rollover

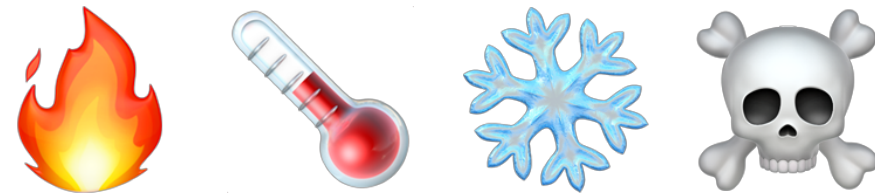
New index on condition

Size

Time

Number of documents

# Index Lifecycle Management



# Order

<https://github.com/elastic/elasticsearch/blob/7.1/x-pack/plugin/core/src/main/java/org/elasticsearch/xpack/core/indexlifecycle/TimeseriesLifecycleType.java>

```
static final List<String> ORDERED_VALID_HOT_ACTIONS = Arrays.asList(
    SetPriorityAction.NAME, UnfollowAction.NAME, RolloverAction.NAME
);
static final List<String> ORDERED_VALID_WARM_ACTIONS = Arrays.asList(
    SetPriorityAction.NAME, UnfollowAction.NAME, ReadOnlyAction.NAME,
    AllocateAction.NAME, ShrinkAction.NAME, ForceMergeAction.NAME
);
static final List<String> ORDERED_VALID_COLD_ACTIONS = Arrays.asList(
    SetPriorityAction.NAME, UnfollowAction.NAME, AllocateAction.NAME, FreezeAction.NAME
);
static final List<String> ORDERED_VALID_DELETE_ACTIONS = Arrays.asList(
    DeleteAction.NAME
);
```



# Frozen Indices

<https://www.elastic.co/guide/en/elasticsearch/reference/6.6/frozen-indices.html>

# Ratio Heap : Storage

Index > Frozen Index > Closed Index

Read-only

# Throttled Thread Pool

1 parallel search / node

100 in queue

# Conclusion

# Examples

<https://github.com/xeraa/java-logging>

**gcp.data.highio.1** Data Ingest Master

An I/O optimized Elasticsearch instance.

**Fault tolerance**

1 zone  2 zones  3 zones

**RAM per Node**

1 GB 2 GB 4 GB 8 GB 16 GB 32 GB 64 GB

Nodes  = RAM per Zone 4 GB

**Summary**

4 GB RAM 120 GB storage × 1 node × 2 zones = 8 GB RAM 240 GB storage

> User setting overrides

**gcp.data.highstorage.1** Data Ingest Master

A storage optimized Elasticsearch instance.

**Fault tolerance**

1 zone  2 zones  3 zones

**RAM per Node**

2 GB 4 GB 8 GB 16 GB 32 GB 64 GB

Nodes  = RAM per Zone 4 GB

**Summary**

4 GB RAM 400 GB storage × 1 node × 2 zones = 8 GB RAM 800 GB storage

> User setting overrides

Machine Learning 1 configuration

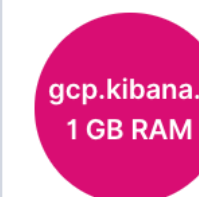
**gcp.ml.1** Machine Learning

## Summary

<b>Version</b>	v7.1.1
<b>ES data memory</b>	16 GB
<b>ES data storage</b>	1.02 TB
<b>Total memory</b>	17.5 GB
<b>Total storage</b>	1.02 TB
<b>Hourly rate</b>	\$0.7328
<b>Monthly rate</b>	\$534.94

## Architecture

### Zone 1



### Zone 2



- gcp.apm.1
- gcp.data.highio.1
- gcp.data.highstorage.1
- gcp.kibana.1

# Helm & Operator

<https://github.com/elastic/helm-charts>

<https://github.com/elastic/cloud-on-k8s>

# PS: Free Security

<https://www.elastic.co/blog/security-for-elasticsearch-is-now-free>

TLS, file & native realm, RBAC



Parse 

Send 

Structure 

Containerize 

Orchestrate 

# And More

## Metrics

## APM / Tracing

## Health

# Questions?

Philipp Krenn

@xeraa