qaware.de

QA|WARE
SOFTWARE ENGINEERING

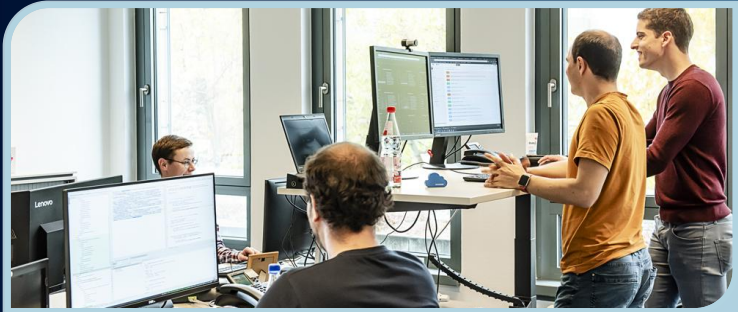
High-Performance Observability Data Pipelines mit Vector

Andreas Kowasch
andreas.kowasch@qaware.de
@akowasch

Mastering Observability 2024

Wir übernehmen Verantwortung und Risiken: Für Prototypen bis hin zu großen Programmen. Wir liefern. Garantiert.

QA|WARE
SOFTWARE ENGINEERING



200 Engineers

35 M€ Umsatz

München, Mainz, Darmstadt, Rosenheim

seit 18 Jahren durchgehend Erfolge in anspruchsvollsten Projekten

Top Provider NPS 100

Top Arbeitgeber: 97% sagen: “QAware ist ein sehr guter Arbeitsplatz"

Wir agieren in kompakten und eingeschwungenen **cross-funktionalen Teams** aus Beratern, Entwicklern und Managern mit folgenden Garantien:

- **Erfolgsgarantie:** Wir übernehmen Verantwortung und tragen Ihre Risiken mit z.B. über Festpreise.
- **Qualitätsgarantie:** Nachhaltige und sichere Software von höchster Qualität– über KPIs belegt und vertraglich fixiert.
- **Zufriedenheitsgarantie:** Ihr werdet mit uns glücklich sein! Auch in kleinen Liefer-artefakten. So sicher das wir auch gern ein Teil unser Vergütung daran binden.

Wir verstehen uns als **Enabler**. Wir transformieren IT-Organisationen direkt über die Zusammenarbeit im Projekt.

Unsere Expertise

Cloud Native Transformation & Host-Ablöse: Reiseleiter in die Zukunft

- Allianz LEAP und Syncier Cloud
- Hellmann HeRo
- Ericsson KDTMES

Data & AI Value: Daten erschließen, vernetzen & wertvoll machen

- BMW Aftersales Info Research
- Krones & MaidITC AI Optimizer
- BMW GenAI Plattform

Digital Business Booster: Geschäftskritische Innovationen ermöglichen & beschleunigen

- DT Magenta Voice
- ubitricity charging platform
- Raiffeisenbank Südtirol Next

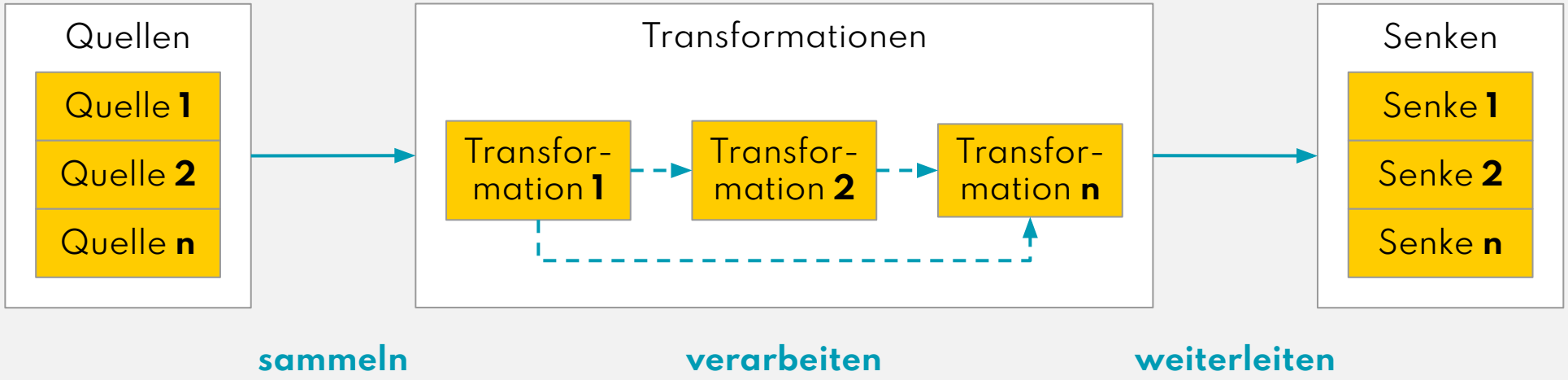
Agenda

- Einführung
- Deployment
- Demo
- Limitierungen & Fazit

Einführung
Was sind Observability-Pipelines?

- Sammeln, Verarbeiten und Weiterleiten von Observability-Daten.

- Gerichteter azyklischer Graph von Komponenten.
  - Jede Komponente ist ein Knoten im Diagramm mit gerichteten Kanten.
  - Daten müssen in eine Richtung fließen, von Quellen zu Senken.
  - Komponenten können null oder mehr Ereignisse erzeugen.



Einführung
Was ist Vector?

Daten und Fakten

- Tool zum Shippen von **Logs**, Metriken (und Traces)
- Open-Source (Mozilla Public License, Version 2.0)
- Entwickelt in Rust
- Multi-Plattform
- Zustellungsgarantien
- 16.8k Stars, 1.4k Forks, 429 Contributor, 78 Releases
- Letztes Release: **v0.38.0** am 7. Mai 2024

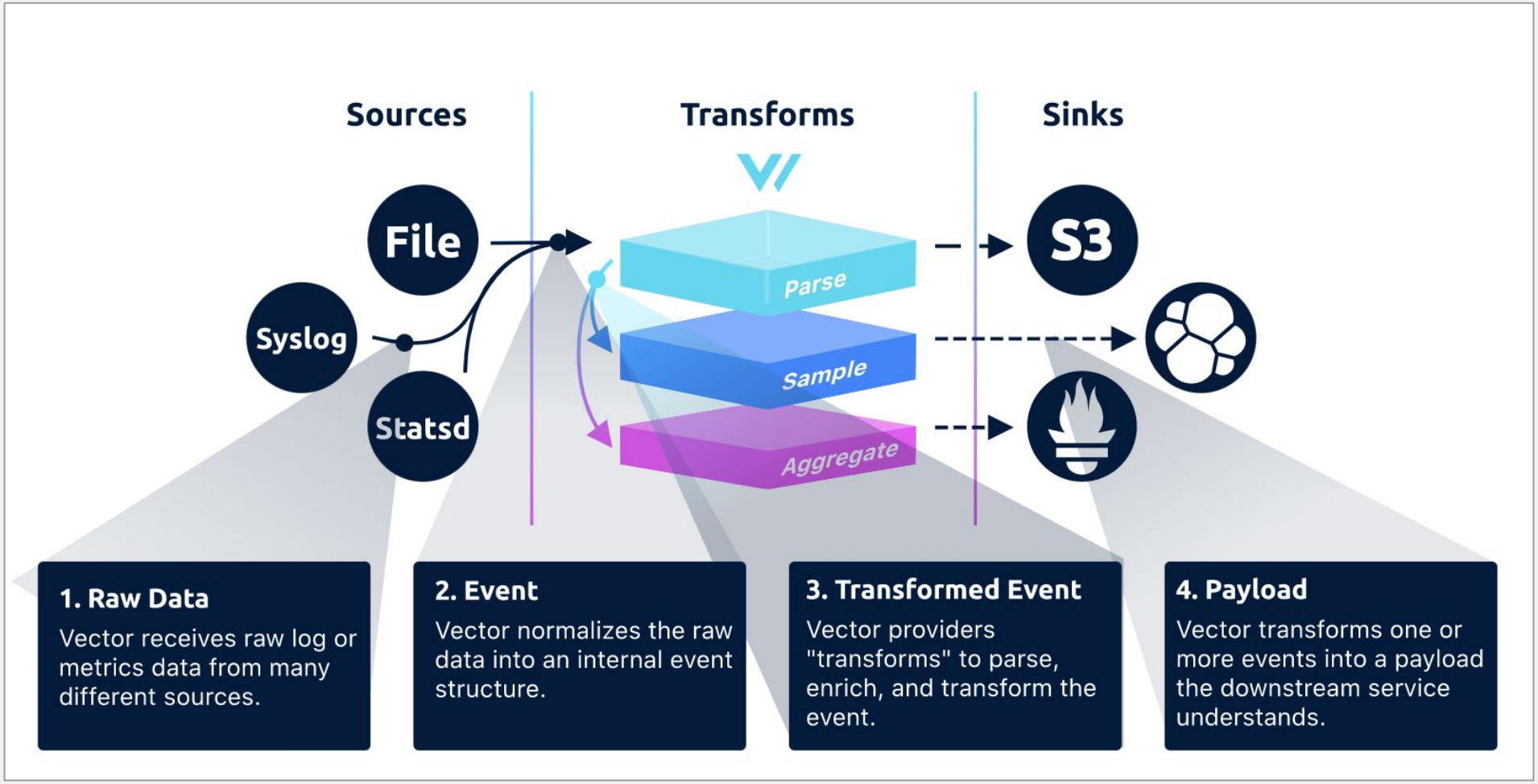
Historie

- 2019: Initiales Release durch Timber.io
- 2021: Übernahme von Timber.io durch Datadog

Alternativen (Logs)

- Logstash
- Filebeat
- Fluentd
- Fluent Bit
- Promtail
- OpenTelemetry Collector
- Grafana Alloy
- …

Einführung
Was ist Vector?



Einführung
Komponenten

- Quellen

AMQP
Apache Metrics
AWS ECS metrics
AWS Kinesis Firehose
AWS S3
AWS SQS
Datadog agent
Demo Logs
dnstap
Docker logs
EventStoreDB metrics
Exec
File
File Descriptor
Fluent
GCP PubSub
Heroku Logplex
Host metrics
HTTP Client/Server
Internal logs/metrics
JournalD
Kafka
Kubernetes logs
Logstash
MongoDB metrics
NATS
NGINX metrics
OpenTelemetry
PostgreSQL metrics
Prometheus Pushgateway
Prometheus remote write
Prometheus scrape
Pulsar
Redis
Socket
Splunk HEC
StatsD
stdin
Syslog
Vector

Einführung
Komponenten

- Transformationen

Aggregate
AWS EC2 metadata
Dedupe
Filter
Log to metric
Lua
Metric to log
Reduce
Remap
Route
Sample
Tag cardinality limit
Throttle

Input

```
{
  "status": 200,
  "timestamp": "2021-03-01T19:19:24.646170Z",
  "message": "SUCCESS",
  "username": "ub40fan4life"
}
```

Transformation

```
transforms:
  modify:
    type: remap
    inputs:
      - logs
    source: |-
      . = parse_json!(string!(.message))
      .timestamp = to_unix_timestamp(to_timestamp!(.timestamp))
      del(.username)
      .message = downcase(string!(.message))
```

Output

```
{
  "status": 200,
  "timestamp": 1614626364
  "message": "success",
}
```

Einführung
Vector Remap Language (VRL)

Functions

- Array
- Codec
- Coerce
- Convert
- Debug
- Enrichment
- Enumerate
- Event
- Path
- Cryptography
- IP
- Number
- Object
- Parse
- Random
- String
- System
- Timestamp
- Type

Literal Expressions

- Array
- Boolean
- Float
- Integer
- Null
- Object
- RegEx
- String
- Timestamp

Dynamic Expressions

- Abort
- Arithmetic
- Assignment
- Block
- Coalesce
- Comparison
- Function call
- If
- Index
- Logical
- Path
- Variable

Errors

- Compile-time, Runtime

Einführung
Komponenten

- Anreicherungstabellen

Beispiel:

IoT-Devices

Tabelle (CSV)

```
status_code,status_message
1,"device status online"
2,"device status offline"
3,"device status connection error"
4,"device status transmitting"
5,"device status transmission complete"
6,"device status not responding"
```

Vector-Konfiguration (Anreicherung)

```
enrichment_tables:
  iot_status:
    type: file
    file:
      path: /etc/vector/iot_status.csv
      encoding:
        type: csv
    schema:
      status_code: integer
      status_message: string
```

Vector-Konfiguration (Transformation)

```
transforms:
  enrich_iot_status:
    type: remap
    inputs:
      - datadog_agents
    source: |-
      . = parse_json!(.status_message)
      status_code = del(.status_code)
      row = get_enrichment_table_record("iot_status", {"status_code" : status_code}) ?? status_code
      .status = row.status_message
```

Input (JSON)

```
{
  "host": "my.host.com",
  "timestamp": "2019-11-01T21:15:47+00:00",
  "status_code": 1,
}
```

Output (JSON)

```
{
  "host": "my.host.com",
  "timestamp": "2019-11-01T21:15:47+00:00",
  "status": "device status online",
}
```

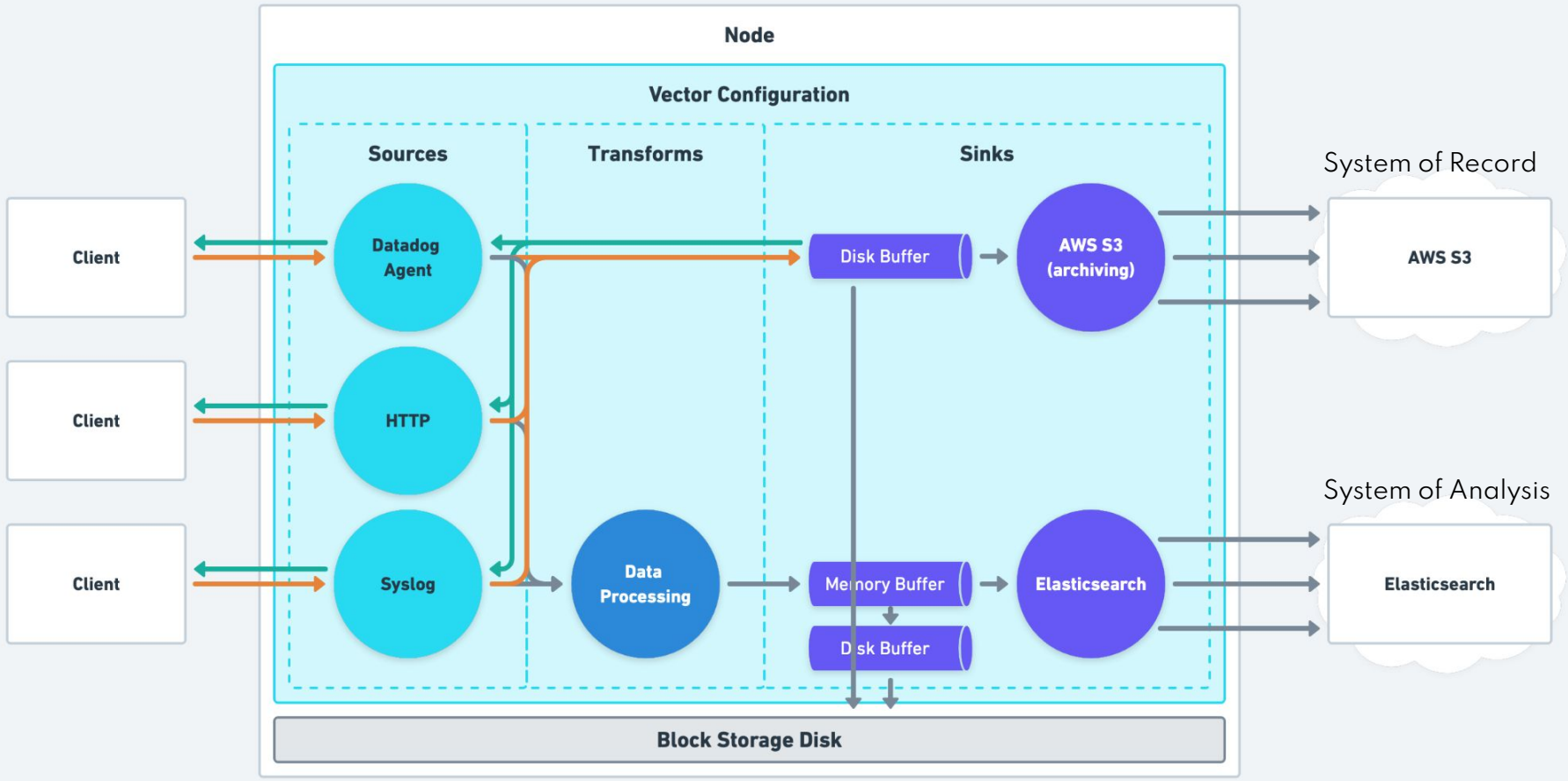
Einführung
Komponenten

- Senken

AMQP
AppSignal
AWS Cloudwatch logs, metrics
AWS Kinesis Data Firehose logs
AWS Kinesis Stream logs
AWS S3
AWS SNS
AWS SQS
Axiom
Azure Blob Storage
Azure Monitor Logs
GCP Chronical Unstructured
GCP Cloud Monitoring
GCP Cloud Storage
GCP Stackdriver
GCP PubSub
Blackhole
ClickHouse
Console
Databend
Datadog events, logs, metrics, traces
Elasticsearch
File
GreptimeDB
Honeycomb
HTTP
Humio logs, metrics
InfluxDB logs, metrics
Kafka
Loki
Mezmo (LogDNA)
MQTT
NATS
New Relic
Papertrail
Prometheus Exporter
Prometheus remote write
Pulsar
Redis
Sematext logs, metrics
Socket
Spunk HEC logs, metrics
StatsD
Vector
WebHDFS
Websocket

Einführung
Weiterleitung und Pufferung

Quelle:
https://vector.dev/docs/setup/going-to-prod/architecting/

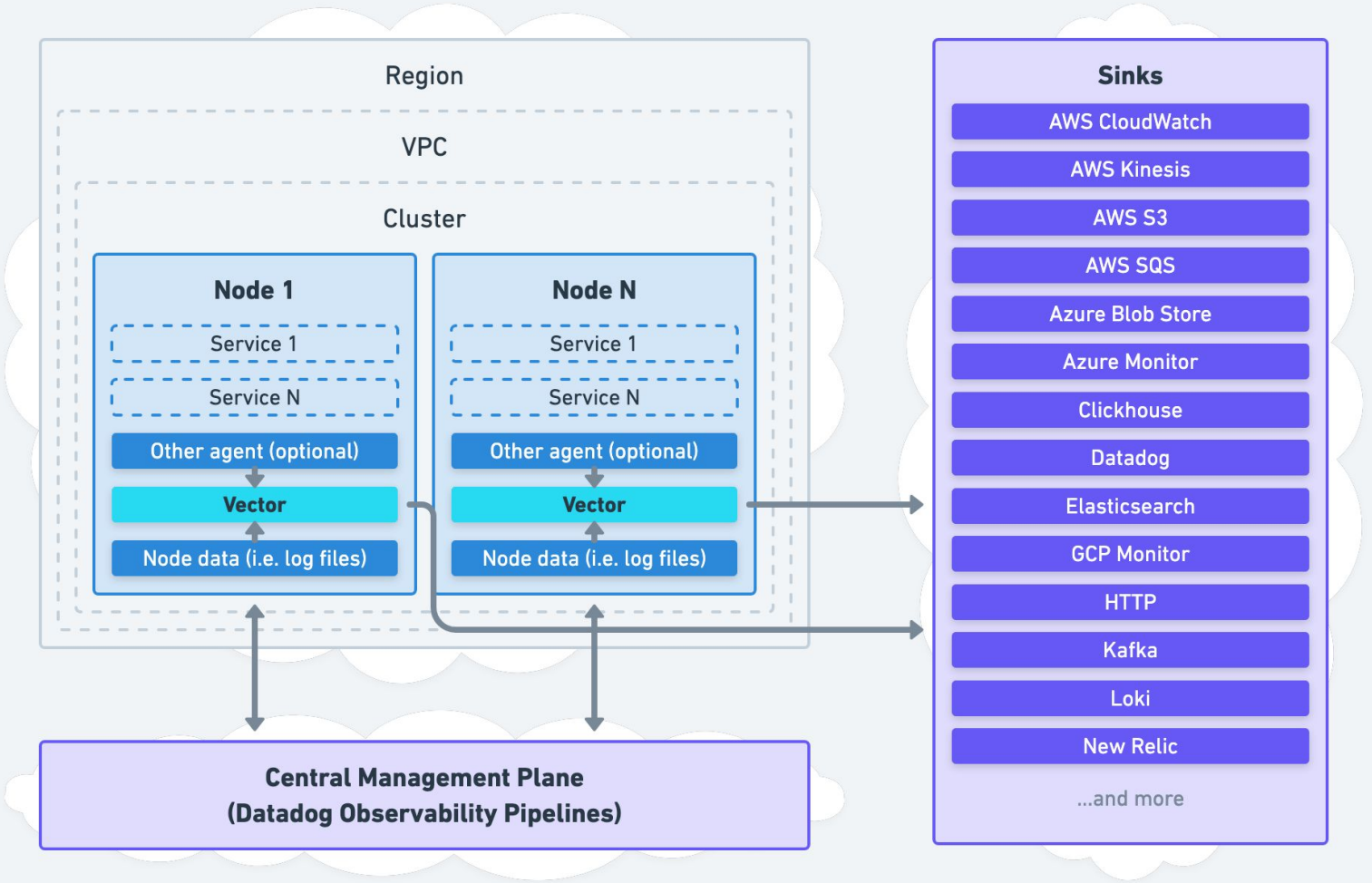


Deployment

Deployment
Referenz-Architekturen

Quelle: https://vector.dev/docs/setup/going-to-prod/arch/agent/

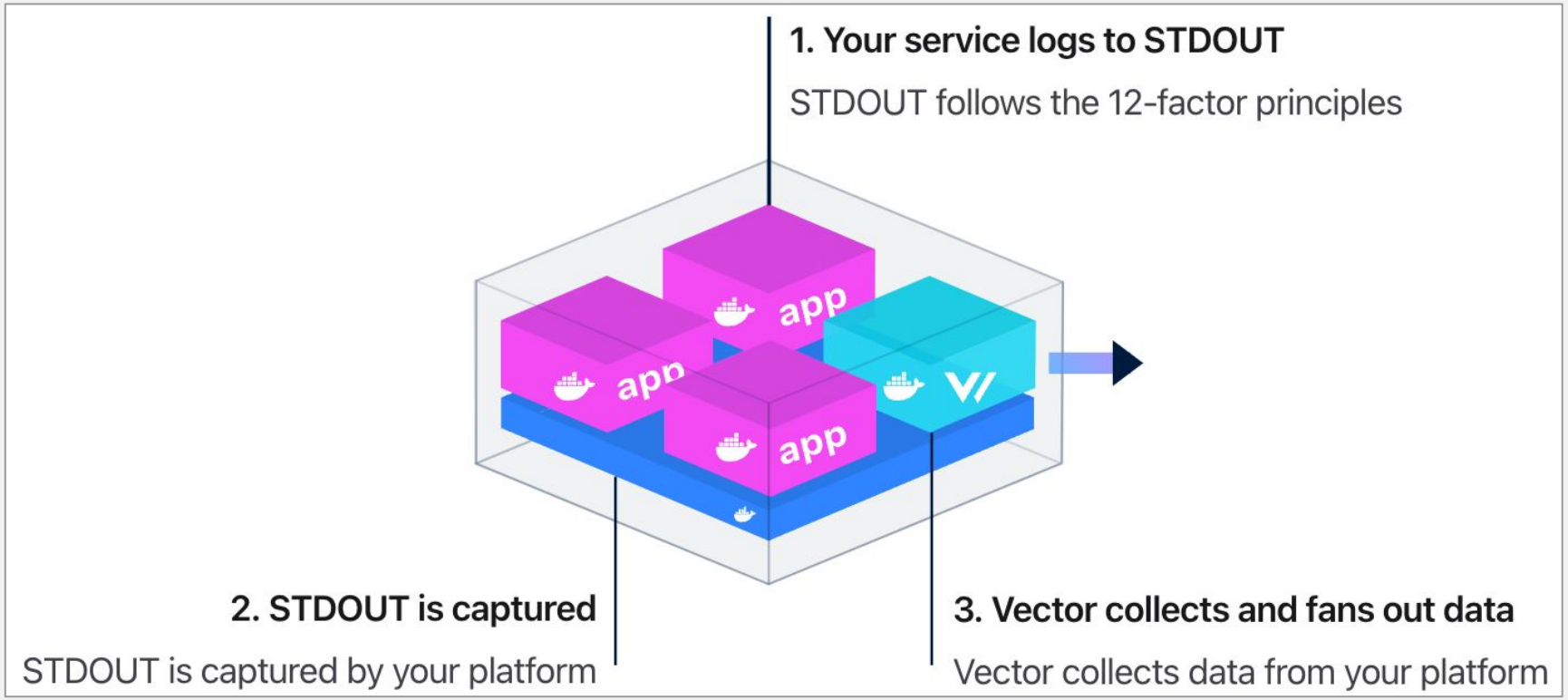
- Agent
- Aggregator
- Unified



Deployment
Referenz-Architekturen

- Agent
  - Daemon
  - Sidecar
- Aggregator
- Unified

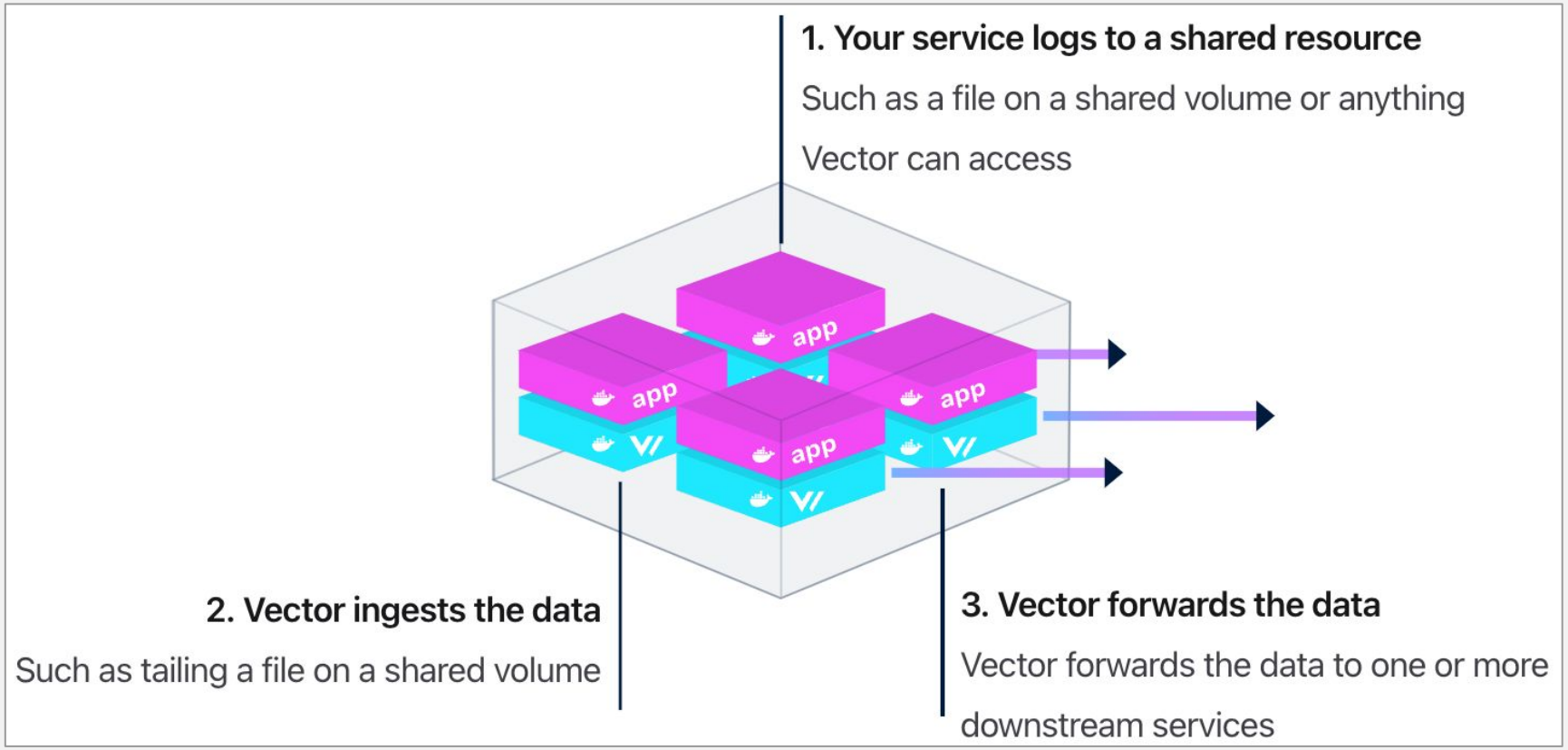
Quelle: https://vector.dev/docs/setup/deployment/roles/



Deployment
Referenz-Architekturen

- Agent
  - Daemon
  - Sidecar
- Aggregator
- Unified

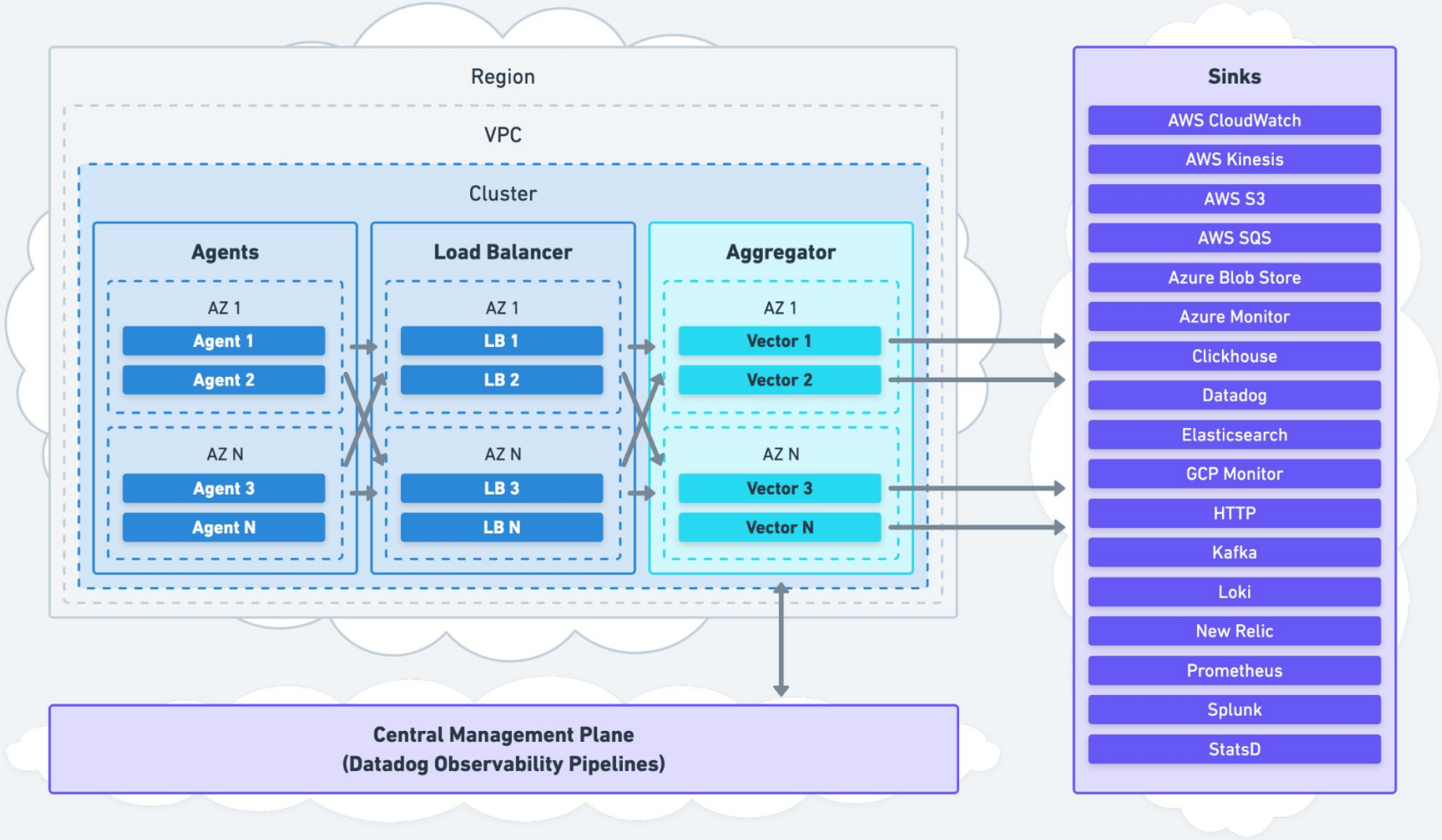
Quelle: https://vector.dev/docs/setup/deployment/roles/



Deployment
Referenz-Architekturen

Quelle: https://vector.dev/docs/setup/going-to-prod/arch/aggregator/

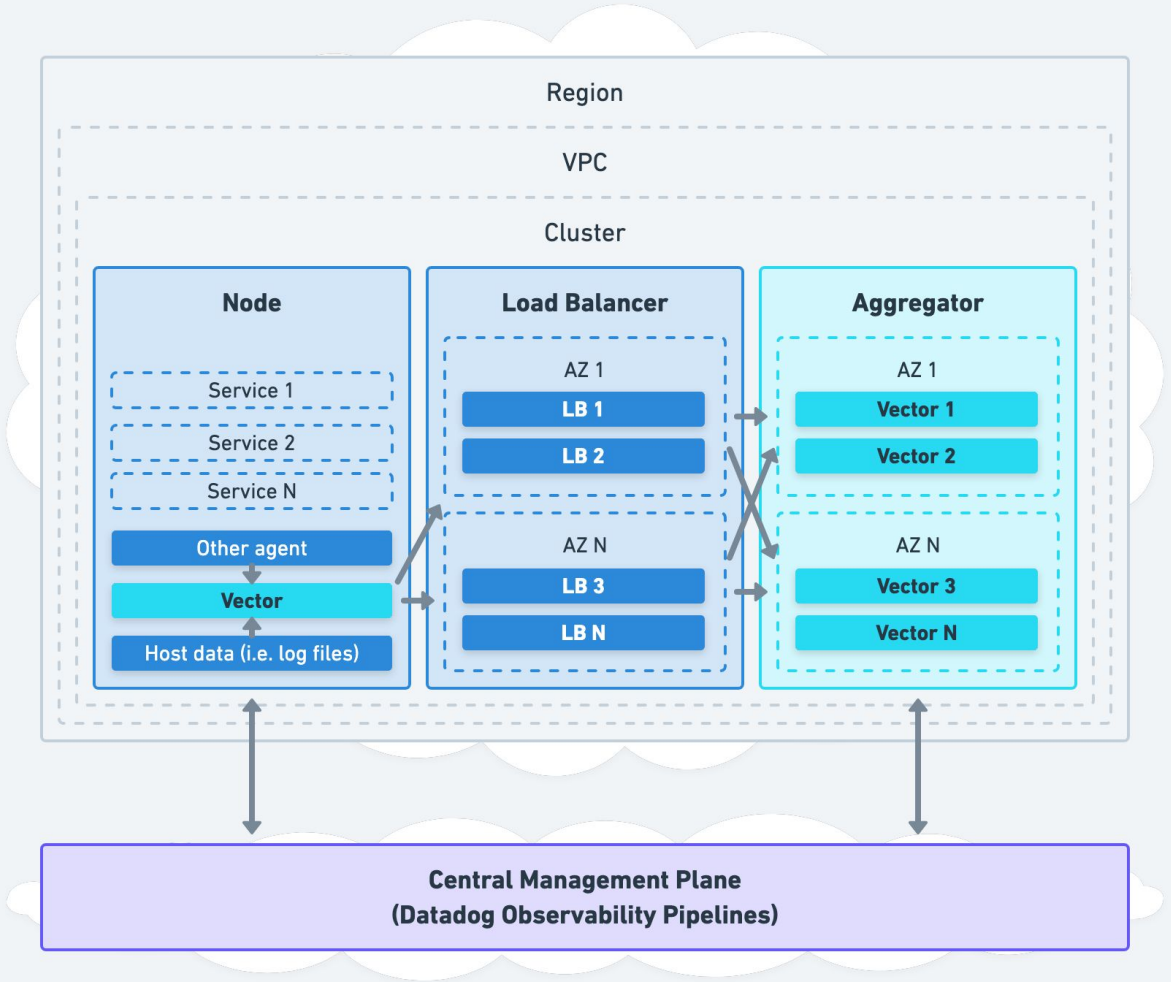
- Agent
- Aggregator
- Unified



Deployment
Referenz-Architekturen

- Agent
- Aggregator
- Unified

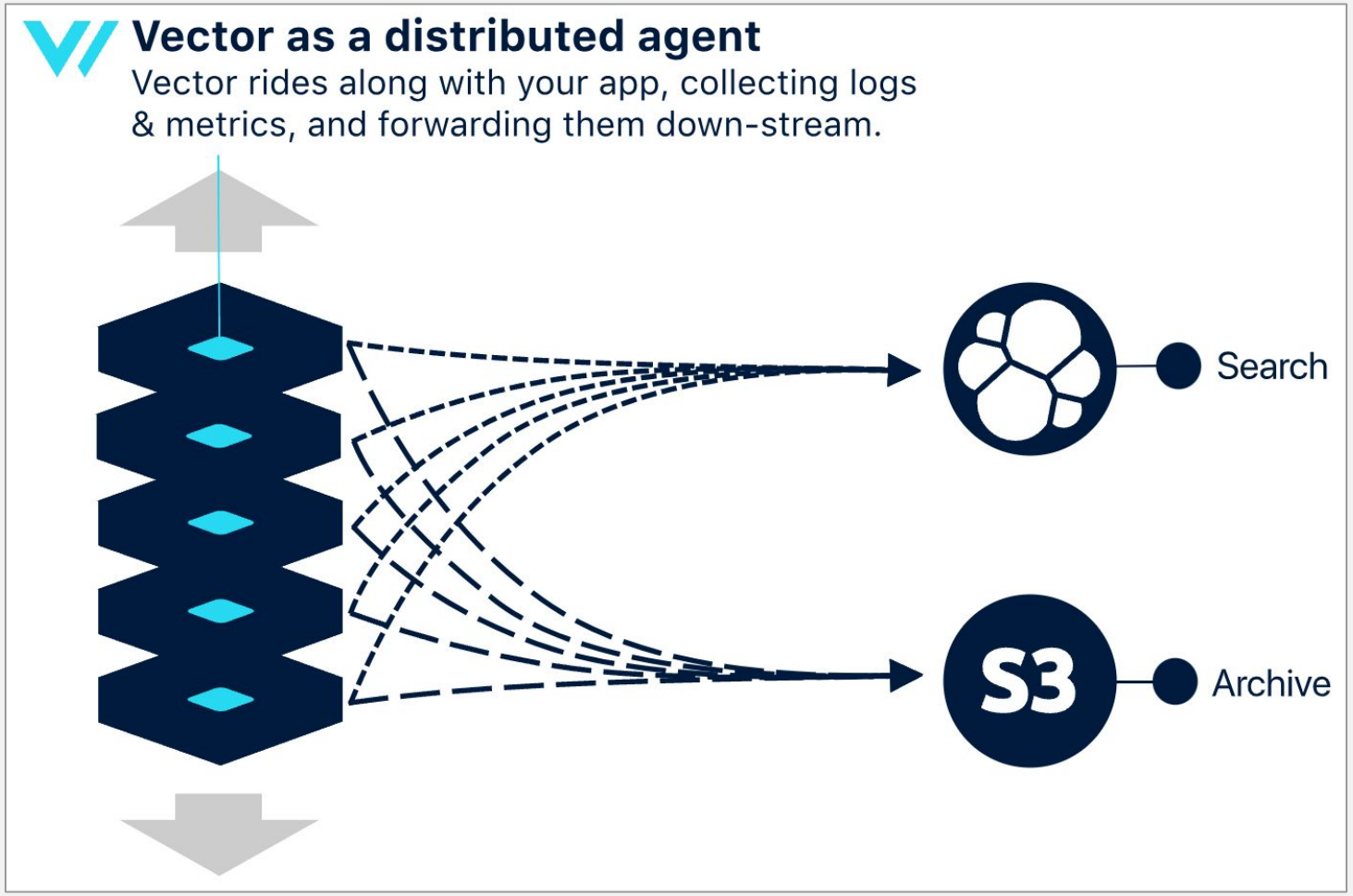
Quelle:
https://vector.dev/docs/setup/going-to-prod/arch/unified/



Deployment
Topologien

Quelle: https://vector.dev/docs/setup/deployment/topologies/

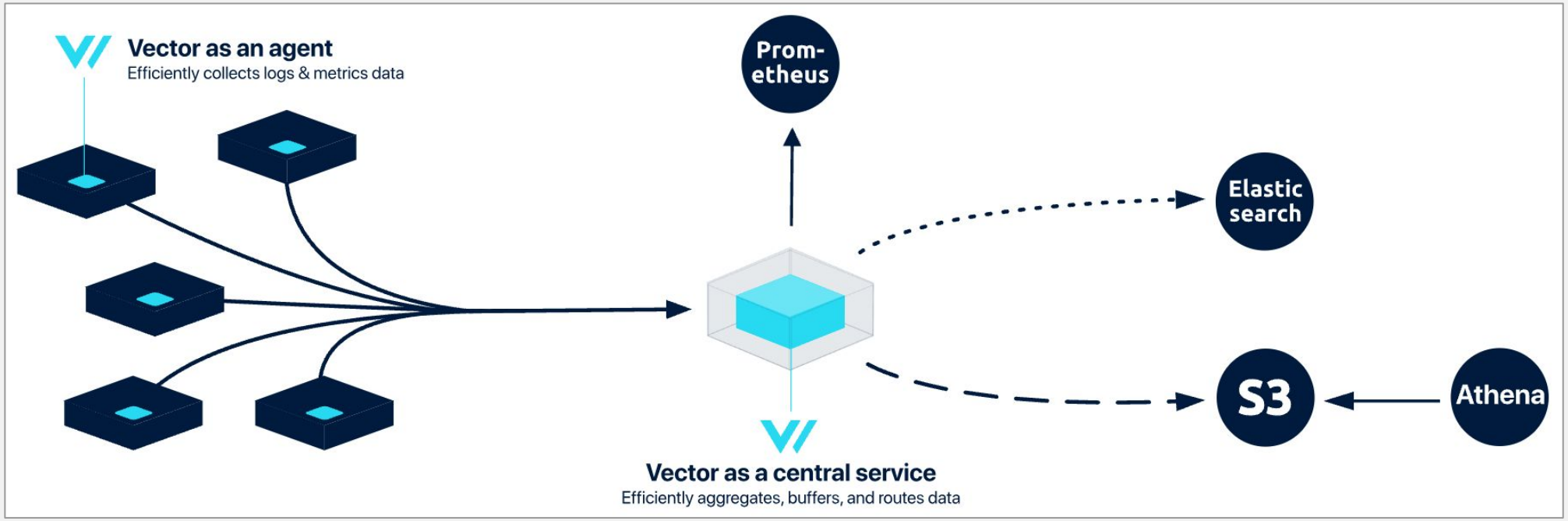
- Verteilt



Deployment
Topologien

- Zentralisiert

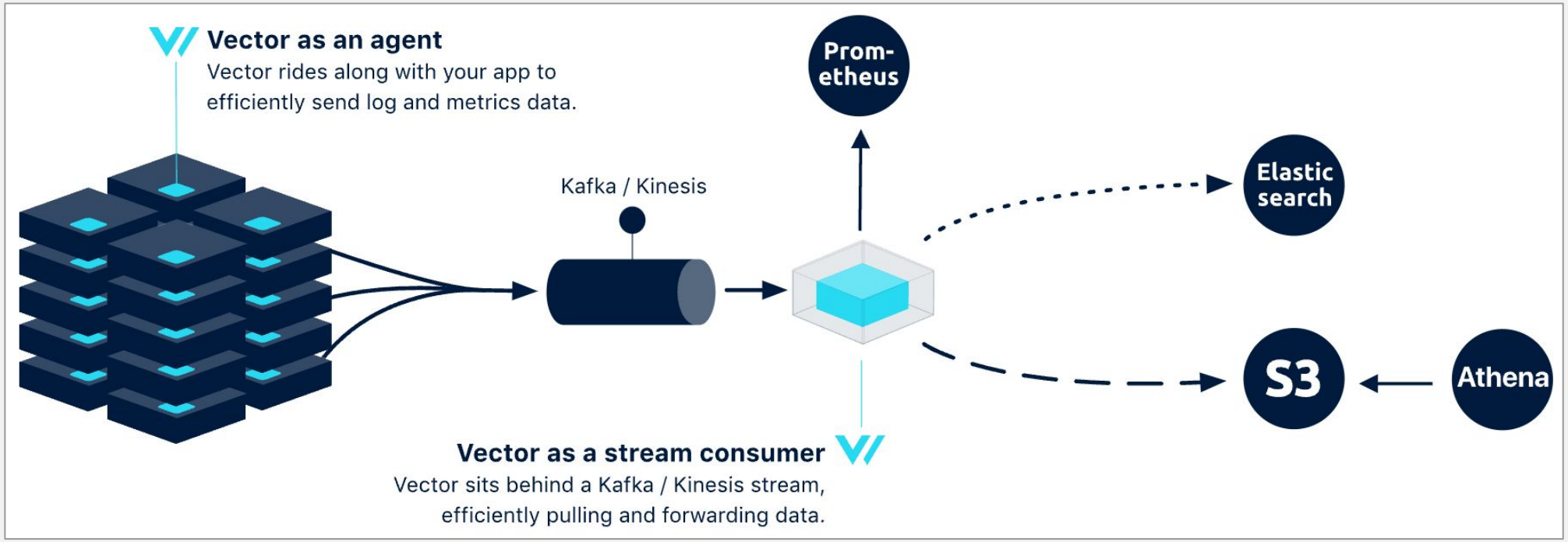
Quelle: https://vector.dev/docs/setup/deployment/topologies/



Deployment
Topologien

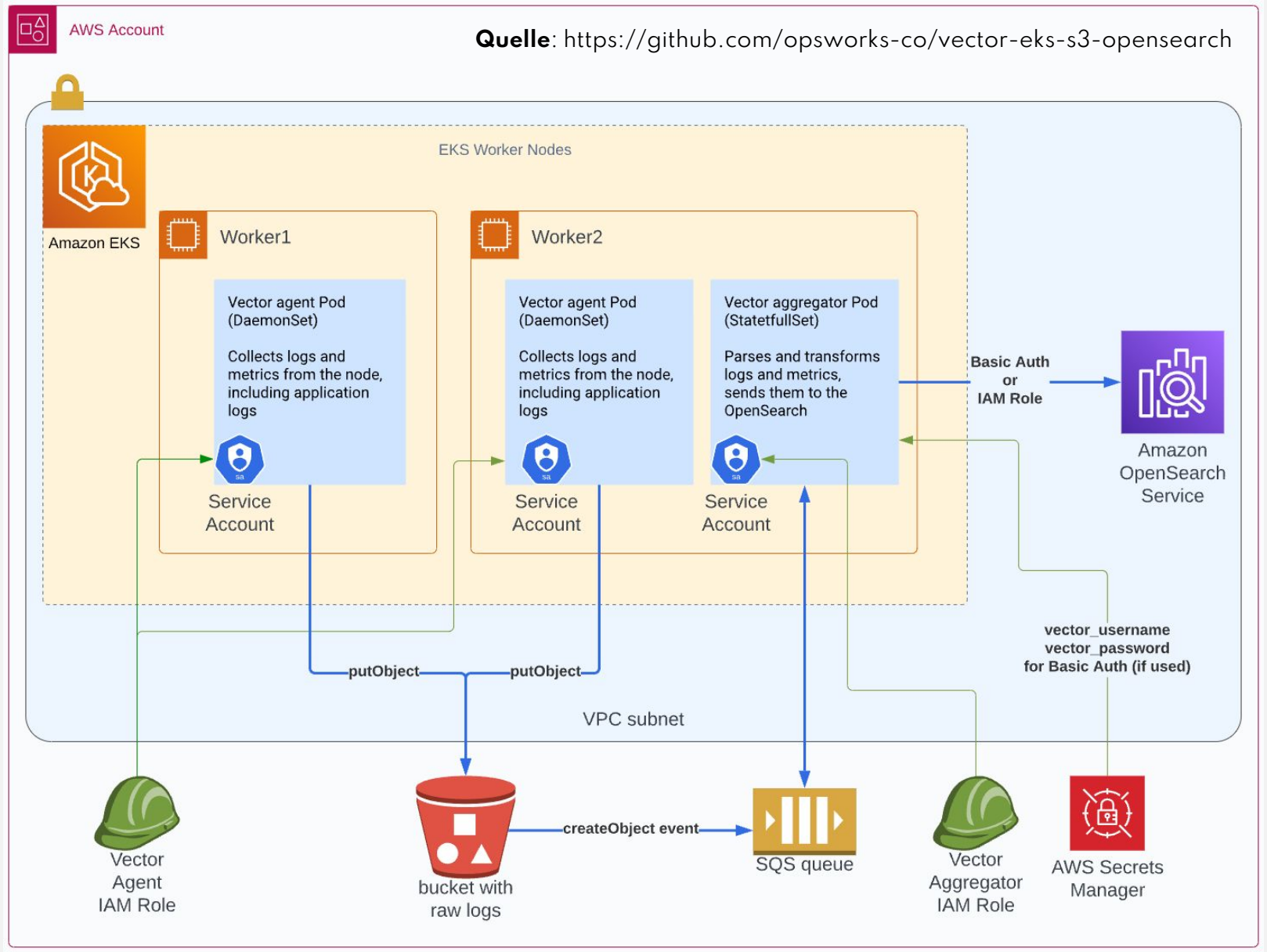
- Stream-basiert

Quelle: https://vector.dev/docs/setup/deployment/topologies/



Deployment Topologien

- Stream-basiert



QA|WARE

Demo

Disclaimer

No Kubernetes clusters were violated
during the preparation of this presentation

QA|WARE

Demo

Disclaimer

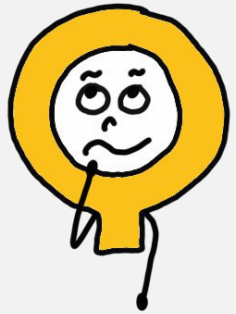
No Kubernetes clusters were violated
during the preparation of this presentation
… at least not badly

Limitierungen & Fazit

Limitierungen
Was fehlt oder könnte verbessert werden?

- Open Telemetry-Support kaum vorhanden

✅ Quelle für Logs	❌ Senke für Logs
❌ Quelle für Metriken	❌ Senke für Metriken
❌ Quelle für Traces	❌ Senke für Traces



- Traces bisher nur für Datadog
- Begrenzte Anzahl an Quellen und Senken
- Unklare Roadmap seit Übernahme durch Datadog
- Keine offiziellen Dashboards für interne Metriken (z.B. Grafana)

Fazit
Wann macht ein Einsatz Sinn?

- Anforderung an hohe Performance und Verfügbarkeit
- Schrittweise Migration eines Bestandsystems
- Teilweise Migration eines Bestandsystems (z.B. nur Aggregator)
- Nutzung von Push-basierten Senken (z.B. Prometheus)
- Anforderung an flexible Datenverarbeitung durch **VRL** oder **Lua**
- Anforderung an Zustellungsgarantien
- Anforderung an Vendor-Neutralität

Pro-Tipp:

Nutzt `vector dev <query>` für Suchanfragen.

qaware.de

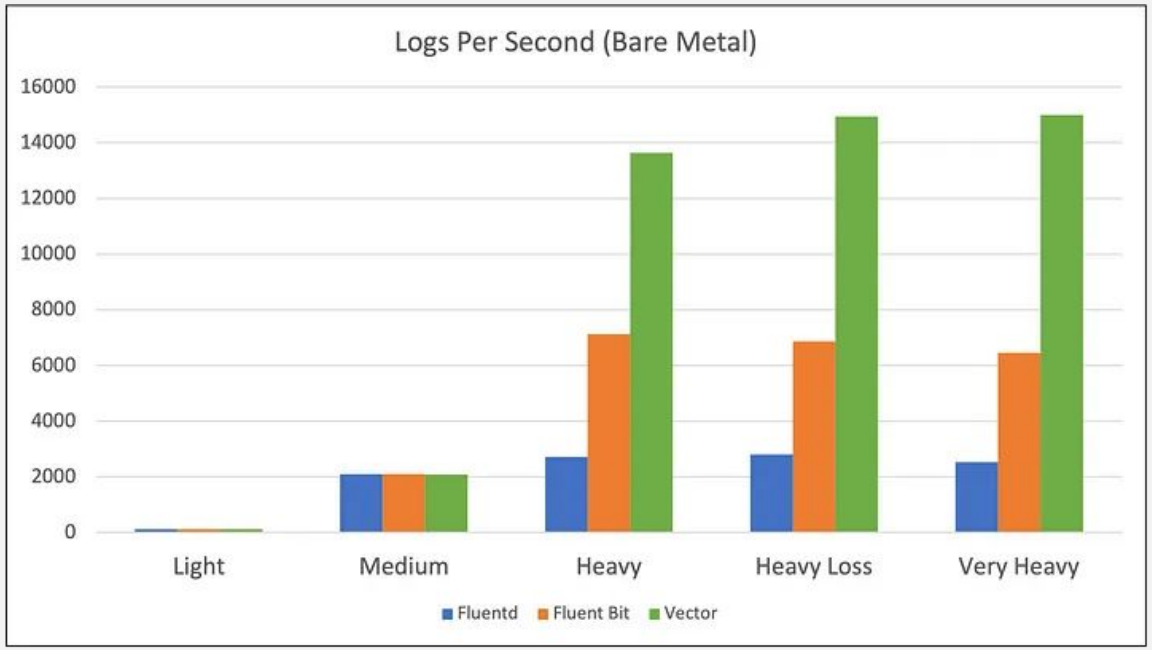
QA|WARE
SOFTWARE ENGINEERING

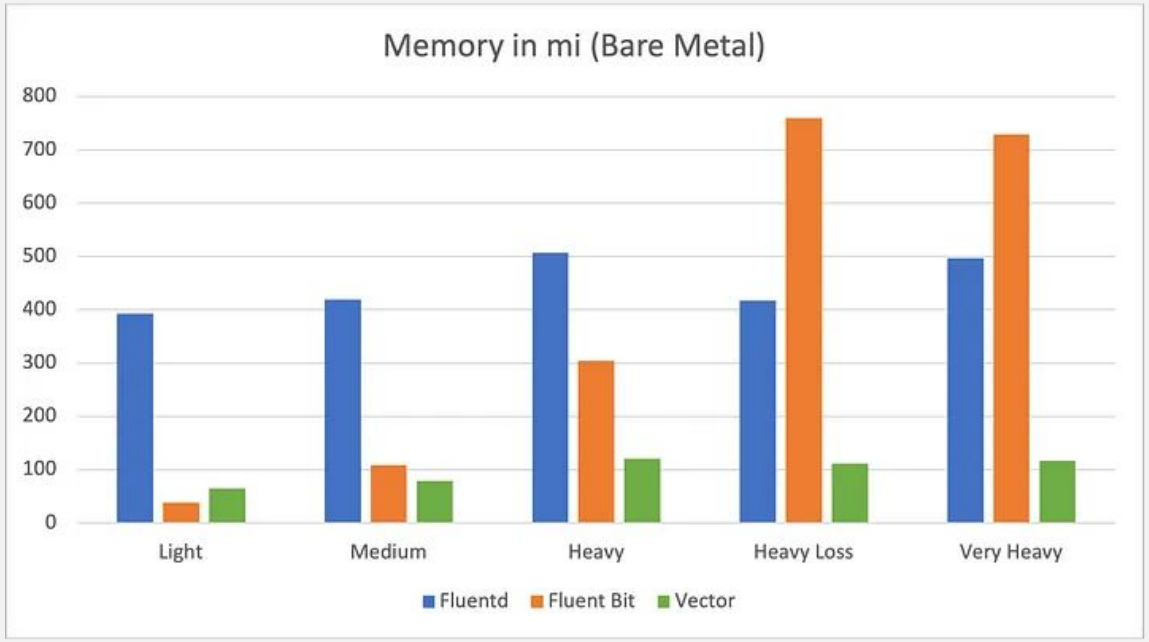
Q&A

QAware GmbH
Aschauer Straße 30
81549 München
Tel. +49 89 232315-0
info@qaware.de

linkedin.com/company/qaware-gmbh
xing.com/companies/qawaregmbh
slideshare.net/qaware
github.com/qaware

Backup
Performance-Vergleich

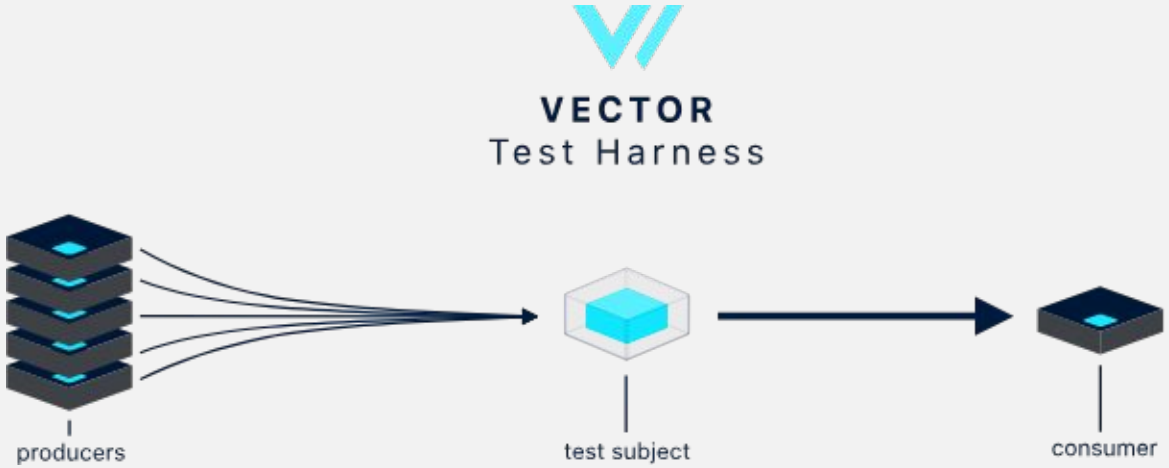




Quelle: https://medium.com/ibm-cloud/log-collectors-performance-benchmarking-8c5218a08fea

Backup
Performance-Tests

Quelle: https://github.com/vectordotdev/vector-test-harness/




```
$ bin/compare -t tcp_to_blackhole_performance

| Metric           | fluentbit | fluentd    | logstash   | vector     |
|:-----------------|:----------|:-----------|:-----------|:-----------|
| IO Thrpt (avg)   | 64.4MiB/s | 27.7MiB/s  | 40.6MiB/s  | 86MiB/s W  |
| CPU sys (max)    | 4         | 3.5 W      | 6.1        | 6.5        |
| CPU usr (max)    | 53.2      | 50.8 W     | 91.5       | 96.5       |
| Load 1m (avg)    | 0.5 W     | 0.8        | 1.8        | 1.7        |
| Mem used (max)   | 614.8MiB  | 294MiB     | 742.5MiB   | 181MiB W   |
| Disk read (sum)  | 9MiB      | 2.6MiB W   | 2.6MiB     | 2.6MiB     |
| Disk writ (sum)  | 14.8MiB   | 13.7MiB    | 11.6MiB    | 11MiB W    |
| Net recv (sum)   | 3.9gib    | 1.7gib     | 2.4gib     | 5.1gib W   |
| Net send (sum)   | 7.9MiB    | 5.7MiB     | 2.6MiB     | 9MiB       |
| TCP estab (avg)  | 663       | 664        | 665        | 664        |
| TCP sync (avg)   | 0         | 0          | 0          | 0          |
| TCP close (avg)  | 1         | 2          | 7          | 4          |

W = winner
fluentbit = 1.1.0
fluentd = 3.3.0-1
logstash = 7.0.1
vector = 0.2.0-6-g434bed8
```

Backup
Correctness-Tests

Test	Vector	Filebeat	FluentBit	FluentD	Logstash	Splunk UF	Splunk HF
Disk Buffer Persistence	✓	✓			⚠	✓	✓
File Rotate (create)	✓	✓	✓	✓	✓	✓	✓
File Rotate (copytruncate)	✓					✓	✓
File Truncation	✓	✓	✓	✓	✓	✓	✓
Process (SIGHUP)	✓				⚠	✓	✓
JSON (wrapped)	✓	✓	✓	✓	✓	✓	✓

Quelle: https://github.com/vectordotdev/vector-test-harness/

	Vector	Beats	Fluentbit	Fluentd	Logstash	Splunk UF	Splunk HF	Telegraf
End-to-end	✓							✓
Agent	✓	✓	✓			✓		✓
Aggregator	✓			✓	✓		✓	✓
Unified	✓							✓
Logs	✓	✓	✓	✓	✓	✓	✓	✓
Metrics	✓	⚠	⚠	⚠	⚠	⚠	⚠	✓
Traces	🚧							
Open	✓		✓	✓				✓
Open-source	✓	✓	✓	✓	✓			✓
Vendor-neutral	✓		✓	✓				✓
Reliability	✓							
Memory-safe	✓							✓
Delivery guarantees	✓					✓	✓	
Multi-core	✓	✓	✓	✓	✓	✓	✓	✓