as a solution for logs?

... seriously?

Jan Šimák

Self promo

currently at {showmax} ENGINEERING

- **devops team lead**

formerly at Seznam

- an architect to oversee an internal cloud project
- project manager
- someone called me a product guy as well

Elasticsearch?

do you know a product called Elasticsearch?

what tasks can it solve for you?

what is the ratio between data and indices on average?

what is hidden under the hood?

Question #1

How big is your ES cluster in terms of total number of nodes?

Question #2

How much do you ingest/index during peak hours?

Question #3

How much data do you keep “warm” (available for querying)?

Pros and Cons

+ can process/ingest almost everything

+ dynamic data type detection and index creation -

+ powerful DSL query language

+ easy to spin up and get started

- enormous demands for resources

- quite difficult to keep it effective

- quite challenging to keep it running on large deployments

- price

Price The Foremost

- compute resources (index/query)
  - one node can do **20k of messages per second on average**
  - querying cost is strictly use case specific
- storage resources
  - **add approx. 60% of original data for indices**
- licensing (when you want more)
  - not everything can be solved by a free/open source way
- **engineering time**

True story bro

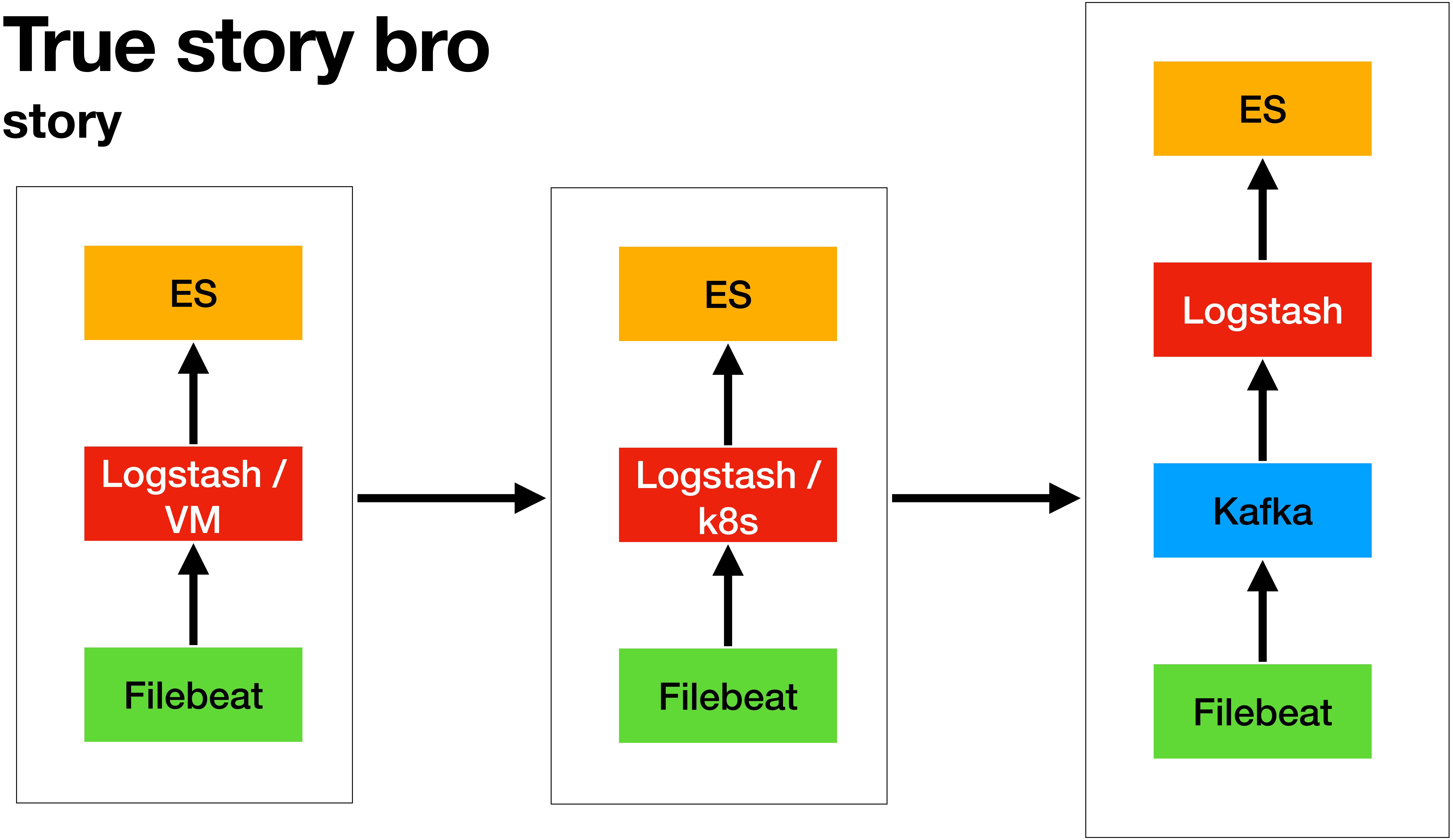
preface

project goals

- one solution for all “company” logs
- near real time process time
- no log structure
- ingest in hundreds of thousands per second (simply high enough)
- 100% availability
- 0% loss of data

True story bro

story



True story bro

touching the sky

ingest 400k/sec on average

200TB of data kept available for querying

availability of the solution quite high, but …

loss of data quite low, but …

data retention from days to weeks based on the input stream

TLS everywhere and **RBAC** implemented without spending a penny for licensing

True story bro

come down to earth

enormous number of shards per node

index field explosion

data type collisions

outdated version of ES

index alias != data protection

True story bro

takeaways

always follow best practices -> thoroughly read Documentation and take it seriously

understand fundamental metrics and alert on them

be strict to request log structure from the start

keep upgrading ES with high priority (e.g. log4shell)

limit on number of fields is your friend not an enemy

ES ain’t an Army Swiss Knife -> introduce more tiers using different technologies

be aware of compatibility matrix

How about an alternative?

AWS OpenSearch

an AWS fork of the official Elasticsearch

bunch of features for free bundled

(cross-cluster replication, multitenacy kibana, document/field level security, ...)

can be run on-prem or as a AWS service

keeps up with the official ES release plan

[WIP] How about an alternative?

Grafana Loki

an alternative solution for logs since 2019?

a completely different approach to indexing (data vs. metadata)

less expensive compared to an ES way

built to ingest logs and search them in mind

could be used as a basis of another tier

Thanks!