The broken promises and proven value of data lakes

Maurits van der Drift

30/1/2020

About myself

- Maurits van der Drift
- Born in 1991
- Study: Economics
- Career:
  - KPN 2015 - 2018
  - IBM 2018 – 2019
  - Schuberg Philis 2019 - current

- The data lake paradigm
- The pains of on-premise
- Arguments against the cloud
- Proven value of data

This presentation might contain:
opinions,
unfunded claims &
Azure bashing.

Data lake = Commercial terminology

- Even Gartner agrees!

Source: Gartner July 2019

The data lake paradigm

The enablers & promises

Enablers of data lakes

- Hadoop (oldskool)
  - Distributed file system & compute
  - Data interfaces
- Cloud (new school)
  - Object stores
  - Managed data processing services

Promises of data lakes

- Storage and compute are cheap
- New analyzing tools
- Unlocking of unstructured data
  - Sound, film, pictures, email, etc..
- Combining data will reveal new patterns
- Data + AI = Business value

And then came the obligations….

- Use data or be disrupted
- Become data driven
- Collect all data
- Save it forever
- Connect everything
- AI led processes
- Monetize your data

It might look and sound cool

But I have never seen it working

About unstructured and streaming data

- Unstructured means it yet has to be structured.
- Structuring data is hard and costly
- Streaming data quality is not 100%
- Ingesting real time data does not add anything if you use it for batch jobs
- Kafka will not thigh your shoelaces

The pains of on-premise

Ever tried upgrading HDP or CDH?

What do we see here?

On-premise

- No infrastructure as a code
- Limited compute
- Management is laborious
- Instable
- Nowadays data often comes from outside

Arguments against the cloud

It is new, scary & expensive

The cloud

- Not safe
- They will look into my data
- Expensive
- Lock in
- Moving data out of it is expensive

Proven value of data

So then, what does work?

When does it fly?

You have a specific, well-defined problem that:

- Touches your core business
- Can be solved by acquirable data
- The solution is the basis of your business model.
- Use a desired state tool like terraform

And how should you work?

Do not use the webportal/gui!

Use Github and a tool like Terraform

Stop trying to architect, let your developers experiment