First thoughts on an architectural set-up of open and distributed internet search

Prof. Dr. Andreas Henrich
University of Bamberg
Media Informatics
andreas.henrich@uni-bamberg.de
Outline

- Use Cases and Requirements
- Data Lakes as an Example
- Conclusions
Outline

- Use Cases and Requirements
- Data Lakes as an Example
- Conclusions
We need to connect our data spaces, computing spaces and web spaces... for setting up a joint **Open Web Data Pool and Web Index** in Europe.

**Access to the Index or in addition to the “Raw Data”?**

https://opensearchfoundation.org/
Basis for a variety of information services

- An **Open Web Data Pool and Web Index**, as a fundamental and indispensable basis for a large variety of public and private information services.

The application scenarios require “Structured Data”? 

https://opensearchfoundation.org/
Enterprise Data Lake Architecture: What to Consider When Designing
[Cloud Technology Partners, Sudi Bhattacharya, Neal Matthews
Enterprise Data Lake Architecture: What to Consider When Designing

[Cloud Technology Partners, Sudi Bhattacharya, Neal Matthews
Enterprise Data Lake Architecture: What to Consider When Designing

[Cloud Technology Partners, Sudi Bhattacharya, Neal Matthews
Outline

- Use Cases and Requirements
- Data Lakes as an Example
- Conclusions
Data Lake Template for Reference Architecture

https://www.dragon1.com/demo/data-lake
Key Benefits Of a Data Lake

1. Scalability
   - storage from *disparate sources* like multimedia, binary, XML; …

2. High-velocity Data
   - data stream processing and large volumes of *historical data*

3. Structure
   - unique arena where structure like *metadata*, speech tagging etc. can be applied on *varied datasets*

4. Storage
   - iterative and immediate *access* to the raw data

5. Schema
   - *schemaless write* and *schema-based read*

Source: Ajit Singh: *Architecture of Data Lake*, 2019, Data science Foundation,
https://datascience.foundation/sciencewhitepaper/architecture-of-data-lake
Architecture of a Data Lake

- Factors to consider:
  - Data Governance and Security Layer
  - Metadata Layer
  - Information Lifecycle Management Layer

- Tiers to manage data flows:
  - Intake Tier
  - Management Tier
  - Consumption Tier

- What is needed according to the CAP theorem?
  - Consistency
  - Availability
  - Partition tolerance

Data Reservoir Overview

1. Data In
2. Data Out
3. Data Refineries
4. Data Interfaces
5. Data Repositories
6. Data Access
7. Data Deposit
8. Data Integration & Governance

Outline

- Use Cases and Requirements
- Data Lakes as an Example
- Conclusions
Use a simple architecture to start with

- The Architecture of a Data-Lake-based search engine
Benefits of a clear architecture

- Standardized schemata
- Clear interfaces / APIs
- Well defined functional blocks

⇒ Will attract various players to contribute
⇒ Will allow for adaptation and specialisation in a generic frame
⇒ Will foster the Open Search Idea