Use XKCD API To Build A Searchable Database of XKCD Comics
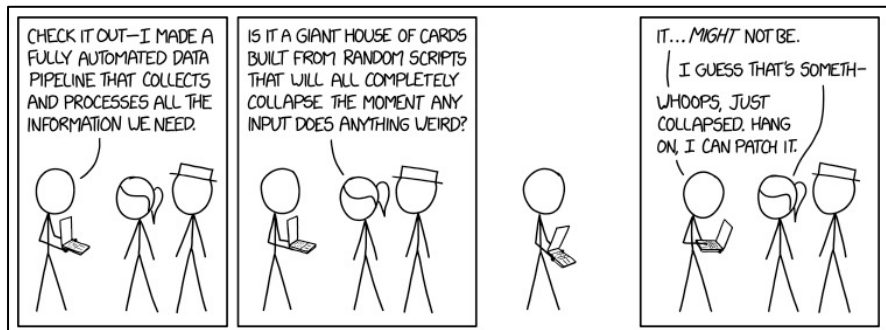
Practical Exam

Goal

XKCD provides regularly comics:

- https://xkcd.com/
- JSON API: https://xkcd.com/2054/info.0.json

https://xkcd.com/2054/

```
{
    "month": "10",
    "num": 2054,
    "link": "",
    "year": "2018",
    "news": "",
    "safe_title": "Data Pipeline",
    "transcript": "",
    "alt": "\"Is the pipeline literally running from your laptop?\" \"Don't be silly, my laptop disconnects far too often to host a service we rely on. It's running on my phone.\"", "img": "https://imgs.xkcd.com/comics/data_pipeline.png",
    "title": "Data Pipeline",
    "day": "3"
}
```
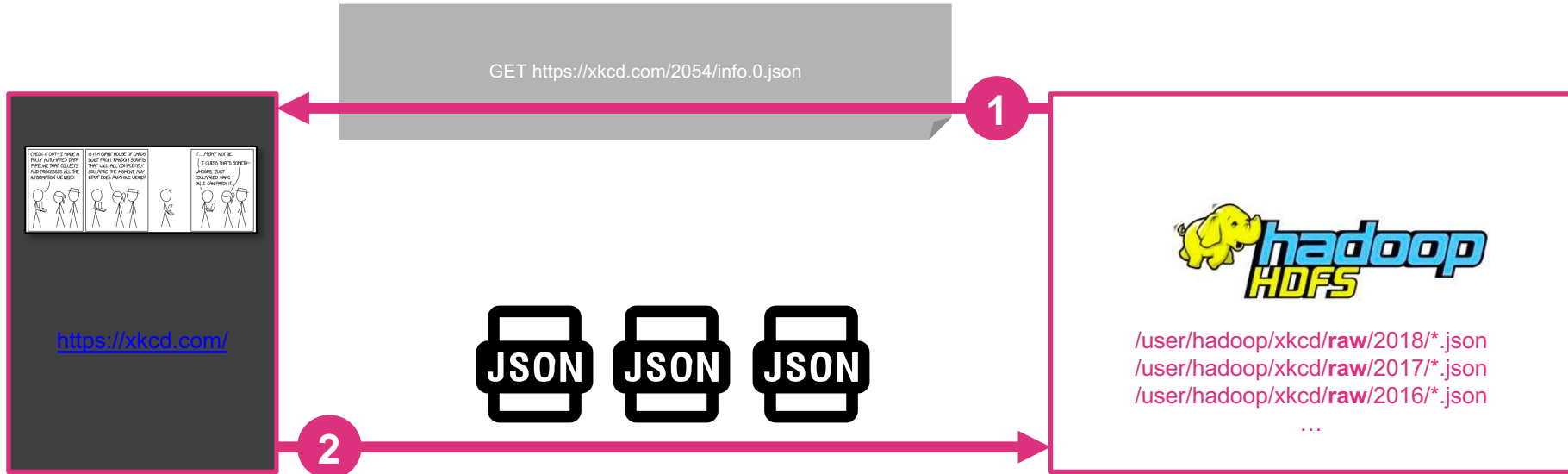
2054.json

www.marcel-mittelstaedt.com

Goal

We want to make use of this data to build a searchable database for XKCD comics.
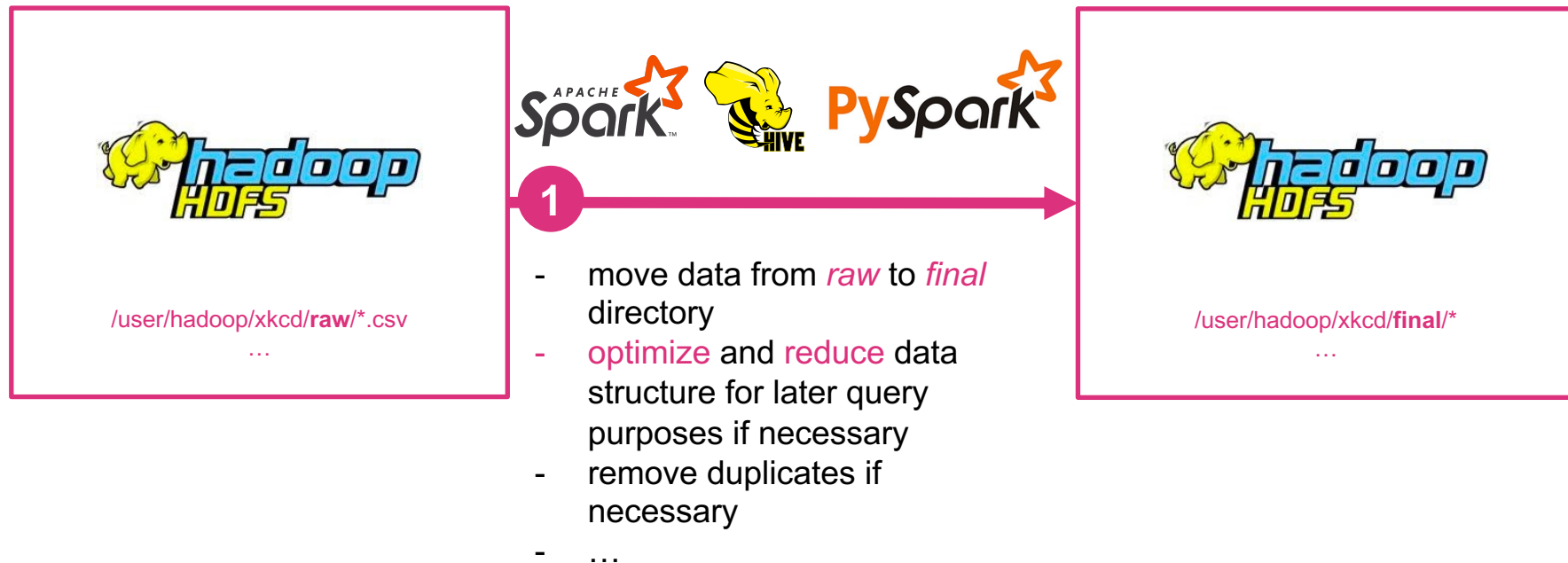
Workflow:

- **Gather** **data** from xkcd.com
- **Save** **raw data** (*JSON files*) to HDFS (partitioned by year, e.g. *2018*, *2017*, *2016*…)
- **Optimize**, **reduce** and **clean** **raw** **data** and save it to **final** directory on HDFS
- **Export** xkcd data **to end-user database** (e.g. MySQL, MongoDB…)
- Provide a simple **HTML Frontend** which is able to:
    - read from end-user database
    - process user input (search phrase…)
    - checks against xkcd data in end-user database
    - Display result (comics containing search phrase)
- The whole data workflow **must be implemented** within an ETL **workflow tool** (e.g. Pentaho Data Integration or Airflow) and **run automatically**

www.marcel-mittelstaedt.com

Dataflow: 2. Raw To Final Transfer

/user/hadoop/xkcd/**raw**/*.csv

…

1

- move data from *raw* to *final* directory
- optimize and reduce data structure for later query purposes if necessary
- remove duplicates if necessary
- …

/user/hadoop/xkcd/**final**/*

…

www.marcel-mittelstaedt.com

Dataflow: **3. Enhance Data And Save Results**
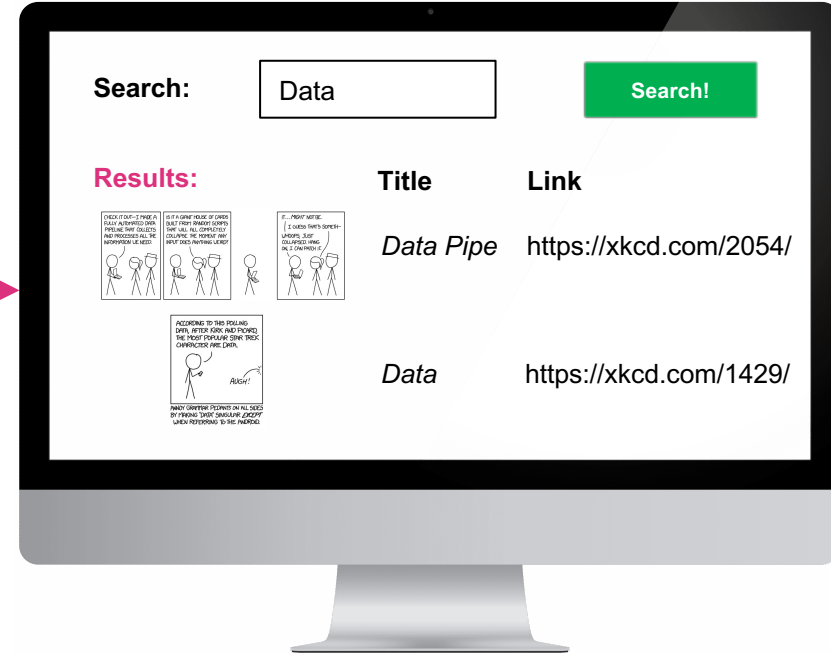

- enhance data (e.g. for later querying)
- use *Hive*, *Spark* or *PySpark*
- save everything to a end-user database (e.g. *MySQL*, *MongoDB*)

/user/hadoop/xkcd/**final**/*

…

www.marcel-mittelstaedt.com

Dataflow: **4. Provide Simple Web Interface**

- Provide a simple **HTML Frontend** which is able to:
    - read from end-user database
    - process user input (search phrase…)
    - checks against xkcd data in end-user database
    - Display result (comics containing search phrase)