

Big Data Projekt

Paul Garus | Benjamin Luhn | Johann Schäfer

Agenda

2

- Idee
- Gesamtarchitekturmodell
- Phasen
 - Preparation
 - Ingest
 - Staging
 - Processing
 - Access
 - Automation
 - Production
- Schlussbetrachtung

Idee

3

Aufnehmen von Stimmungen im NFL-Umfeld

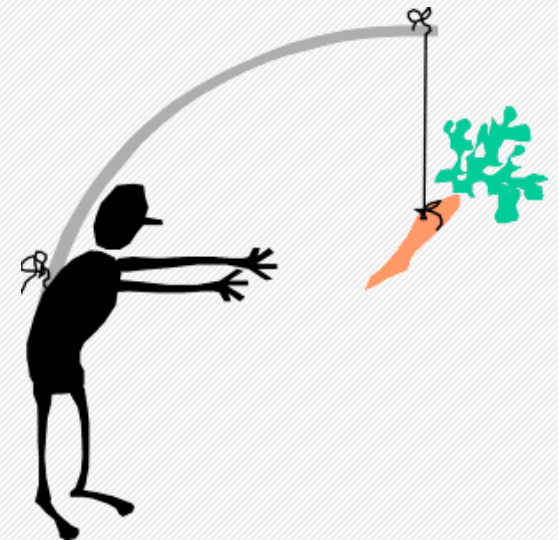
- *Reaktionen* im Internet auf Football erkennen
- Welche *Schlagwörter* werden im Rahmen von NFL häufig verwendet?
- Welches Team wird aktuell am häufigsten *genannt*?
- Welches Team ist aktuell *Favorit*?
- Aus welchen *Ländern* kommen diese Reaktionen?
- Wie ist das *Feedback* der Internetnutzer?
- Was bewegt die *NFL-Welt*?
- Wie ist die *Stimmung* in den Teams - welche Prognosen lassen sich daraus ableiten?



Motivation

5

- Starkes persönliches Interesse an American Football und der NFL
- Eigenes Ausüben von American Football in der Freizeit
- Erhöhtes Medieninteresse in Deutschland (NFL-Spiel in Deutschland?)
- Anstehende Playoffs und Superbowl
- Große Menge an Daten und großes Auswertungspotential



Gesamtarchitektur

6

Projektplanung

7

- Einarbeitung und Konzeption
- Datenaufnahme und initiale Ablage
- Datentransformation
- „Zieldatenmodell“ aufbauen
- Visualisierung + Schnittstelle entwickeln
- Automatisierung und Verfeinerung

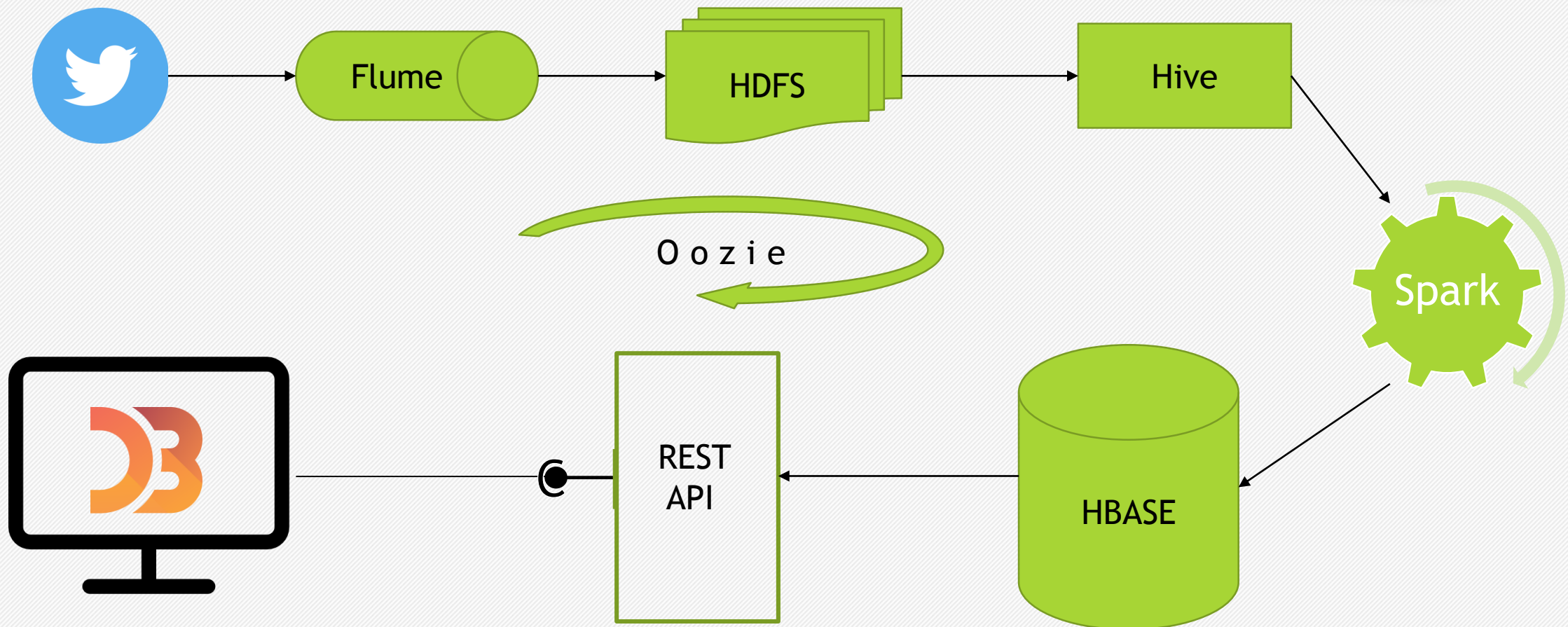
→ **Devise:** Erst eine einfache, aber vollständige Pipeline aufsetzen, die später erweitert werden kann



PROJEKTPLANUNG

Gesamtüberblick

8

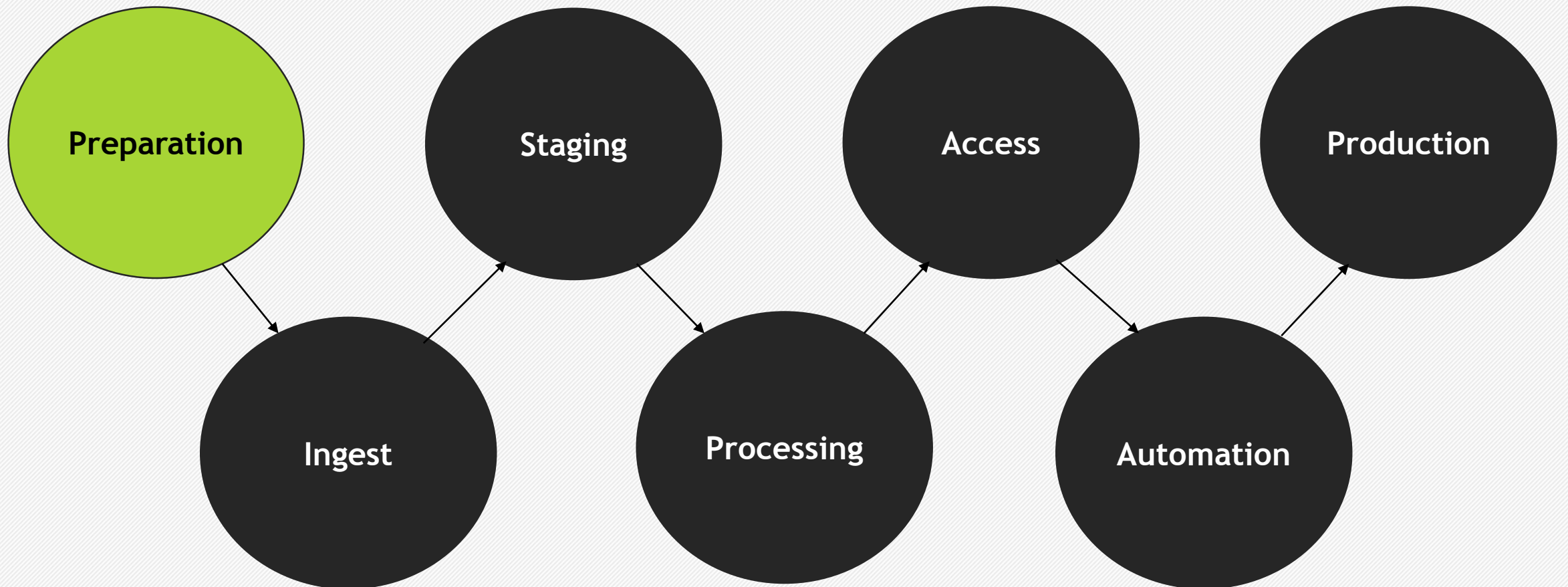


Phasen

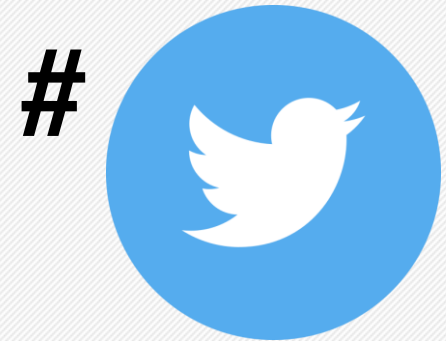
9

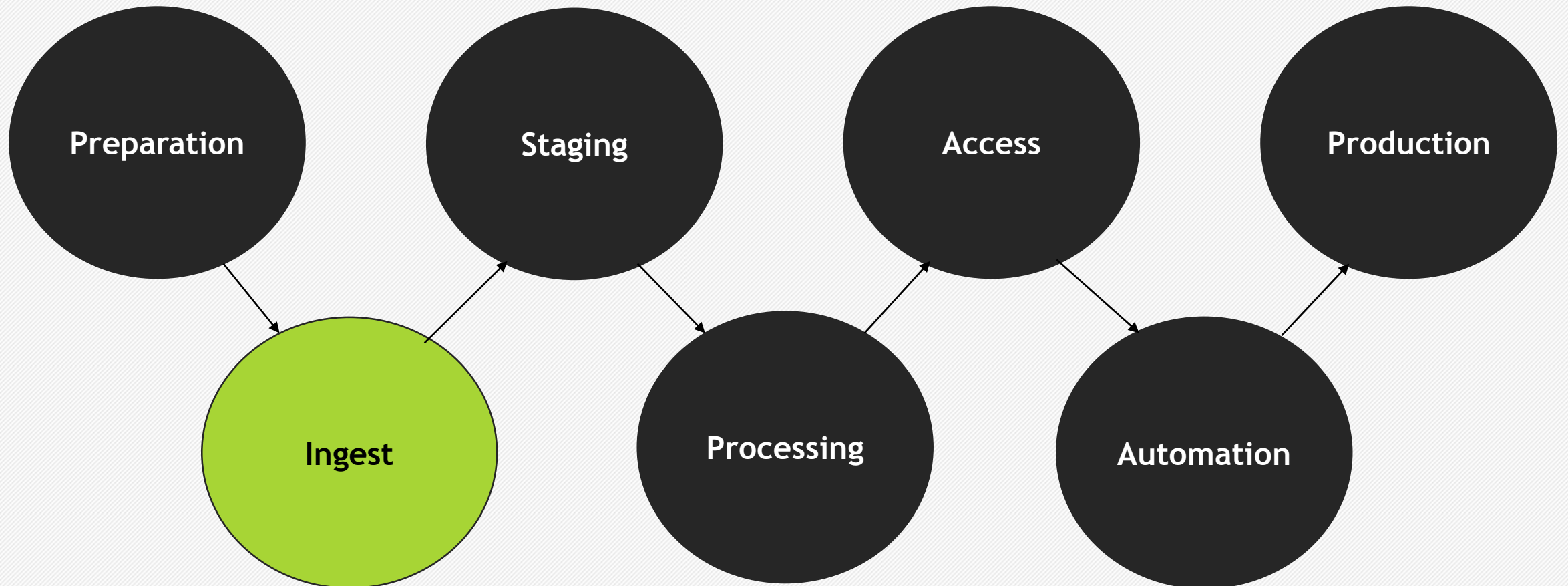
Phasen

10



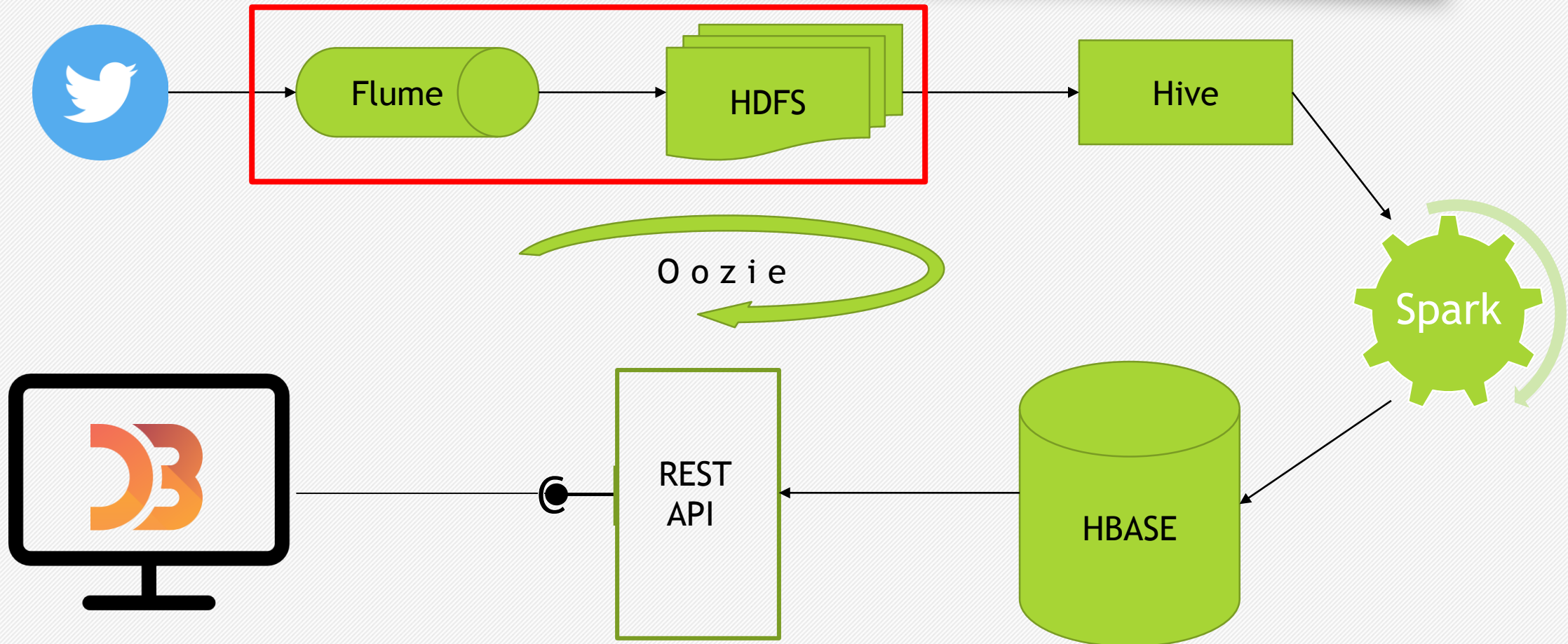
- Datenquelle: Twitter als Kurznachrichtendienst
- Bietet die Möglichkeit:
 - Schnelle Reaktionen
 - Gedanken und Emotionen der User
 - Thematische Sortierung der Inhalte (# Hashtags)
 - Reaktionen aus aller Welt
 - Ereignisse und Reaktionen zeitlich beieinander (Super Bowl)





Gesamtüberblick

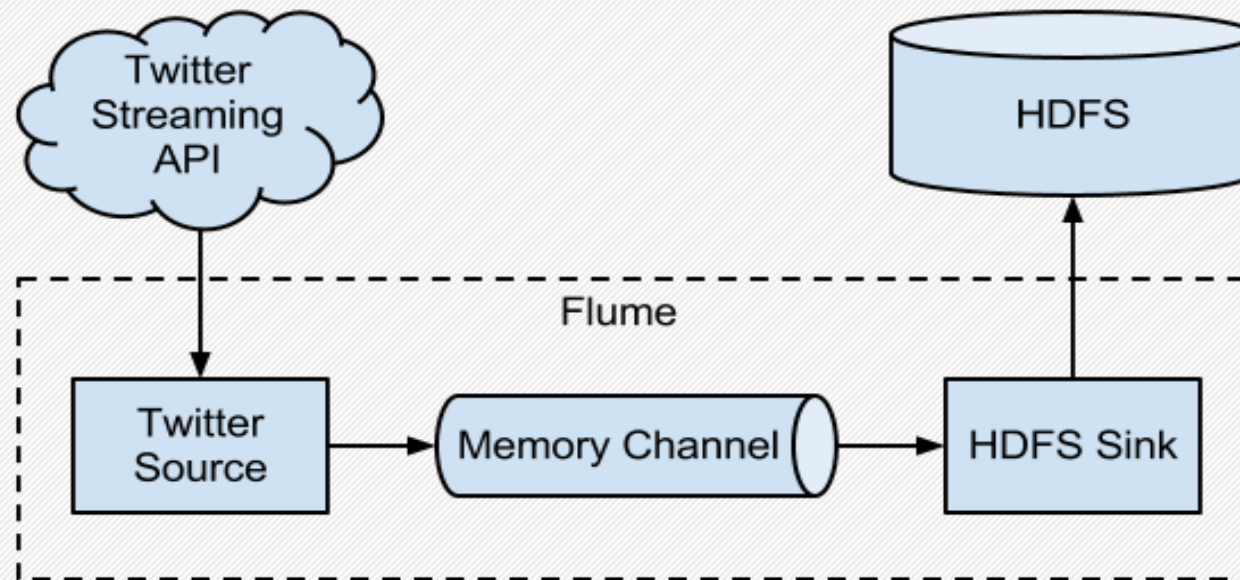
13



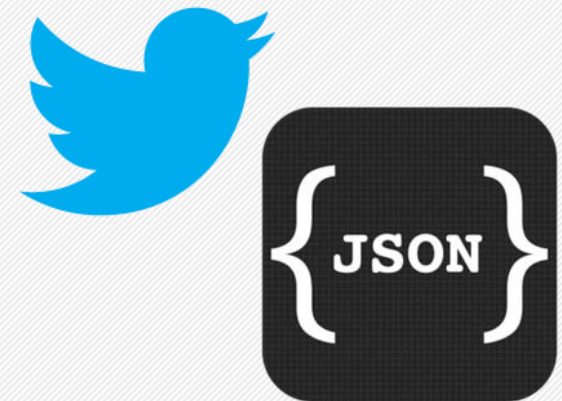
Ingest

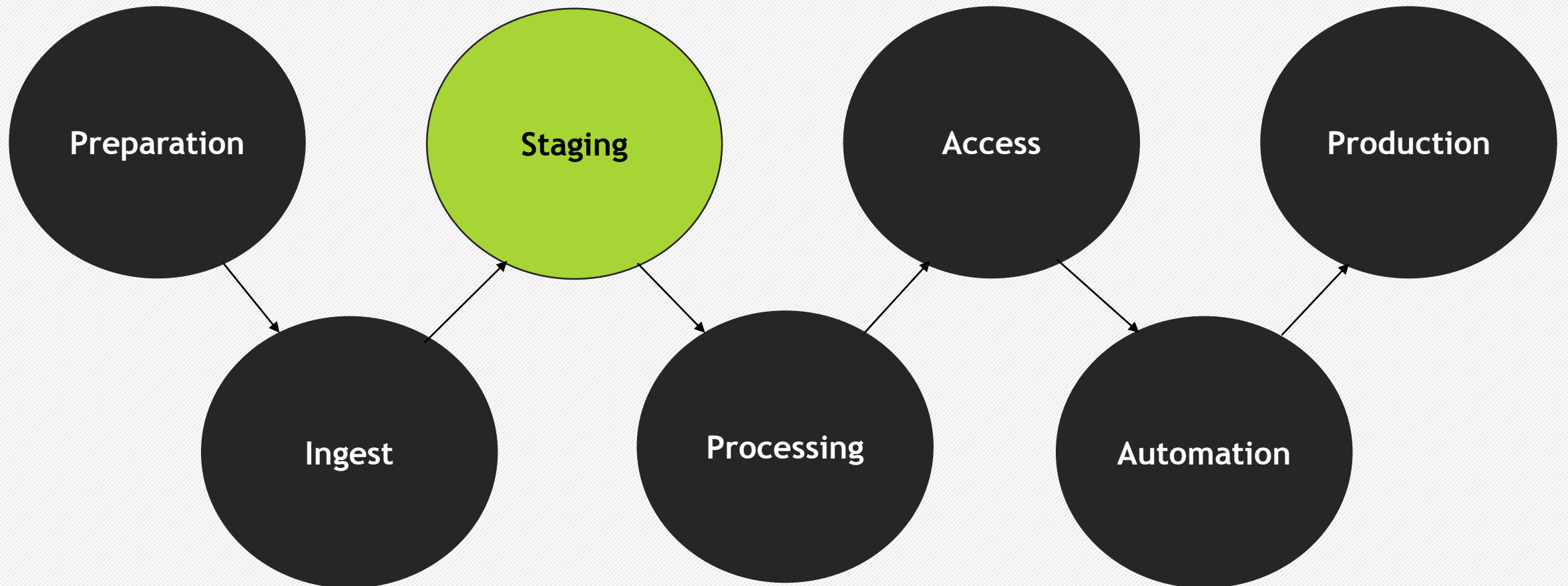
14

- Twitter Streaming API



- JSON-Format
- Durchschnittlich 4 Tweets pro JSON
- 200 Keys pro Tweet
- Keys:
 - „created_at“: „Sun Jan 08 18:06:47 +0000 2017“,
 - „place“: null
 - „source“: „<a href=\http...\“
 - „text“: „Rodgers for MVP!“
 - „timestamp“: „1253412234“
 - „hashtag“: „nfl,superbowl“





Staging

17

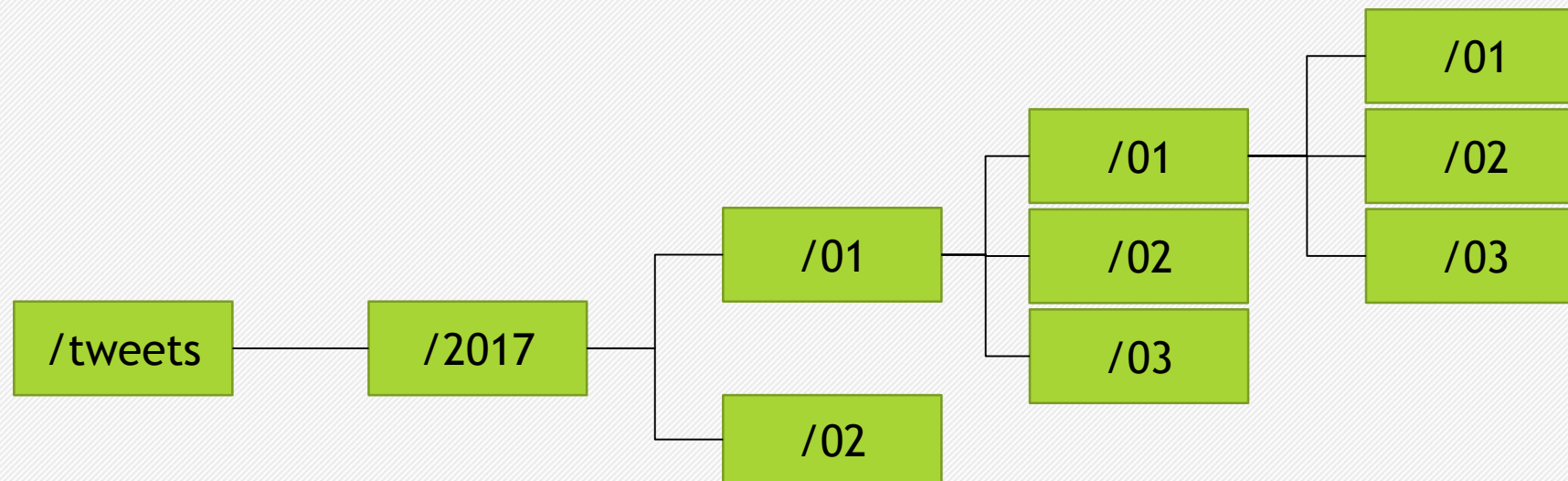
Root

/Year

/Month













/Day

/Hour



Staging

18

Startseite / user / cloudera / tweets / 2017 / 02 / 05 						Verlauf	Trash
<input type="checkbox"/>	Name	Size	User	Gruppe	Berechtigungen	Date	
	 ↑		flume	cloudera	drwxrwxrwt	February 07, 2017 08:35 PM	
<input type="checkbox"/>	 .		flume	cloudera	drwxrwxrwt	February 05, 2017 03:16 PM	
<input type="checkbox"/>	 00		flume	cloudera	drwxrwxrwt	February 04, 2017 05:17 PM	
<input type="checkbox"/>	 01		flume	cloudera	drwxrwxrwt	February 04, 2017 06:37 PM	
<input type="checkbox"/>	 02		flume	cloudera	drwxrwxrwt	February 04, 2017 07:09 PM	
<input type="checkbox"/>	 03		flume	cloudera	drwxrwxrwt	February 04, 2017 09:24 PM	
<input type="checkbox"/>	 05		flume	cloudera	drwxrwxrwt	February 05, 2017 04:02 AM	
<input type="checkbox"/>	 12		flume	cloudera	drwxrwxrwt	February 05, 2017 06:42 AM	
<input type="checkbox"/>	 14		flume	cloudera	drwxrwxrwt	February 05, 2017 08:08 AM	
<input type="checkbox"/>	 16		flume	cloudera	drwxrwxrwt	February 05, 2017 09:08 AM	
<input type="checkbox"/>	 17		flume	cloudera	drwxrwxrwt	February 05, 2017 10:05 AM	

Staging - Archivierung

19

- Durch Ablagestruktur explizite Archivierung nicht zwingend notwendig
- Nutzung der Hadoop Archivierung
- Zielbild: Integration in Oozie Workflow

Beispiel:

„hadoop archive -archiveName 20170208.har -p /user/cloudera
tweets/2017/02/08/* tweets/2017/02/08/“

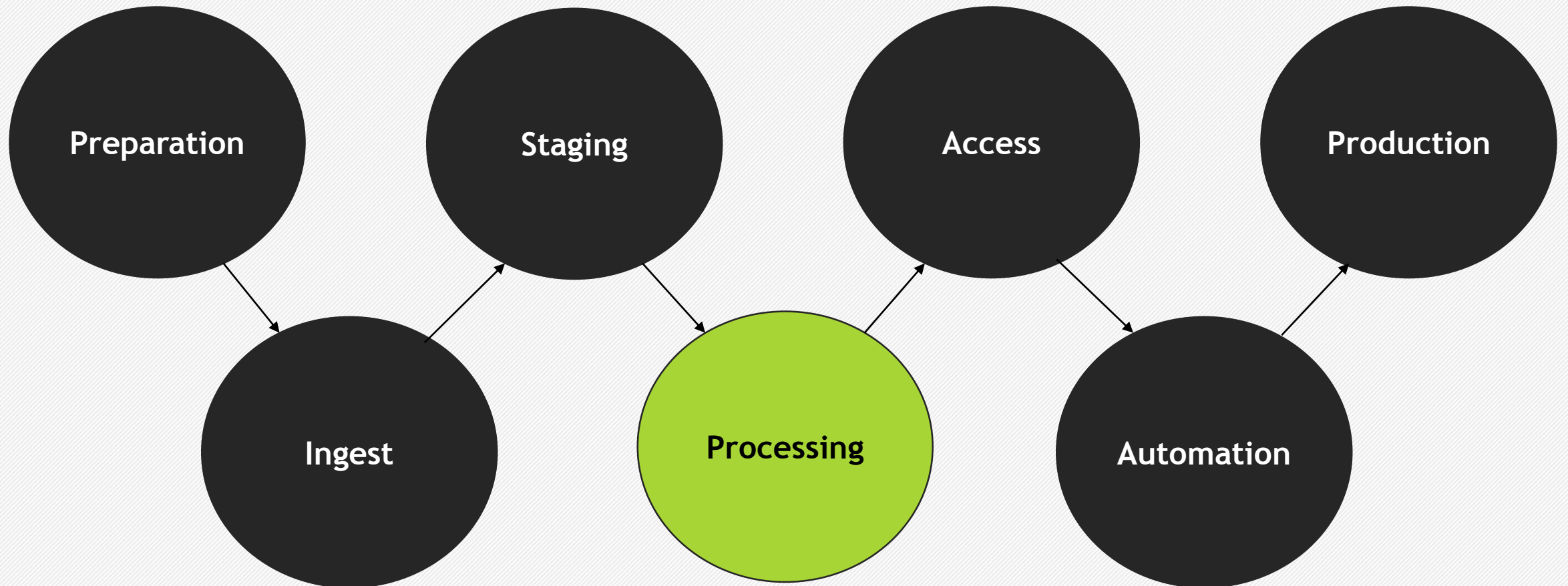


„Konzeption für Mandantenfähigkeit und Zugriffsbeschränkung“

- Nur notwendige Rechte
- Technischer Schnitt: Flume, Spark, HDFS, HBASE (=> Technical Users)
- Querschnittlicher Schnitt: Rest-Services, Spark-Jobs (=> Groups)

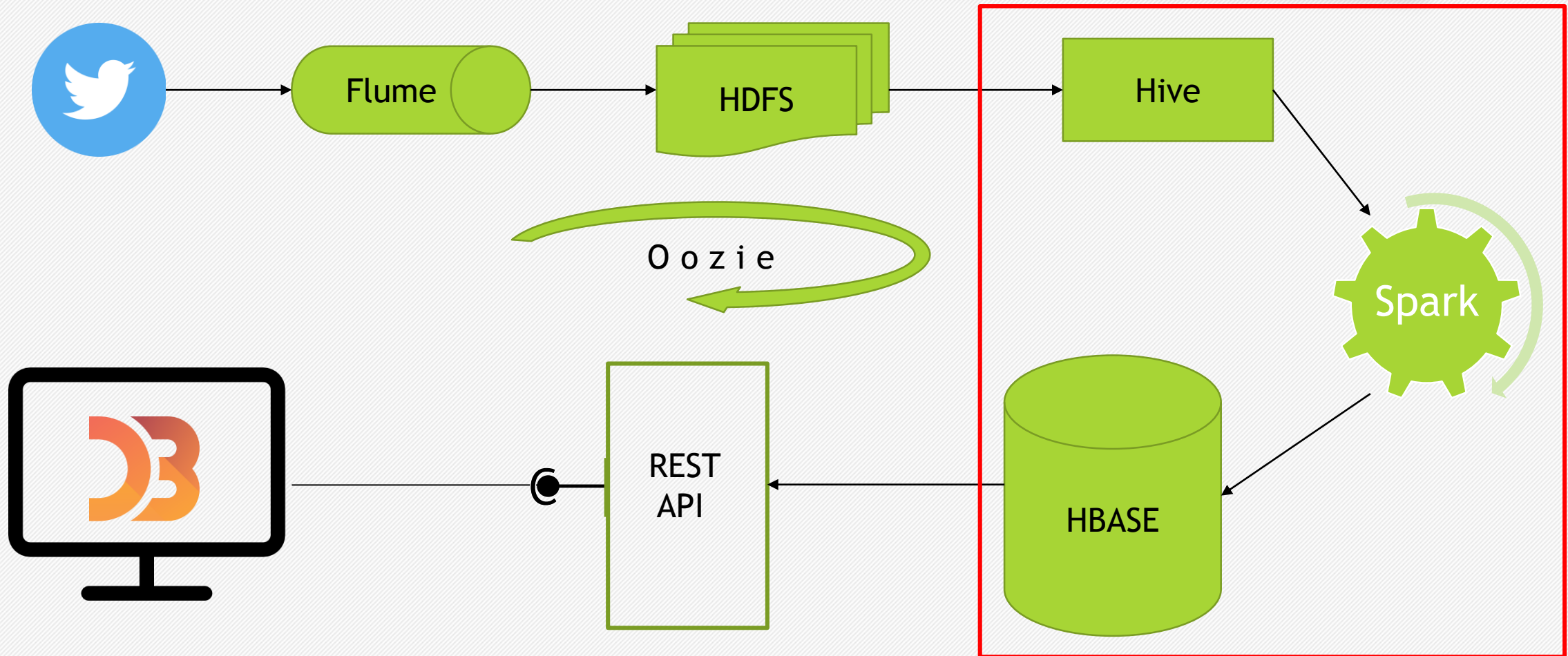
Basis: Rechtesystem von Hadoop





Gesamtüberblick

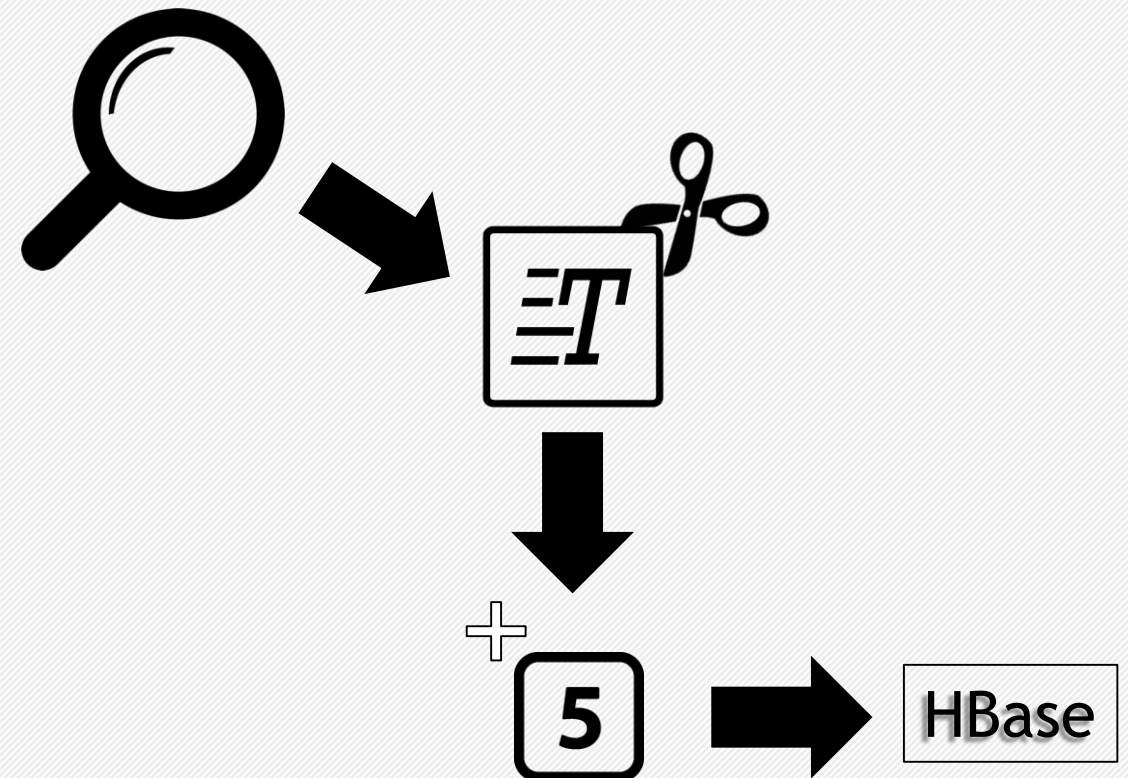
22



- Hive-Tabelle
 - CREATE EXTERNAL TABLE tweets (.....)
 - Eigenes .Jar zum parsen des JSONs
- Partitionierung der gesamten Daten zur besseren Verarbeitung
 - PARTITIONED BY (datehour INT)
 - Bessere Zuordnung zu bestimmten Ereignissen bzw. Spielen (Super Bowl)



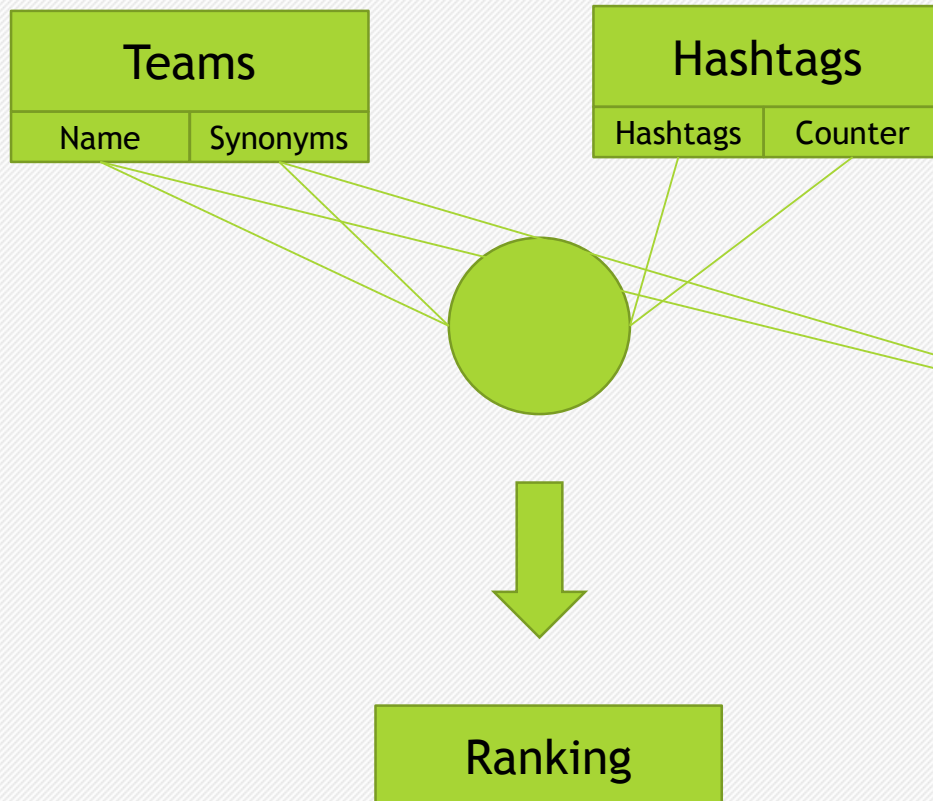
- Jobaufbau:
 - Select-Abfrage nach 'Hashtags'
 - Konvertieren in Datentyp String
 - Trim-Funktion zum Schneiden des Strings
 - Counter zum Zählen der Tweets
 - Counter und Tweet in HBase speichern



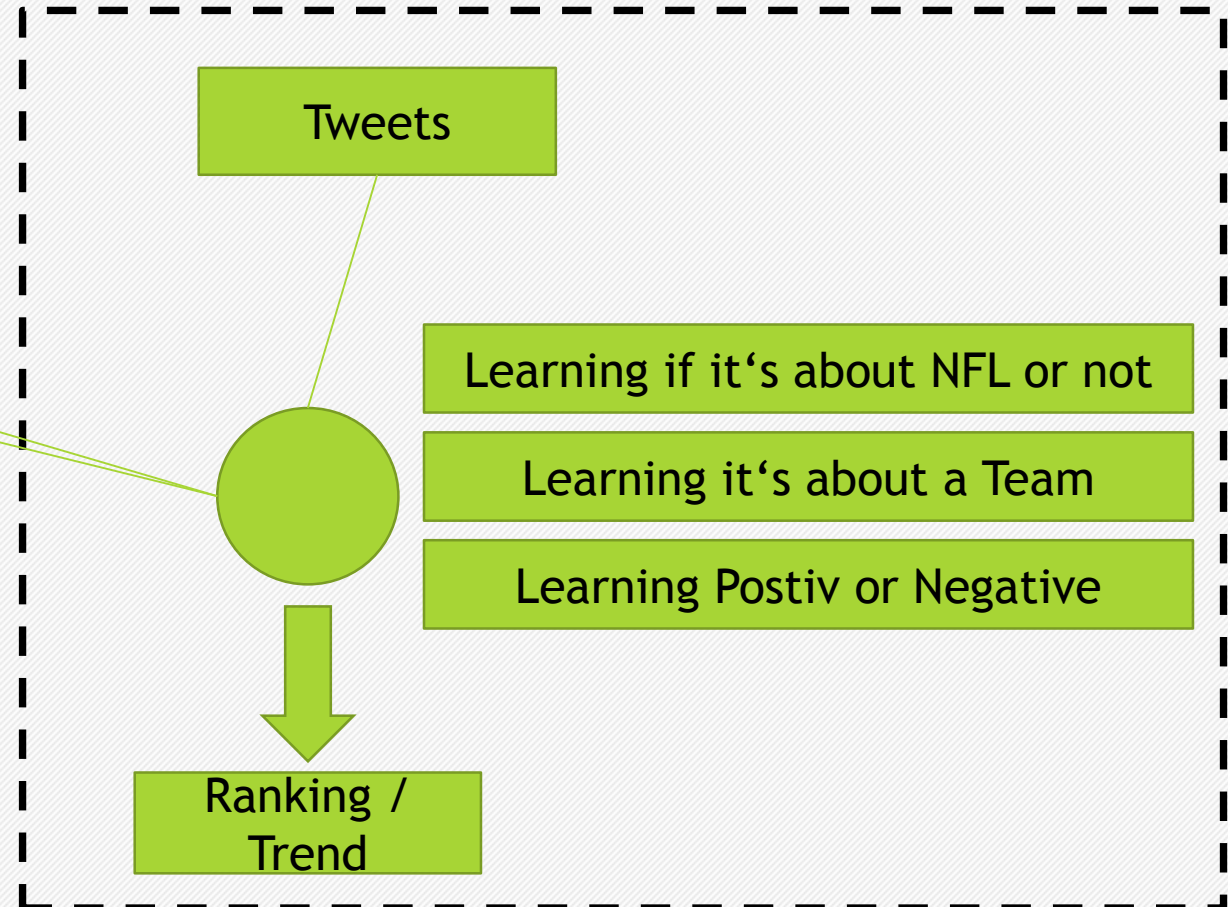
Spark - Zielbild

25

Ranking bilden



Zielbild Machine Learning



- Ablage der transformierten Daten von Spark
- „Zieldatenmodell“
- Import von relationalen Metadaten (Teamdaten)

Teams	
Name	Synonyms

Hashtags	
Hashtags	Counter



Processing - HBase Archivierung

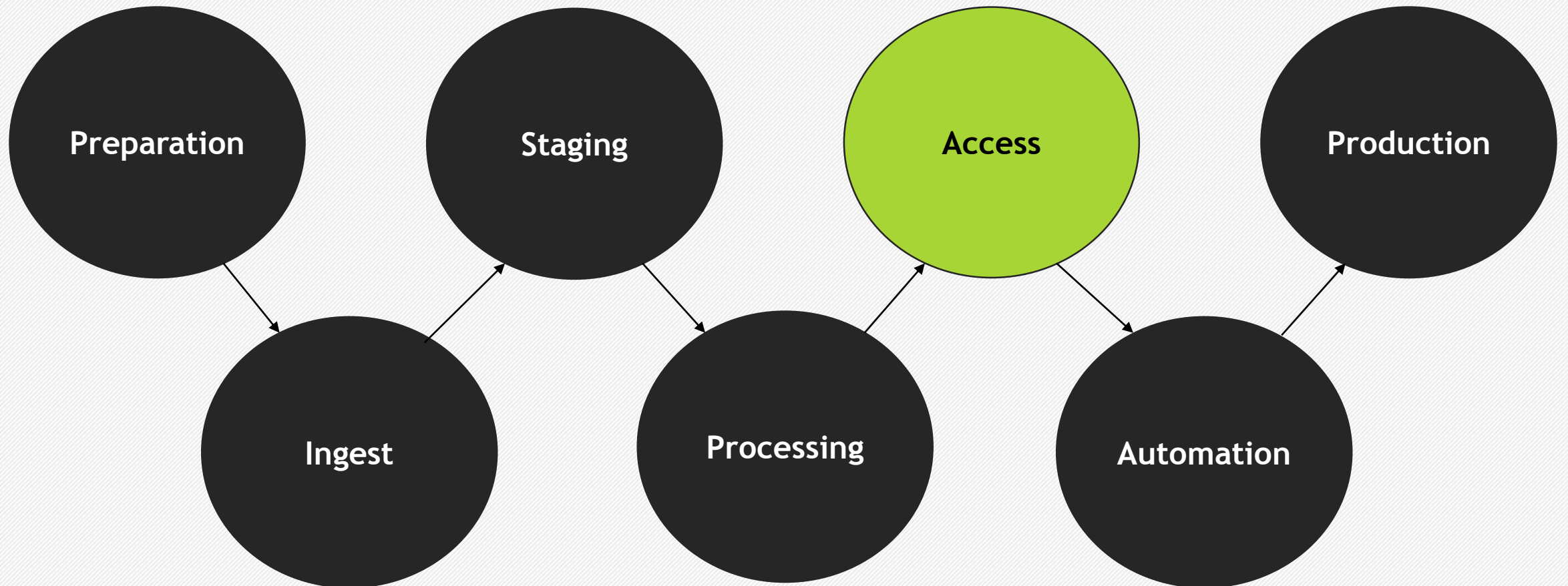
27

- Ergebnisdaten werden mit Timestamp versehen
- Aktualität kann dadurch identifiziert werden
- Löschen von alten Daten per Truncate - könnten durch Rohdaten in HDFS erneut erzeugt werden
- Alternativ z.B. Erzeugung von Snapshots möglich

„Truncate ‘hashtags‘“

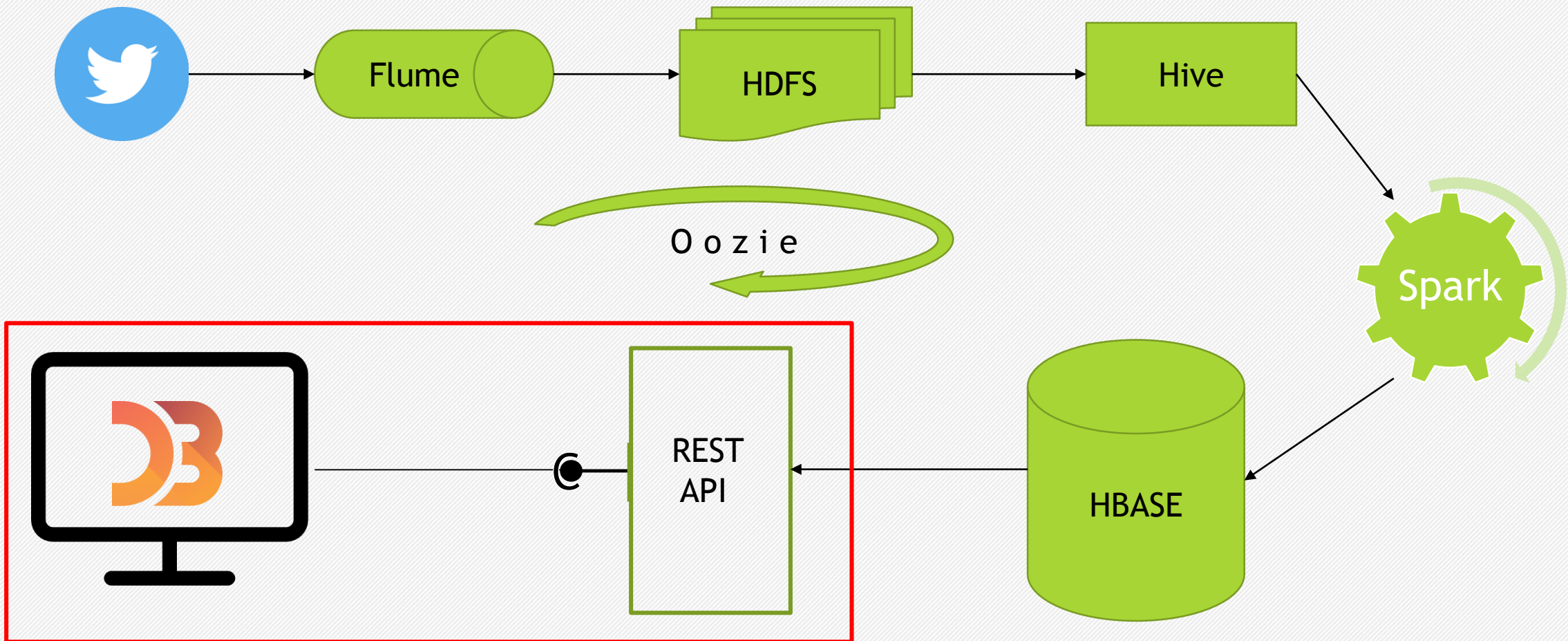
„Snapshot ‘hashtags‘ ‘hashtag_20170210‘“





Gesamtüberblick

29



Access - drei Möglichkeiten

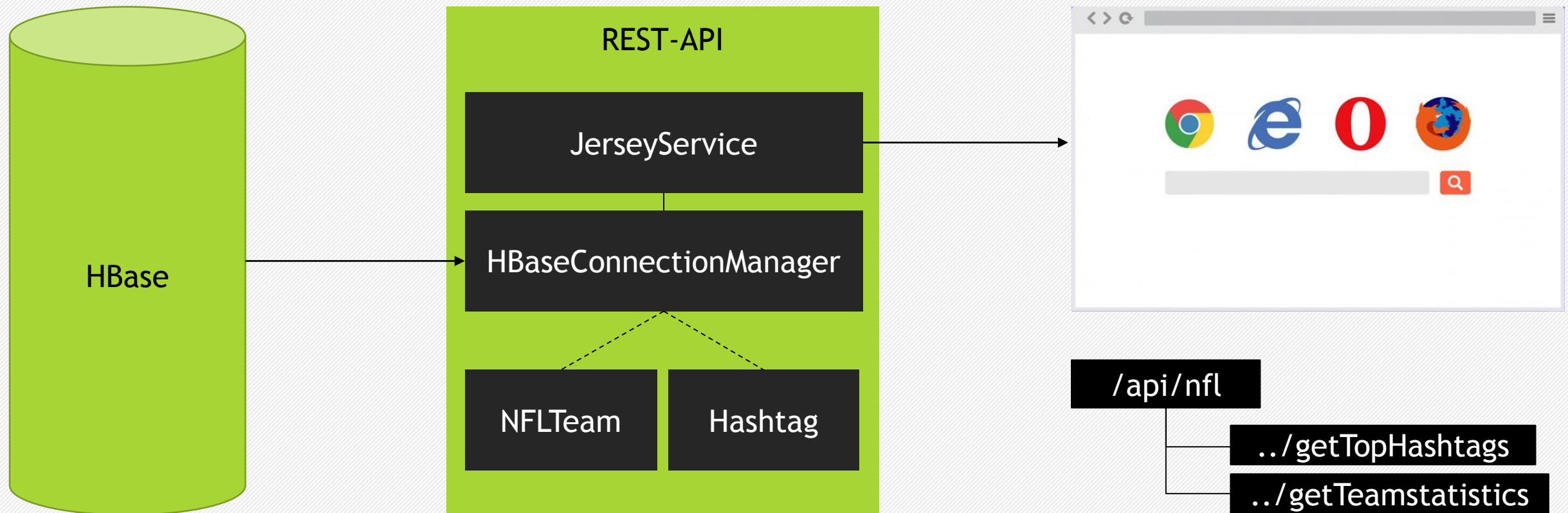
30

- Nativer Hadoop-Zugang durch gängige APIs HBase / Hue
- Eigene Rest-API für Zugriff
- Visualisierung durch D3.JS, Bootstrap

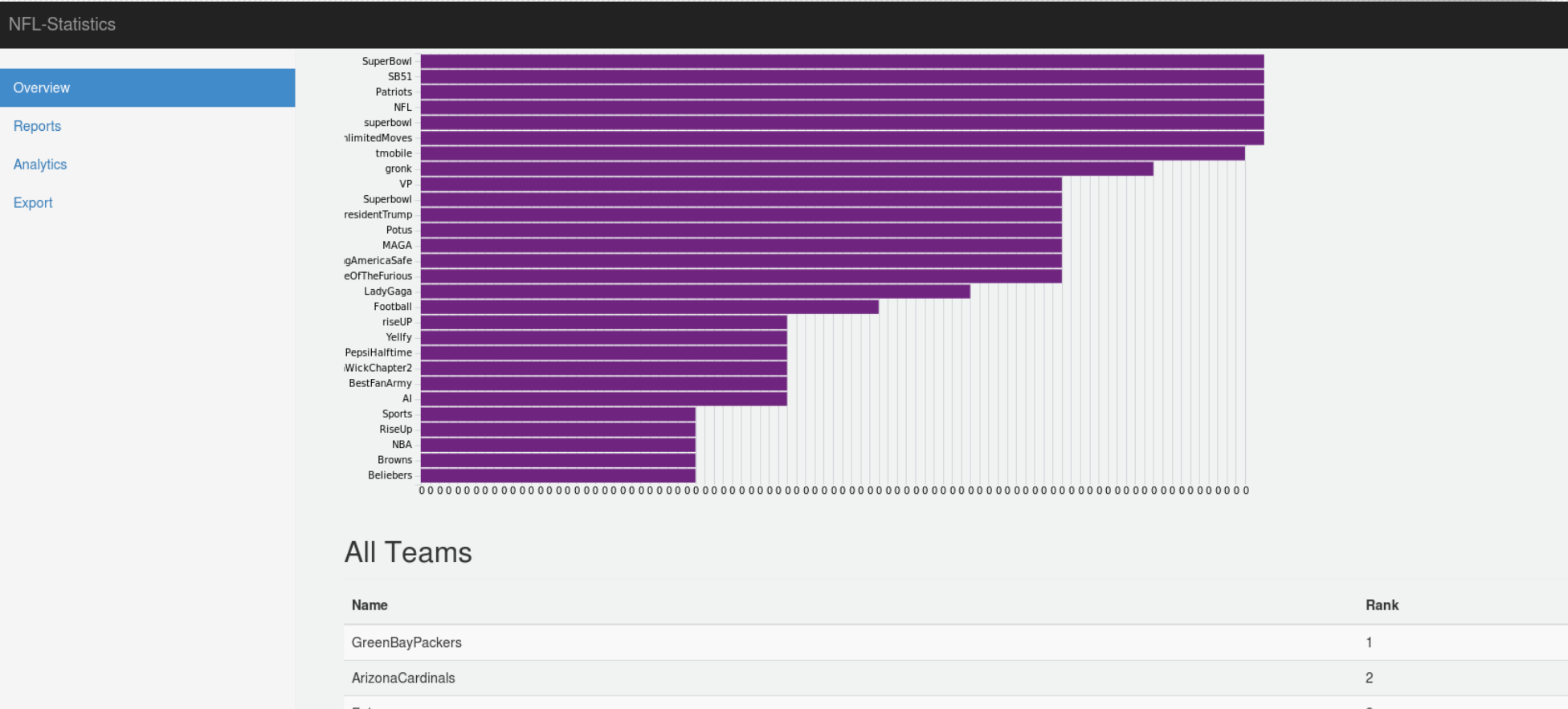


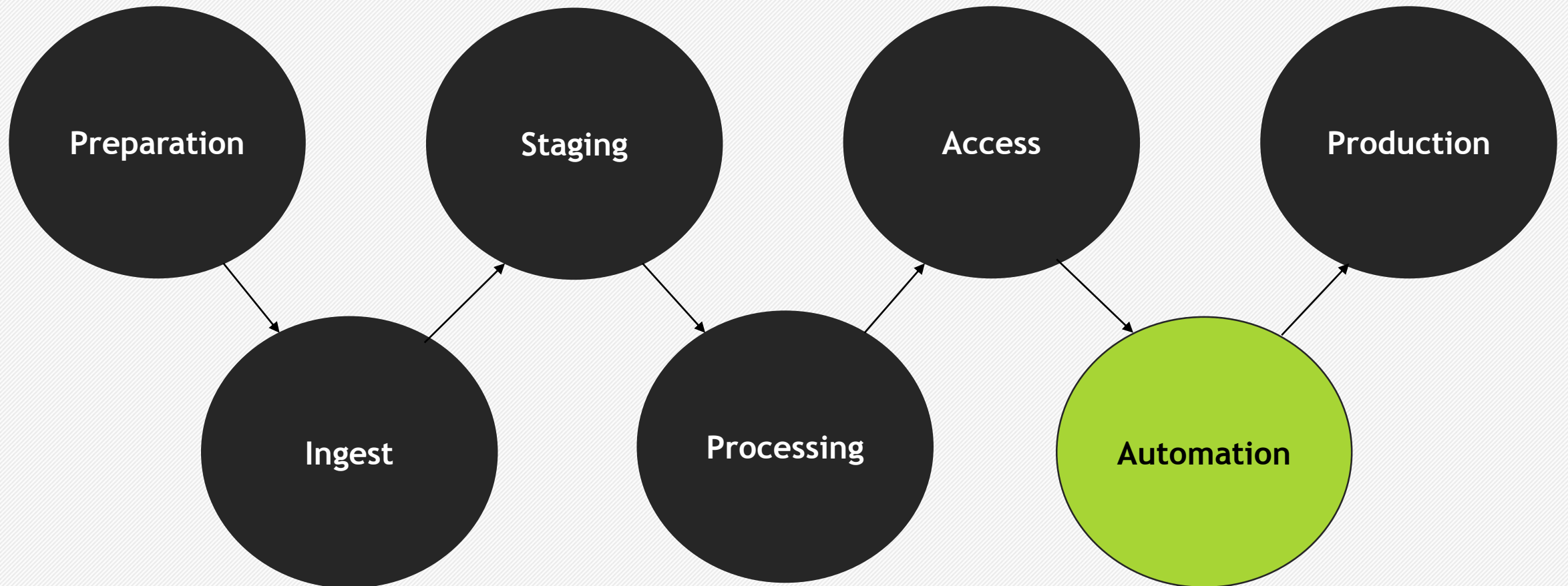
Access

31

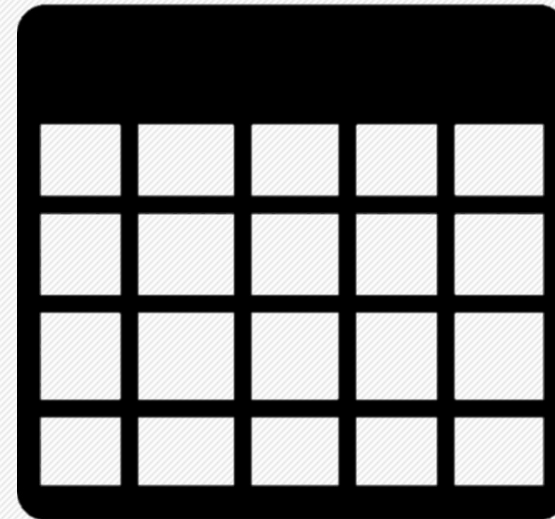


Access - Frontend

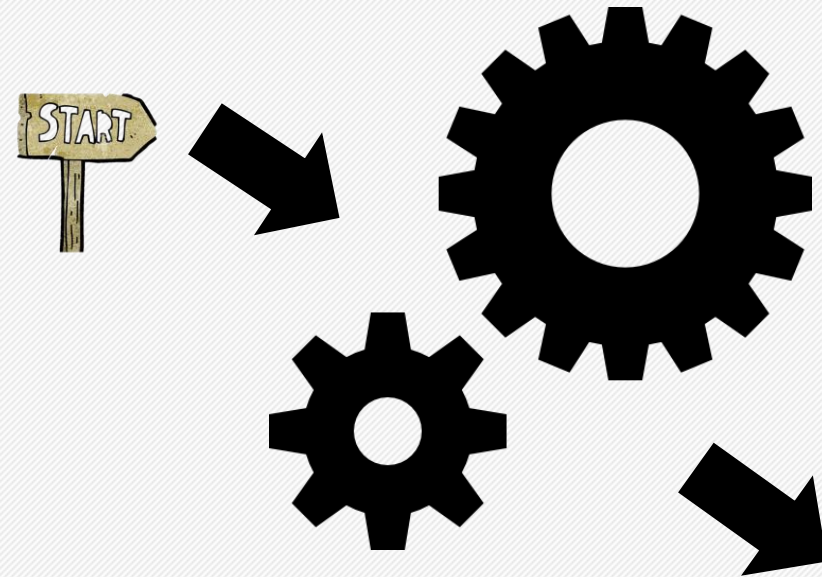


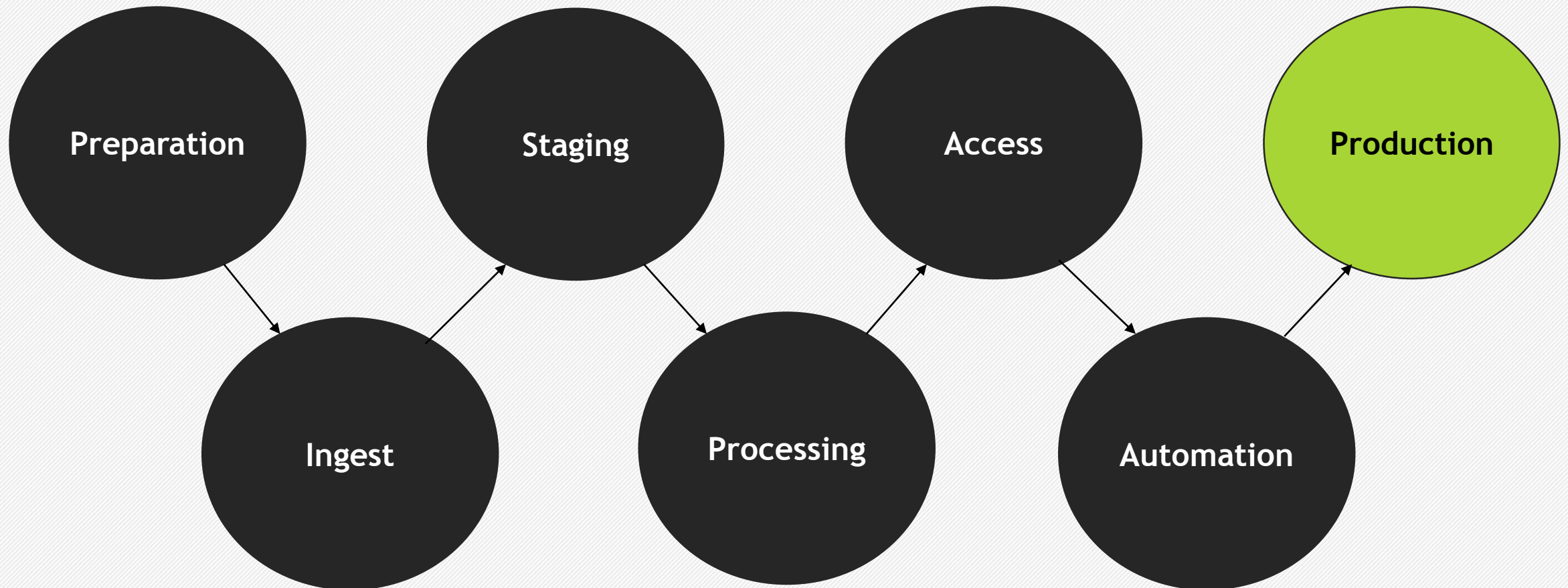


- Oozie triggert den Workflow
- Bedingung: Flume muss bereits laufen
- 1. Partitionierung (Hive)
 - 1.1 Fügt Partitionierung hinzu
 - 1.2 Erstellung von Tabellen




- 2. Datenverarbeitung (Spark)
 - 2.1 Stößt stündlich den Spark-Job an
 - 2.2 Mitgabe der spezifischen Parameter






Production - CI

37

Travis CI [Blog](#) [Status](#) [Help](#)

Cubaner

My Repositories +

✓ Cubaner/BigDataProjekt # 30

⌚ Duration: 6 min 37 sec

📅 Finished: about 3 hours ago

Cubaner / BigDataProjekt  build unknown

Current

Branches

Build History

Pull Requests

More options

✓ development minor changes

🔗 #30 passed

🔄 Restart build

👤 Commit ef53171

👤 Compare e8812ac..ef53171

👤 Branch development

👤 Gemba69 authored and committed

⌚ Elapsed time 6 min 37 sec

📅 about 3 hours ago

Job log

View config

Remove log

Raw log

1 Worker information

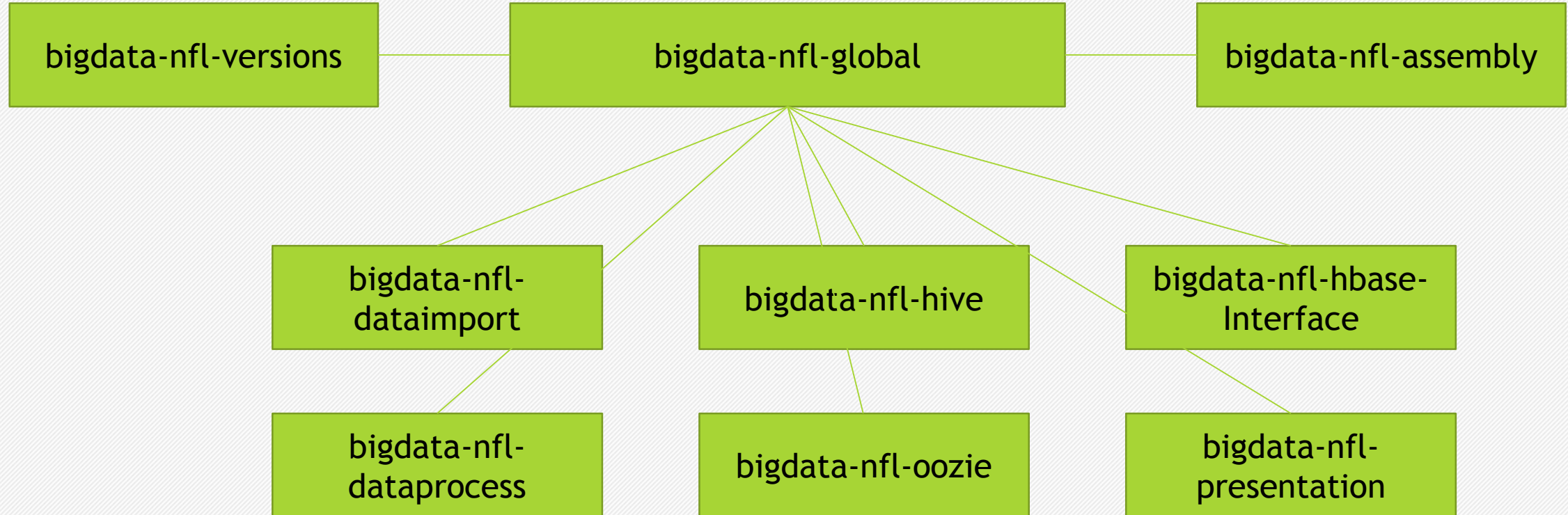
6 Build system information

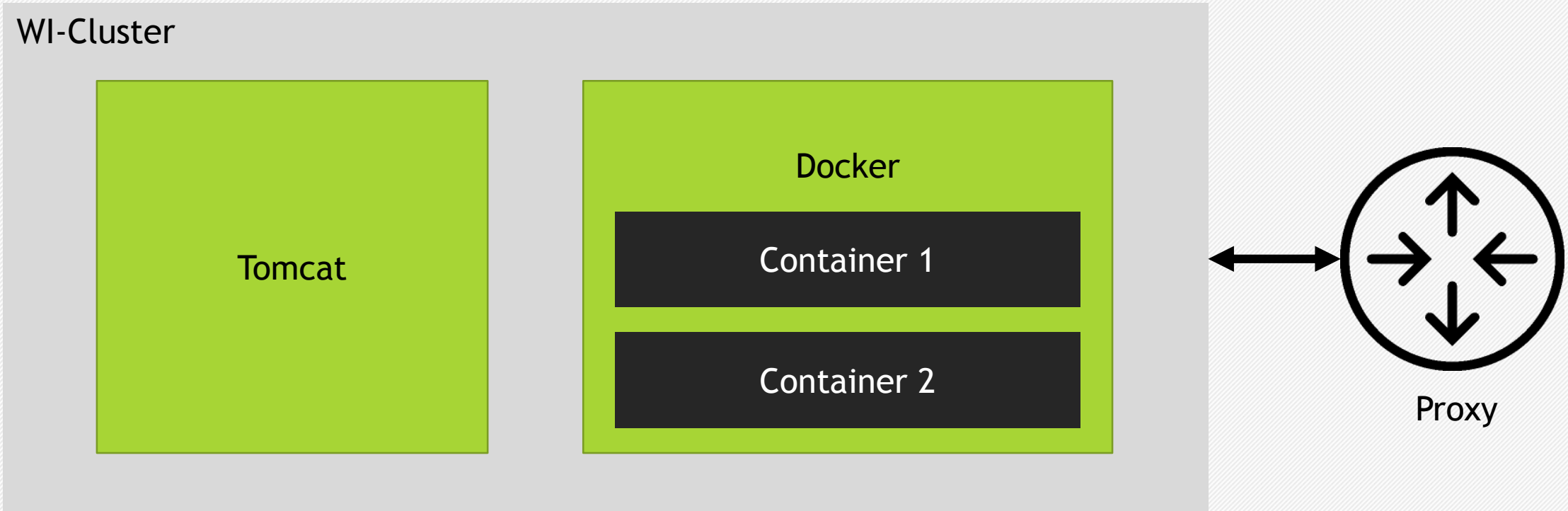
73

74 \$ export DEBIAN_FRONTEND=noninteractive

118 \$ jdk_switcher use oraclejdk8

119 Switching to Oracle JDK8 (java-8-oracle), JAVA_HOME will be set to /usr/lib/jvm/java-8-oracle





Schlussbetrachtung

40

Aktueller Implementierungsstand

41

- Dateningest via Flume und Ablage in HDFS
- Zugriff via HIVE
- Einfache Spark Transformation zum Zählen der Hashtags
- Oozie Workflow mit Partitioning und antriggern von Spark
- HBASE Rest API
- D3.JS Frontend



- Instabilität der Virtualisierungsumgebung und der einzelnen Services während der Entwicklungszeit
- Versuch des Einrichtens einer zentralen VM (Installation, Proxy, ...)
- Hoher Konfigurationsaufwand
- Hoher Hardwarebedarf
- Schwieriges Durchdringen aller verfügbaren Komponenten -> Viele wirre Wege führen zum Ziel
- Schwierige Fehler- und Lösungssuche



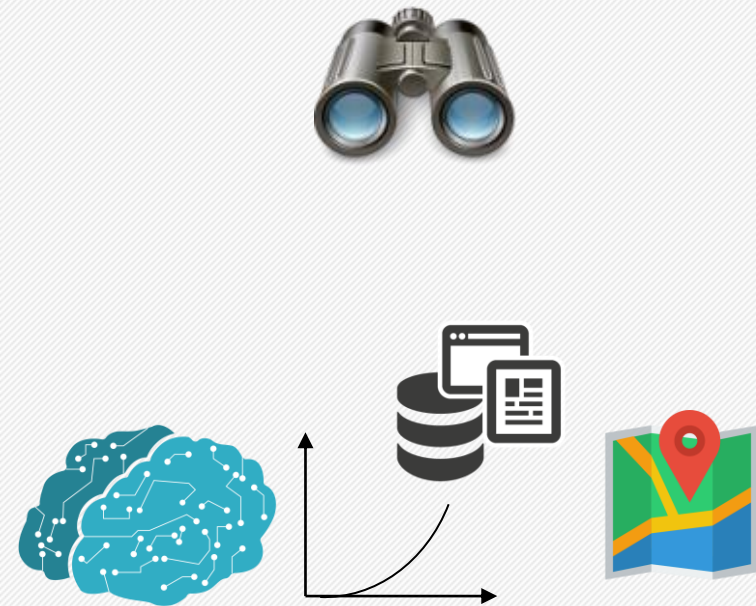
Gain (Lessons Learned)

43

- Automatisierung ist im Hadoop Umfeld äußerst wichtig und hilfreich
- Lange Entwicklungszyklen
- Produktivitätserhöhung durch Einarbeitung
- Einblicke in vielen verschiedenen BigData Entwicklungsbereichen
- Verständniserweiterung
- Scala, Maven Vertiefung, Server, ...























- Höherer Grad der Automatisierung (Skripte)
- Fertigstellung der Maven-Module
- Testautomatisierung
- Erweiterung der Eingangsdaten
- Machine-Learning Ansatz integrieren
- Google API für Lokalisierung der Tweets



Startseite

Status **Alle Systemzustandsprobleme**  8 Konfiguration ▾

● Cloudera QuickStart (CDH 5.7.0,...) ▾

●  Hosts	 1	
●  Flume		▾
●  HBase	 1	▾
●  HDFS	 3	▾
●  Hive	 2	▾
●  Hue		▾
●  Impala		▾
●  Key-Value Store ...		▾
●  Oozie		▾
●  Solr		▾
●  Spark		▾
●  Sqoop 1 Client		▾
●  Sqoop 2		▴
●  YARN (MR2 Incl...	 1	▴
●  ZooKeeper		▴

45

Vielen Dank für Ihre Aufmerksamkeit

46

Initialer Import der Teams aus einer .txt in Hbase

47

- Textdatei befindet sich im Projektordner `BigDataProjekt/bigdata-nfl-dataimport/resources/`
- Anschließend folgende Befehle ausführen:
 1. `cloudera@quickstart> hbase shell`
 2. `hbase> create 'teams', 'team', 'rank', 'devision', 'synonyms'`
 3. `hbase> exit`
 4. `cloudera@quickstart> hadoop fs -put /home/cloudera/NFLTEAMS.txt /user/cloudera`
 5. `hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.separator=, -Dimporttsv.columns=HBASE_ROW_KEY,rank,devision,synonyms,team teams /user/cloudera/NFLTEAMS.txt`
 6. `cloudera@quickstart> hbase shell`
 7. `hbase> scan 'teams'`