

# Big Data Engineering



## Trendanalyse Premier League

Andreas Schäfer (790689)  
Mario Hellenkamp (660608)  
Marius Spancken (790667)



# Agenda

---

- › Idee
- › Verantwortlichkeiten
- › Konzeption
- › Umsetzung
- › Live Demo
- › Herausforderungen
- › Fazit

# Idee



- Twitter Posts mit Bezug zu Fußballspielen und Wettquoten dazu benutzen um die Stimmung der Twitter Nutzer mit den Quoten in Vergleich zu setzen.



Gruppe A		Gruppe B		Gruppe C	
Brasilien	Mexiko	Spanien	Chile	Kolumbien	Elfenbeinküste
Kroatien	Kamerun	Niederlande	Australien	Griechenland	Japan
Brasilien - Kroatien	00:14.00 - 0:20.00 / 1. Halbzeit	Spanien - Niederlande	01:13.00 - 1:21.00 / 1. Halbzeit	Kolumbien - Griechenland	00:14.00 - 1:00.00 / 1. Halbzeit
Mexiko - Kamerun	01:13.00 - 1:20.00 / 1. Halbzeit	Chile - Australien	00:14.00 - 1:00.00 / 1. Halbzeit	Elfenbeinküste - Japan	00:14.00 - 1:00.00 / 1. Halbzeit
Brasilien - Mexiko	01:13.00 - 1:20.00 / 1. Halbzeit	Australien - Niederlande	00:14.00 - 1:00.00 / 1. Halbzeit	Kolumbien - Elfenbeinküste	00:14.00 - 1:00.00 / 1. Halbzeit
Kamerun - Kroatien	00:14.00 - 1:00.00 / 1. Halbzeit	Spanien - Chile	00:14.00 - 1:00.00 / 1. Halbzeit	Japan - Griechenland	00:14.00 - 1:00.00 / 1. Halbzeit
Kamerun - Brasilien	00:14.00 - 1:00.00 / 1. Halbzeit	Australien - Spanien	00:14.00 - 1:00.00 / 1. Halbzeit	Japan - Kolumbien	00:14.00 - 1:00.00 / 1. Halbzeit
Kroatien - Mexiko	00:14.00 - 1:00.00 / 1. Halbzeit	Niederlande - Chile	00:14.00 - 1:00.00 / 1. Halbzeit	Griechenland - Elfenbeinküste	00:14.00 - 1:00.00 / 1. Halbzeit
Gruppe A		Gruppe B		Gruppe C	
Platz	Team	Platz	Team	Platz	Team



# Verantwortlichkeiten

Input

- Mario Hellenkamp

Processing

- Andreas Schäfer

Output

- Marius Spancken

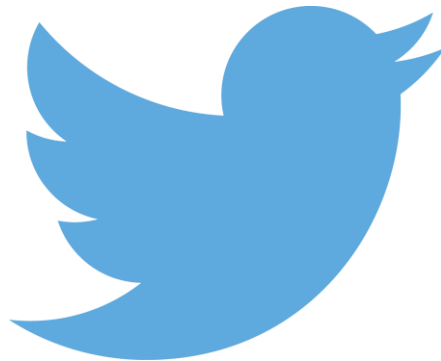






# Umsetzung – Einlesen der Tweets

- › Java-Applikation mit Twitter4J
- › Sammelt Tweets mit Hashtags der 20 Premier-League Mannschaften
- › Speicherung in HDFS
- › Format: Ein Tweet je Zeile: Zeit|User|Text
  - › Angepasst für TextInputFormat von MapReduce





# Umsetzung – Einlesen der Wettquoten

- › Quelle: [betclic.com](http://betclic.com)
- › Quellformat: XML
- › Java-Programm liest folgende Quoten ein:
  - › 3-Wege-Quoten: Gewinn Team A, Unentschieden, Gewinn Team B
  - › ca. 50 Ergebnisquoten je Spiel
- › Speicherung der Quoten in MongoDB







# Umsetzung - MapReduce

- › Quelle: Tweets aus HDFS
- › Analyse der Tweets mit SentiStrength-Verfahren der Universität Wolverhampton
  - › Konzipiert für die Messung von positiven bzw. negativen Stimmungen in Sozialen Netzwerken
  - › Optimiert für kurze Aussagen
  - › Auf Grundlage von Wörterlisten, die Stimmungen aufzeigen und deren Stärke (Adjektive)
  - › Ergänzt um relevante Begriffe im Fußball wie Gewinnen, Verlieren etc.
  - › Bewertung jedes Tweets mit einem Positiv- und Negativwert.





# Umsetzung - MapReduce

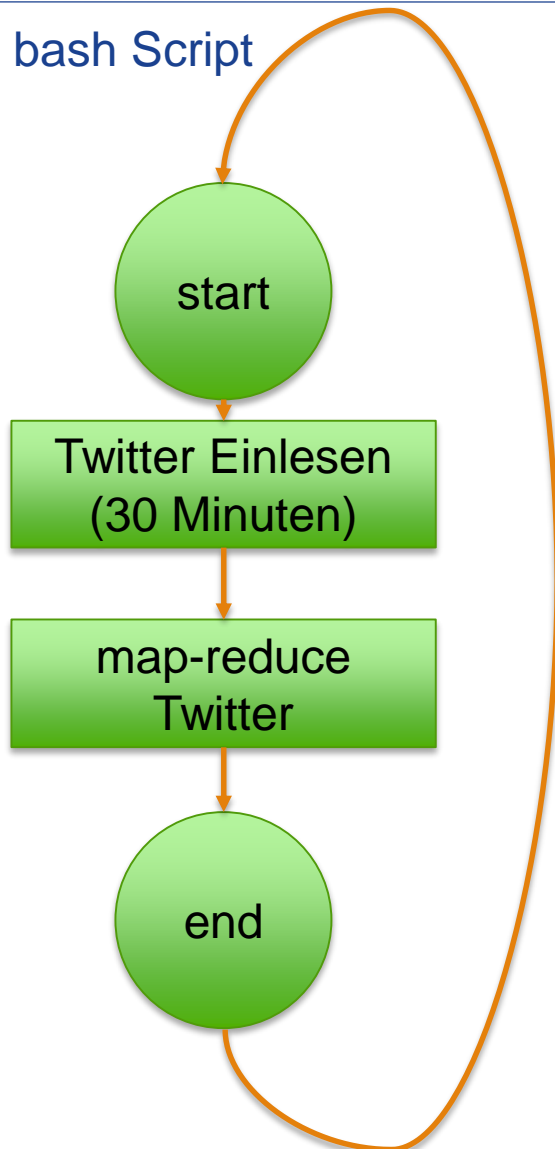
- › Zählt für definierten Zeitraum:
  - › Anzahl aller Tweets
  - › Anzahl Tweets mit Hashtags einer Mannschaft
  - › Anzahl Tweets mit Hashtag von nur einer Mannschaft
  - › Wörter die in Verbindung mit einem Hashtag verwendet werden
  - › Anzahl positive und negative Tweets
  - › Bildung einer Summe von SentiStrength-Werten



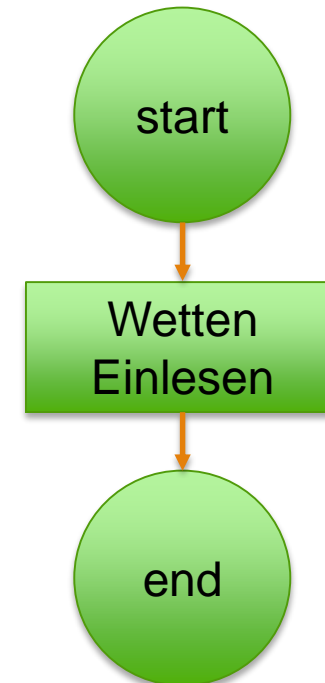


# Umsetzung - Scheduling

bash Script



Oozie Job (alle 60 Minuten)





# Umsetzung - Ausgabe

DB



Webserver



Apache  
Tomcat

Version 8.0

Browser

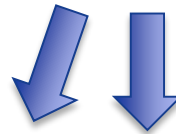
HTML

- Matches
- Tweets
- Statistiken
- Bewertungen

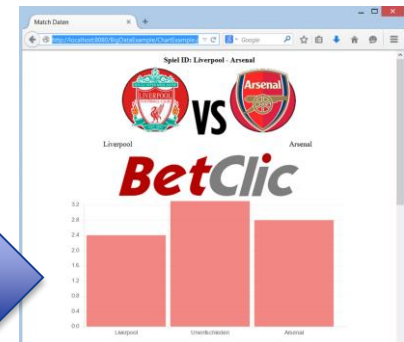
Index.jsp  
DBOutput.jsp  
Dashboard.jsp  
Architecture.jsp



MongoDBConnector.jar



OpenCloud JavaScript  
Opencloud.jar Charts.js





**Live Demo**















# Herausforderungen

- › Cloudera Twitter Beispiel nicht erfolgreich auf der VM zum laufen gebracht
  - › Komplexität des Beispiels
  - › Oozie Konfiguration → Dienst wurde nicht richtig gestartet
- › Einbinden des MapReduce Jobs in den Oozie Workflow
  - › Manuelles Ausführen vom MapReduce Job erfolgreich
  - › Kompletter Workflow in Oozie daher nicht möglich
- › Performance/Restriktionen der Virtuellen Maschinen
  - › Zerstörte VMs nach einem Neustart
  - › Proxyeinstellungen der VM
  - › Verwendung des Cloudera Managers nicht möglich



# Skalierbarkeit

- › Erweiterung auf weitere Sportarten/Ligen (fachliche)
- › Erweiterung des Clusters (technisch)

Komponente	Skalierbar (fachlich)	Skalierbar (technisch)
WettenEinlesen.jar		
TwitterEinlesen.jar		
MapReduceTwitter.jar		
bash Script		
MongoDB		
Apache Tomcat		



# Fazit

- › Planarchitektur weitestgehend umgesetzt
- › Durchgängiger Workflow
- › Aussagekräftige Ergebnisse
- › Viele Probleme im Zusammenhang mit VM
- › Verhalten von Softwarekomponenten unterschiedlich je nach Umgebung (Linux, Windows, Cluster, lokale VM)