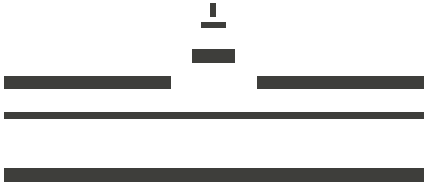


Westfälische
Wilhelms-Universität
Münster

TMetrics

Data-Mining mit Twitter - ein Projektseminar

wissen.leben
WWUMünster

12. März 2014

Gliederung

Gliederung

Projektvorstellung - Beteiligte

Leitung des Projektseminars:

- Prof. Dr. Jan Vahrenhold
- Wolfgang Paul

Teammitglieder:

- Daniel Günther
- Wladimir Haffner
- Olaf Markus Köhler
- Sebastian Lichtenfels
- Erwin Quiring
- Andreas Riddering
- Björn Roß
- Jens Sandmann
- Torsten Scholz
- Tobias Wenzel

Projektvorstellung - Ideenfindung

Erste Aufgabe: Projektidee finden

- in 2er Teams fünf verschiedene Ideen
- zwei Favoriten gewählt und kombiniert

Projektvorstellung - Die Projektidee

- Favorit 1: Kino-Modul, Meinungsbild
- Favorit 2: Clustering
- Ideen wurden kombiniert

Projektvorstellung - Die Projektidee

Resultierende Projektidee:

- Aktivität über einen Zeitraum
- Meinungsbild (Sentiment-Analyse)
- Clustering der Tweets
- Kino-Modul

Projektvorstellung - Die Projektidee

Resultierende Projektidee:

- Aktivität über einen Zeitraum
- Meinungsbild (Sentiment-Analyse)
- Clustering der Tweets
- Kino-Modul
- News-Modul

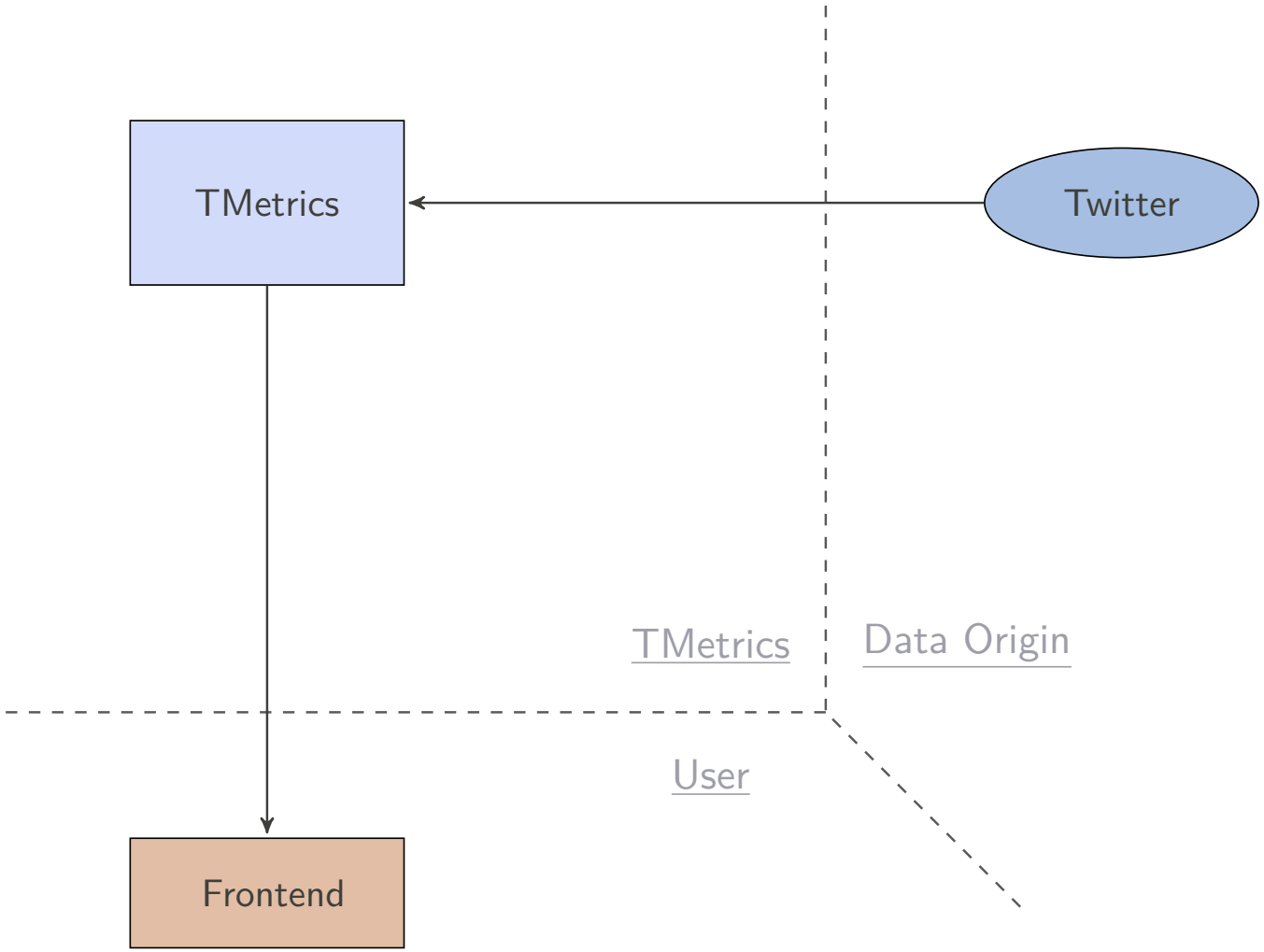
Und wie setzen wir das jetzt um?

Projektvorstellung - Verwendete Technologien

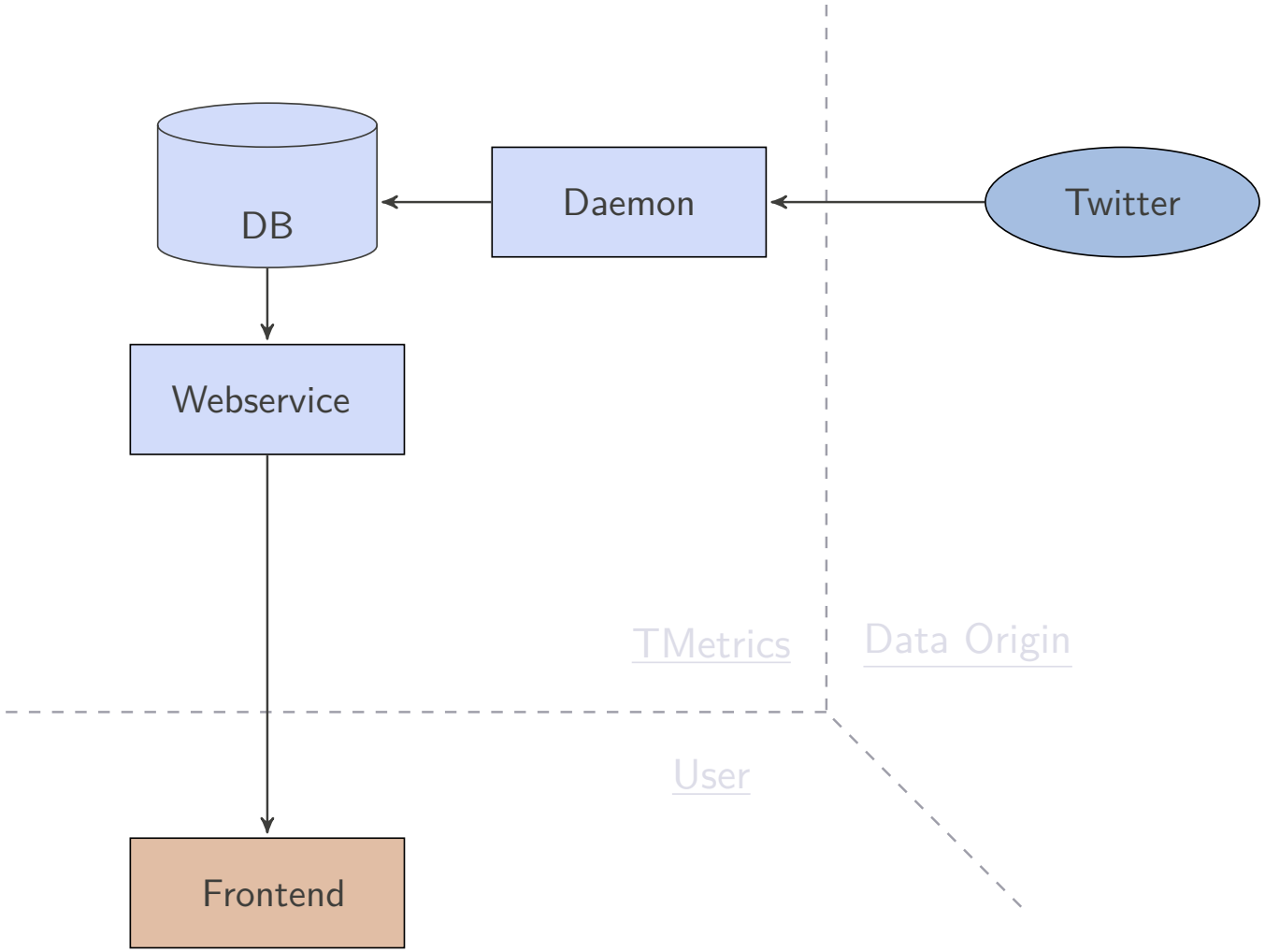
- Im Team bekannte Sprachen? $\rightarrow$ Java
- REST als Serverschnittstelle (Tomcat und Apache)
- Nutzen durch verfügbare Bibliotheken?
- Wie die Daten aus Twitter herausholen, speichern und wieder ausgeben?

Gliederung

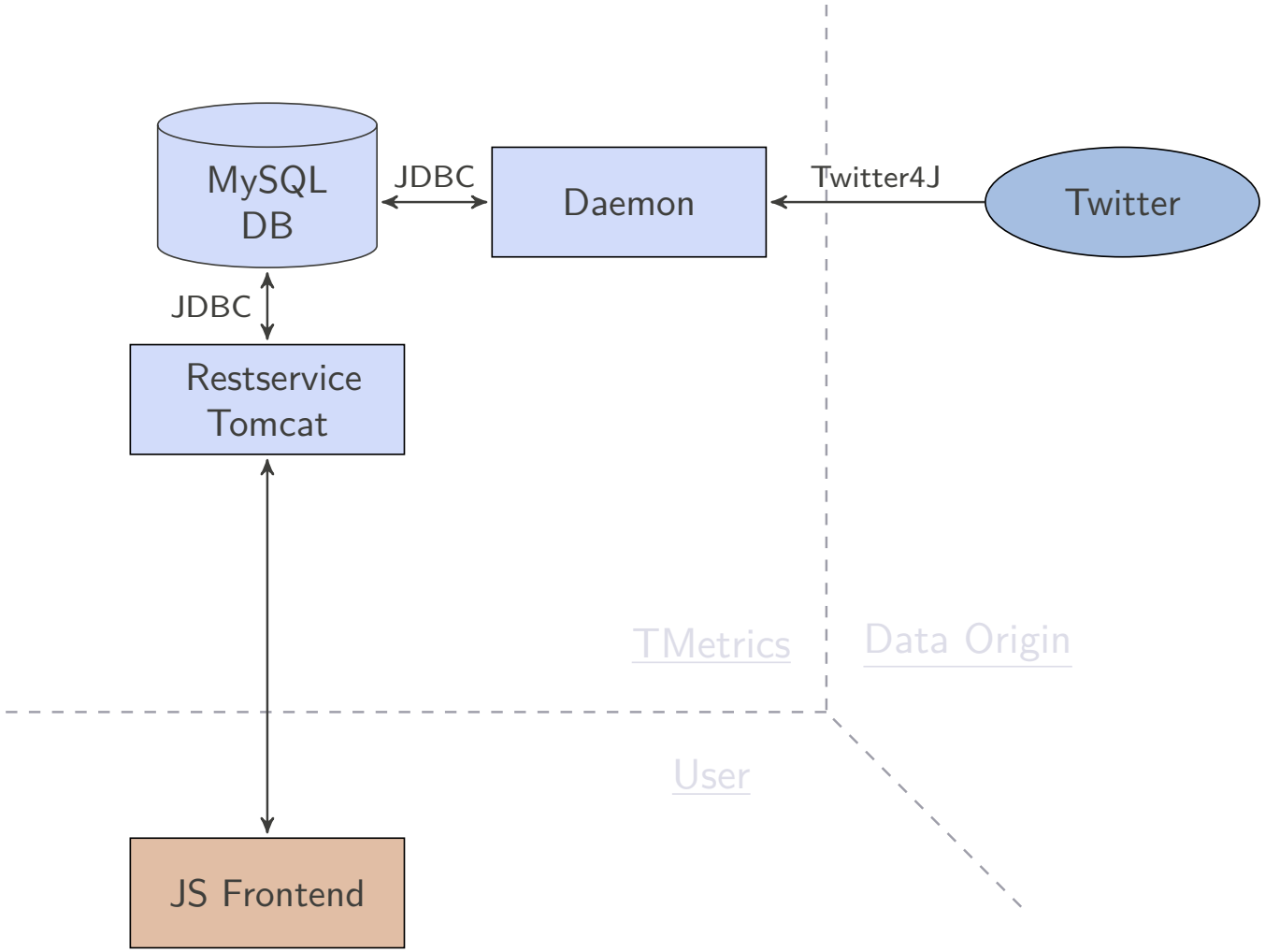
Aufbau - Architektur



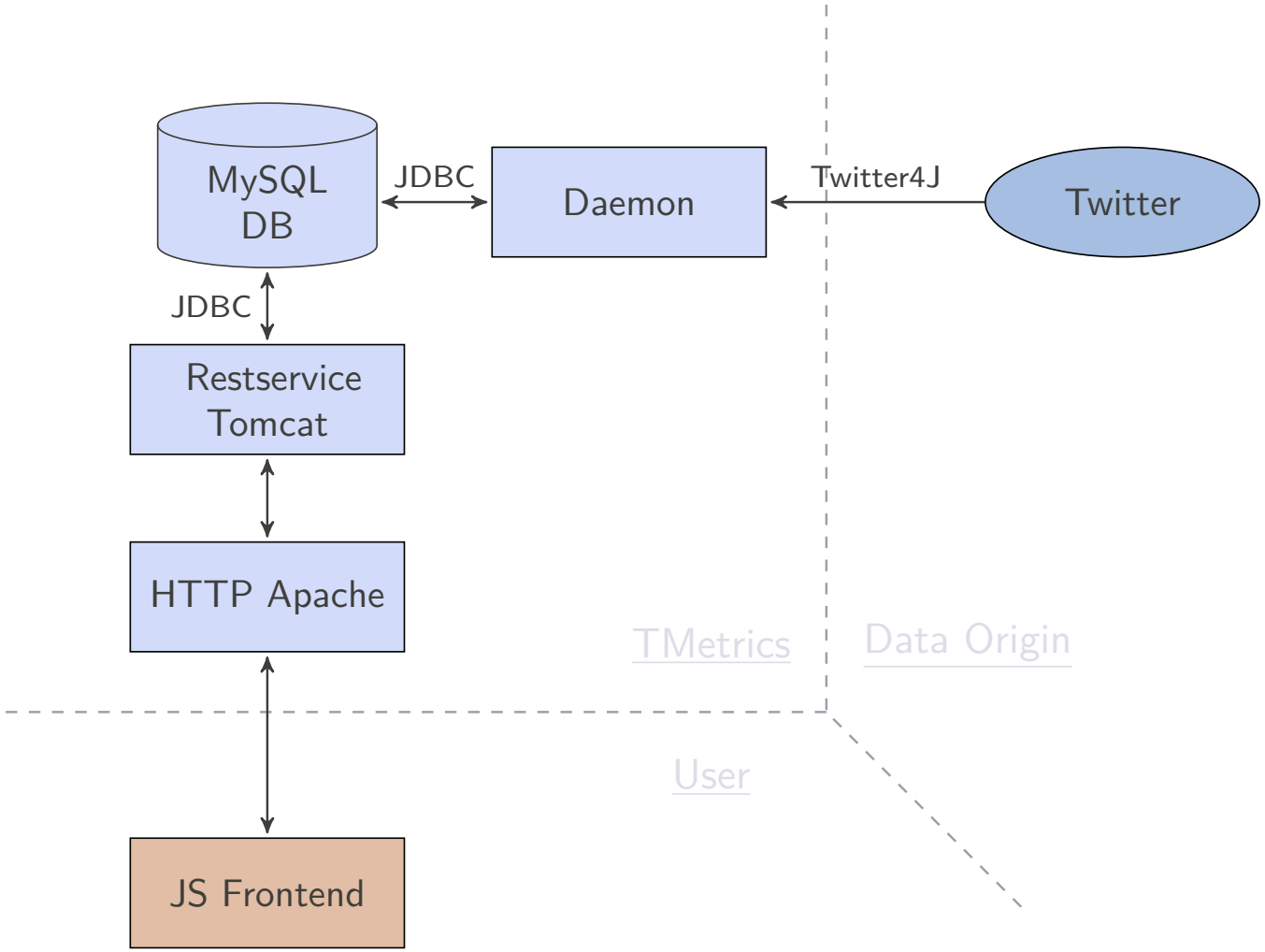
Aufbau - Architektur



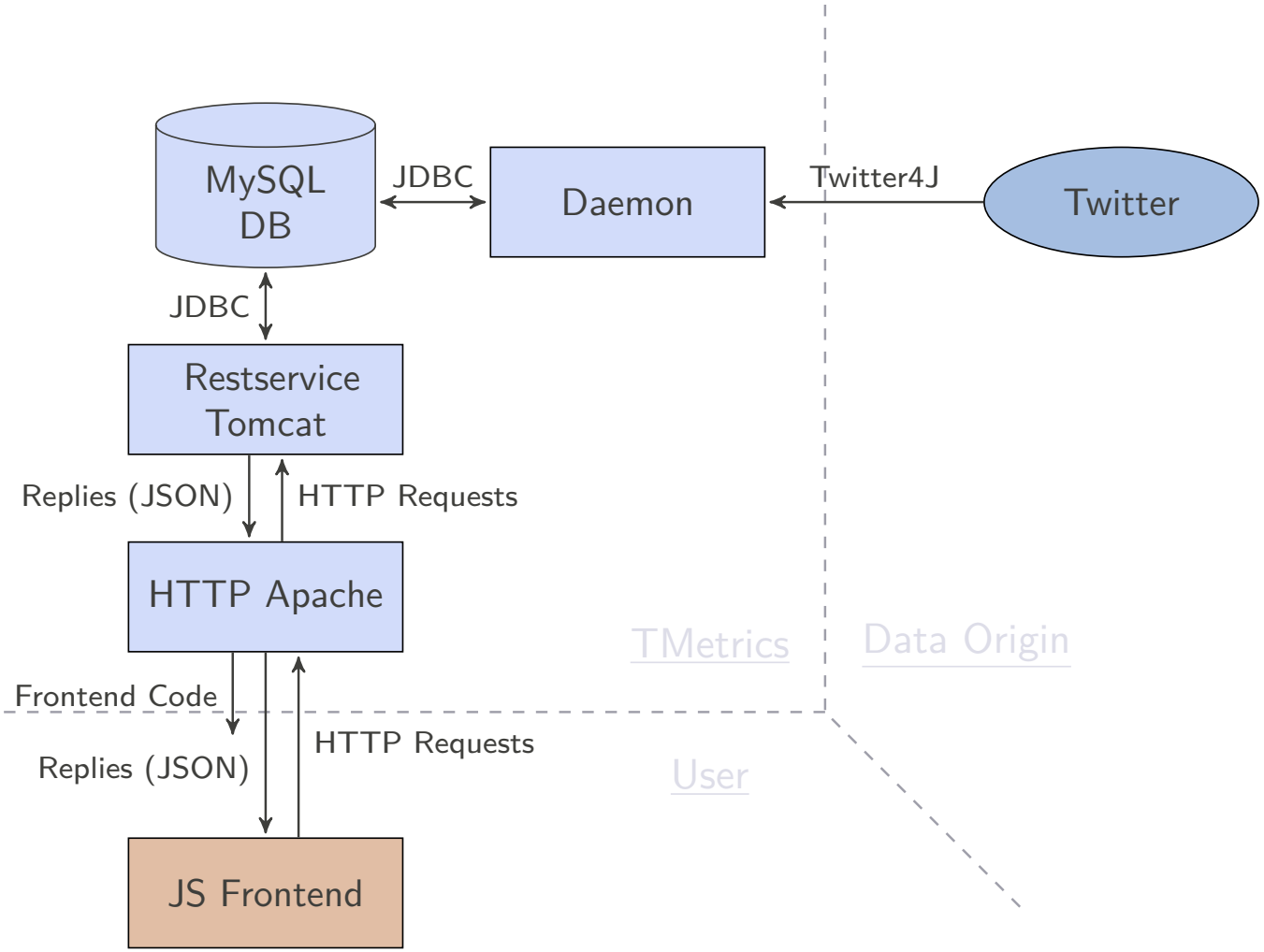
Aufbau - Architektur



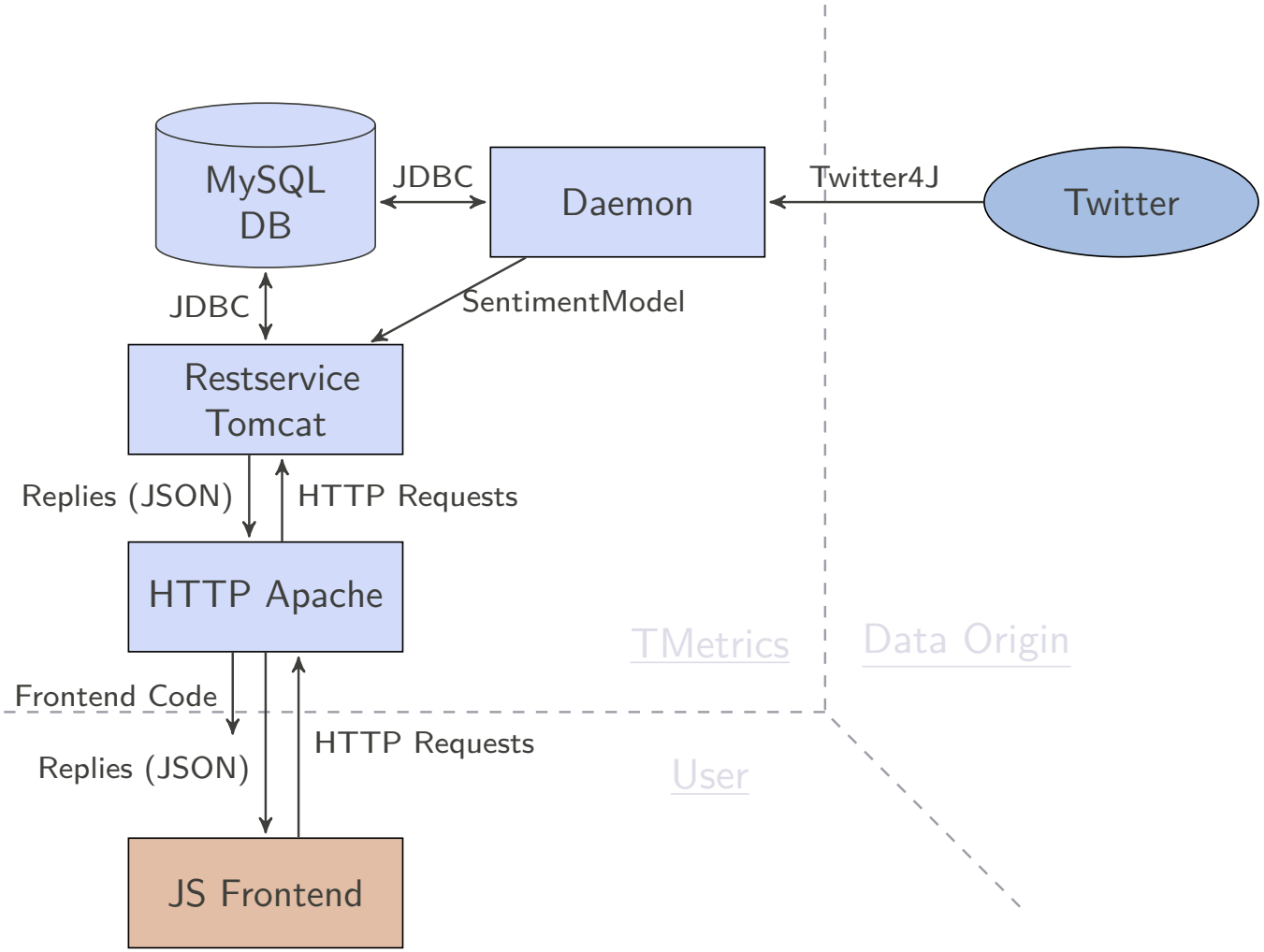
Aufbau - Architektur



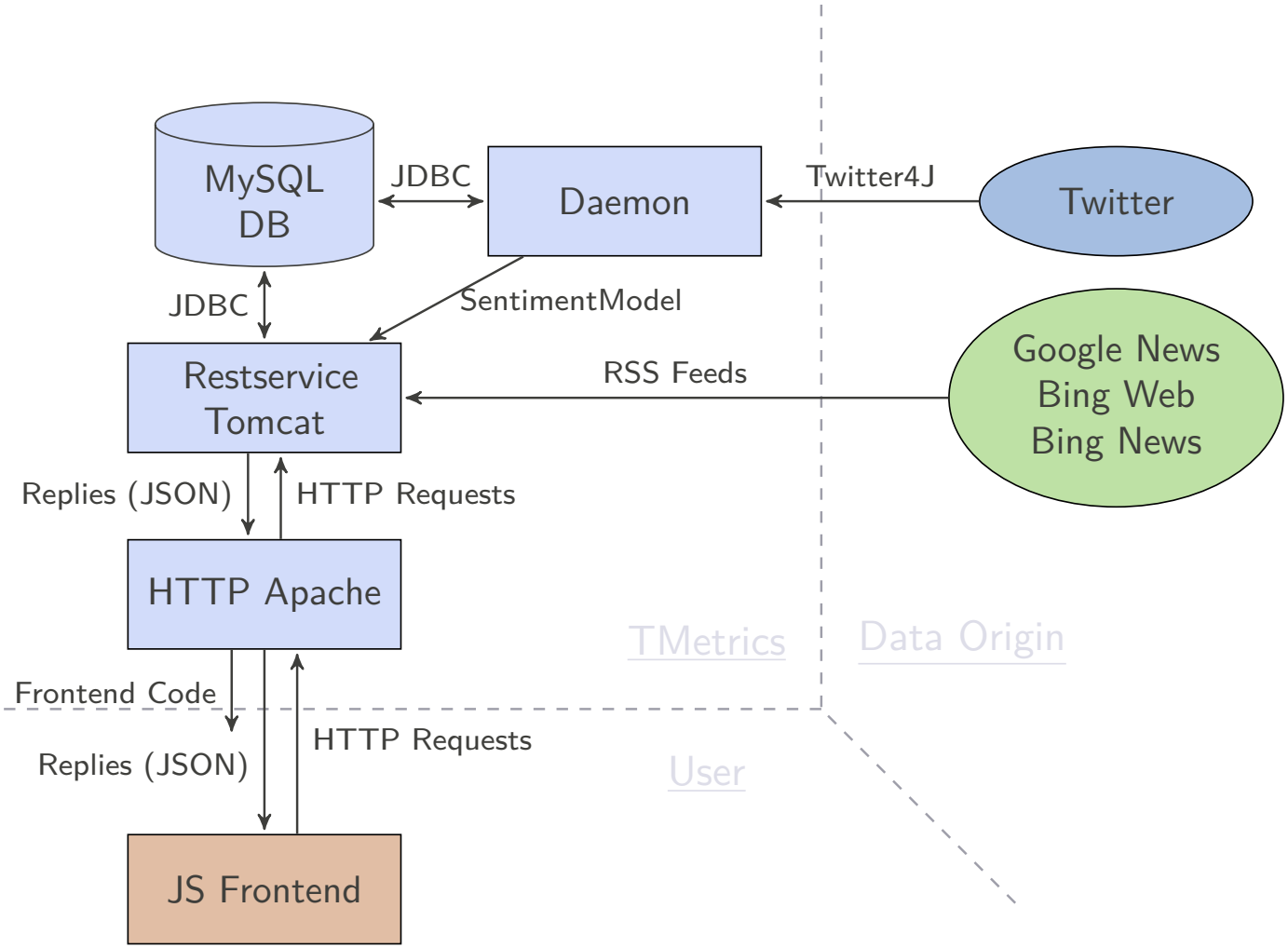
Aufbau - Architektur



Aufbau - Architektur

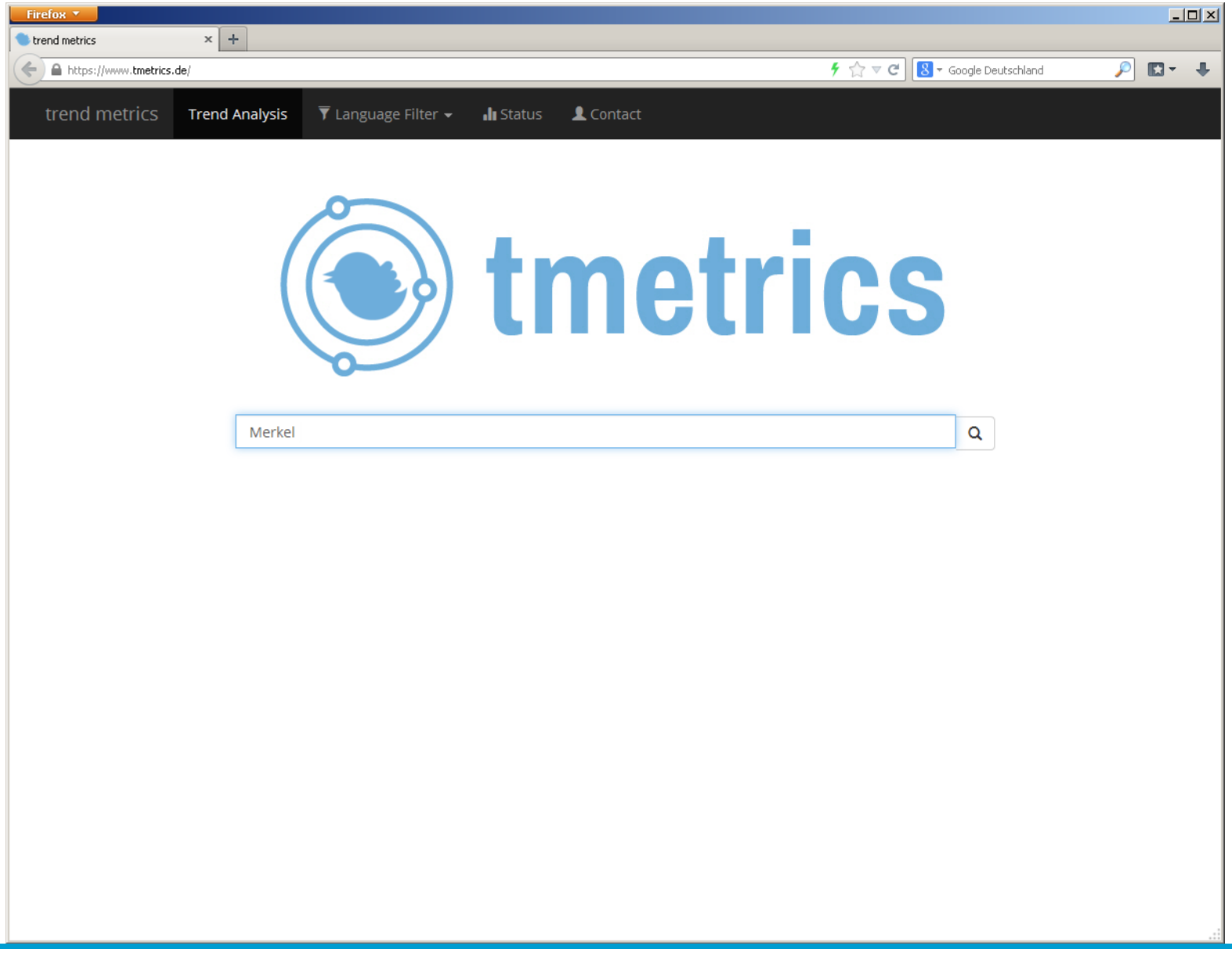


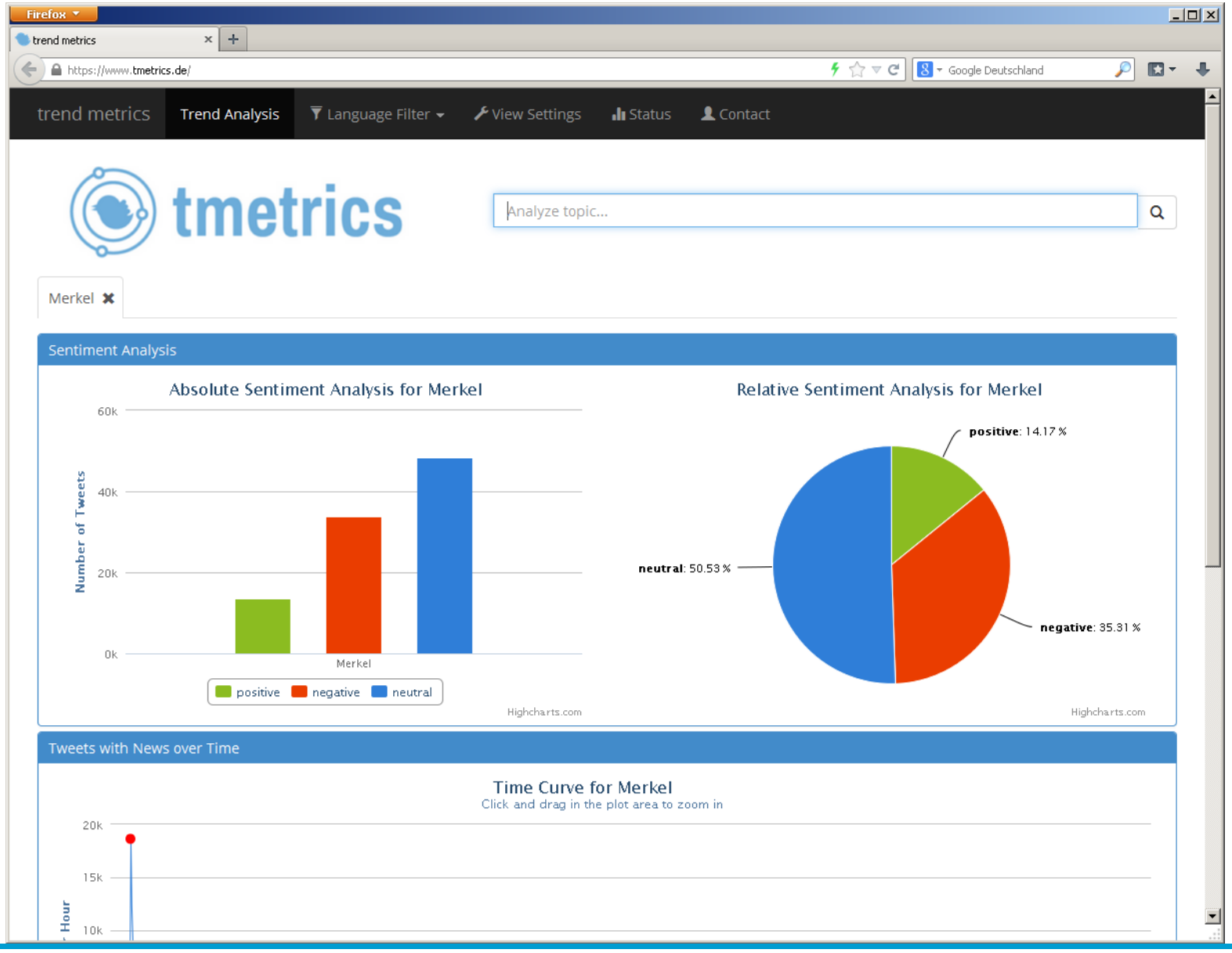
Aufbau - Architektur

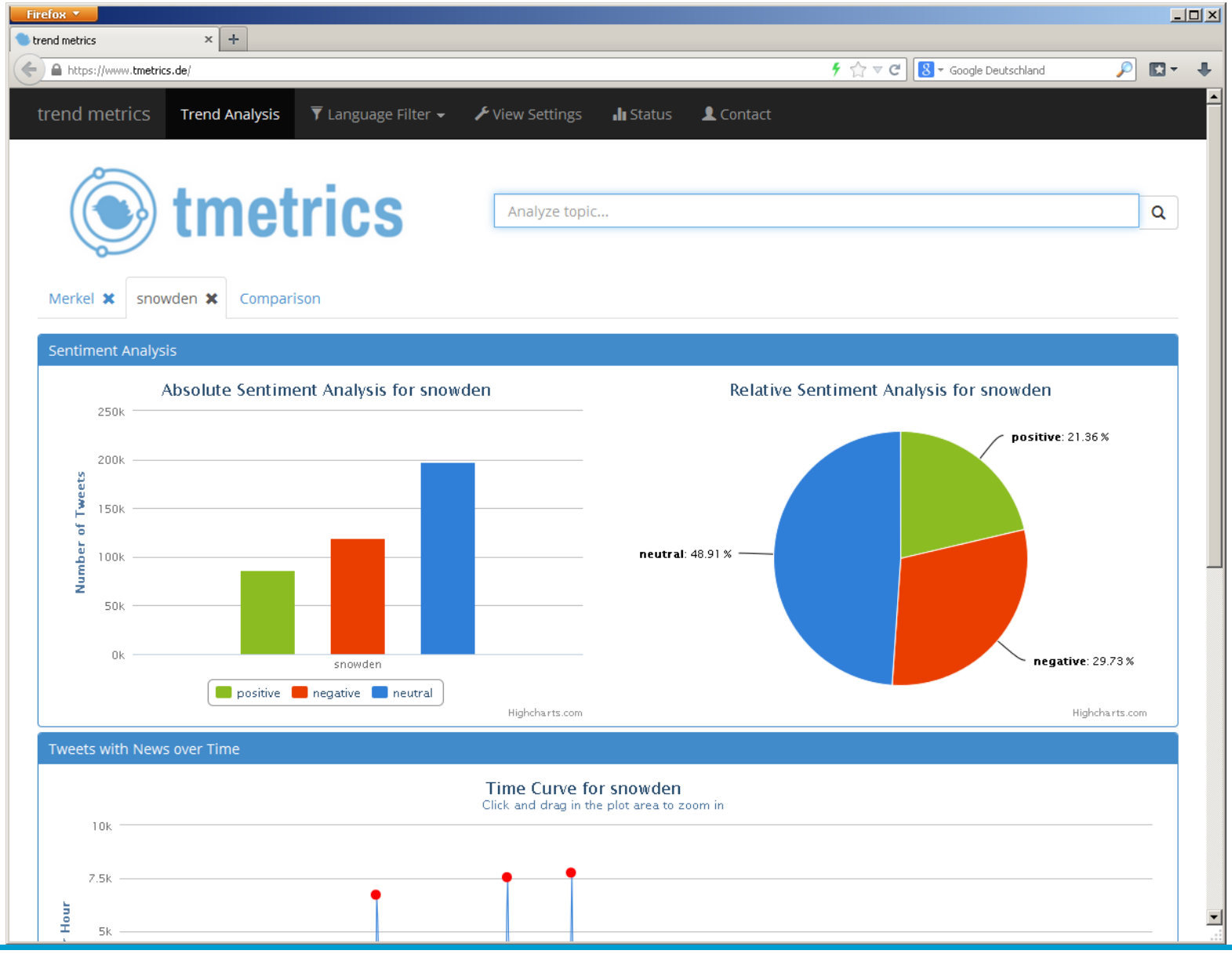


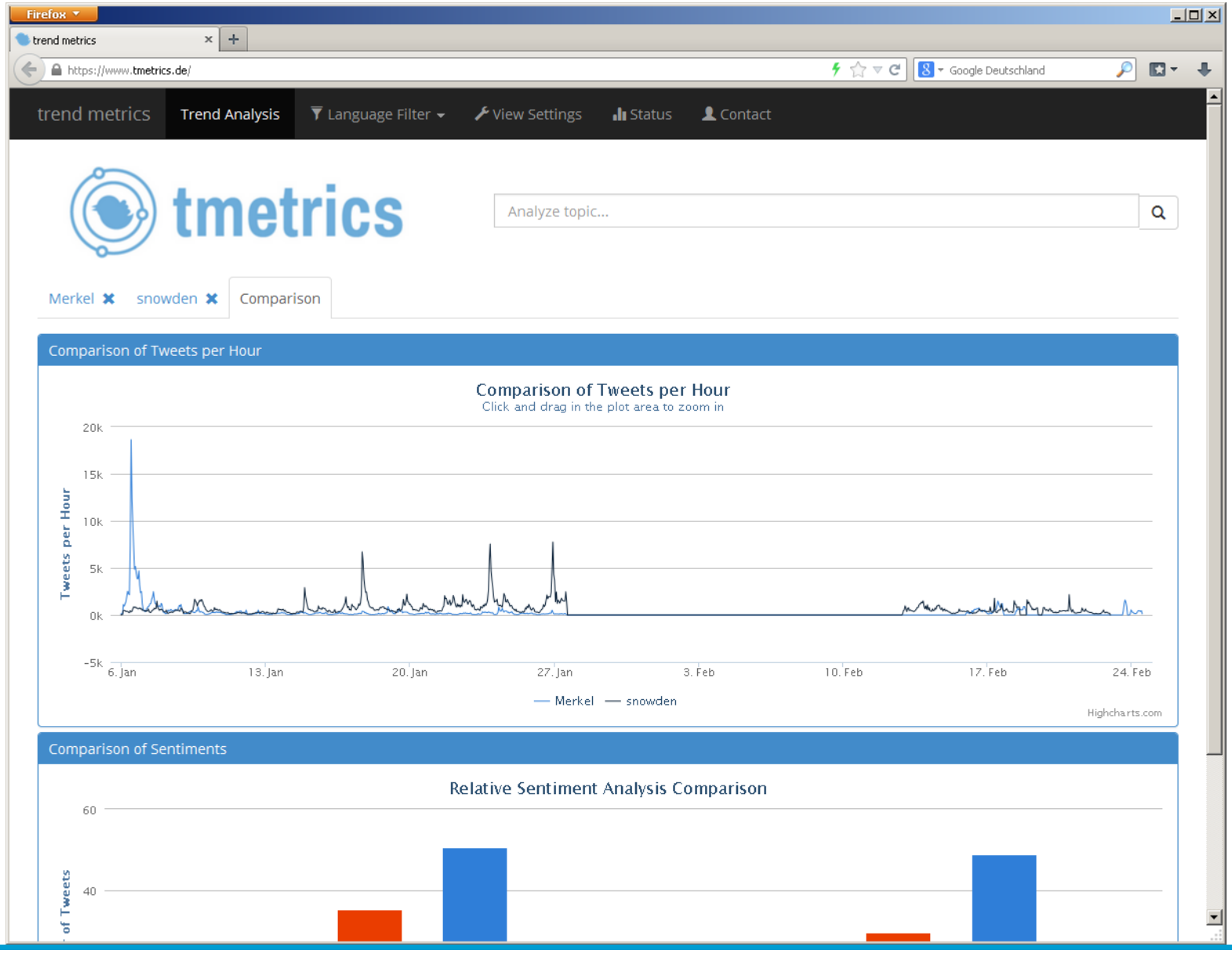
Aufbau - Frontend

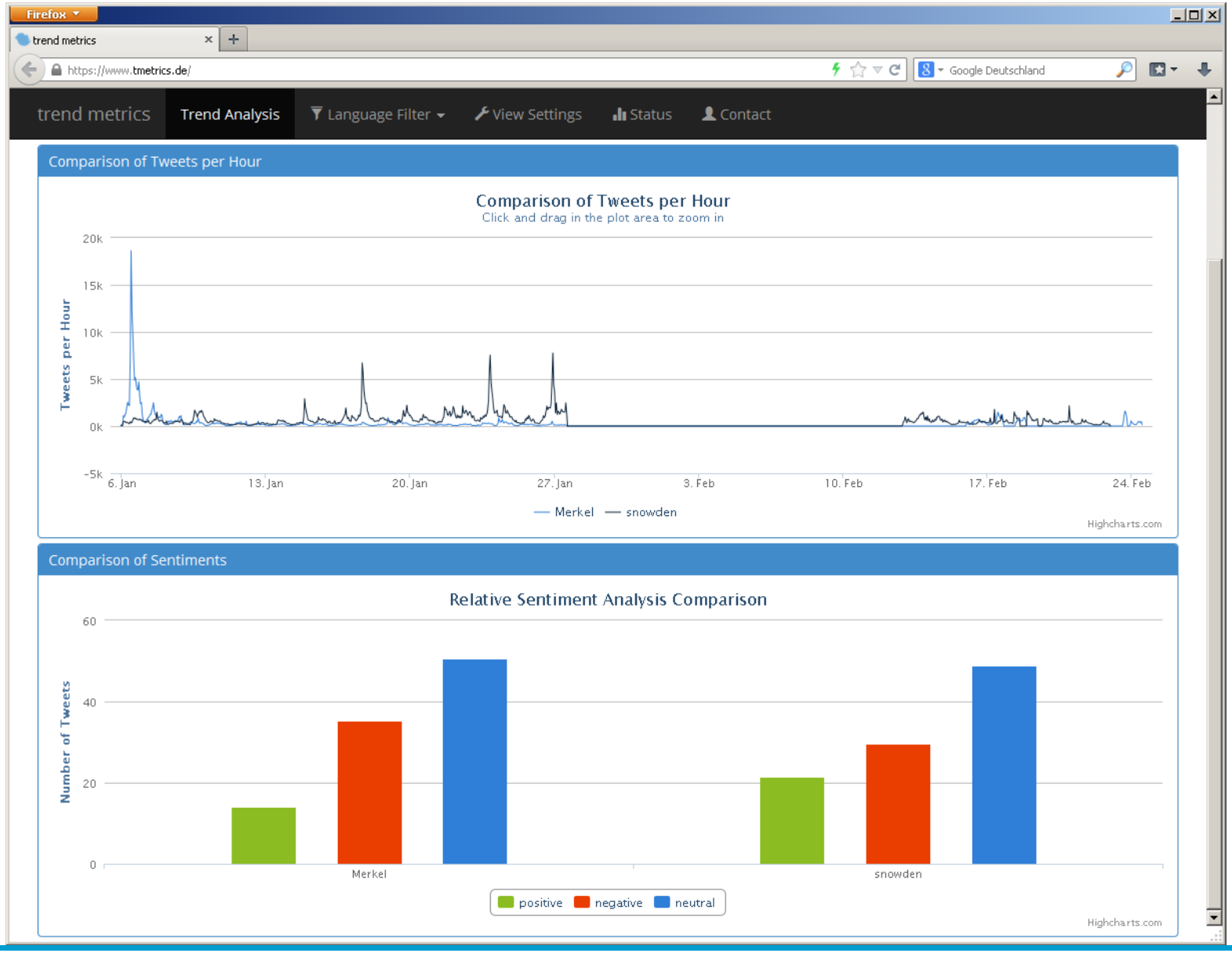
- Darstellung im Browser
- Verwendung verschiedener Frameworks (jQuery, Bootstrap, Highcharts)
- Darstellung mittels HTML5 und CSS3











Aufbau - Frontend

- Einfach und intuitiv zu bedienen
- Reiter und Zoom sind bekannte Bedienkonzepte
- Anzeigebereich nicht überladen, sondern übersichtlich gehalten
- Weitere Details zur Funktionalität folgen in der Demonstration

Gliederung

Gliederung

Module - Daemon - Aufgaben

- Sammeln von Tweets zu Begriffen
- Speichern von Tweets in der Datenbank
- Verwendung der Twitter API

Module - Daemon - Aufgaben

- Sammeln von Tweets zu Begriffen
- Speichern von Tweets in der Datenbank
- Verwendung der Twitter API

Module - Daemon - Aufgaben

- Sammeln von Tweets zu Begriffen
- Speichern von Tweets in der Datenbank
- Verwendung der Twitter API

Datenbank Daemon TwitterAPI

- Sentiment berechnen

Module - Daemon - Twitter-API

- Twitter hat eine offizielle API
- Search-API: REST-Anfragen liefern JSON-Objekte
- Twitter API unterliegt Restriktionen
- Kommunikation mit Twitter über Twitter4J

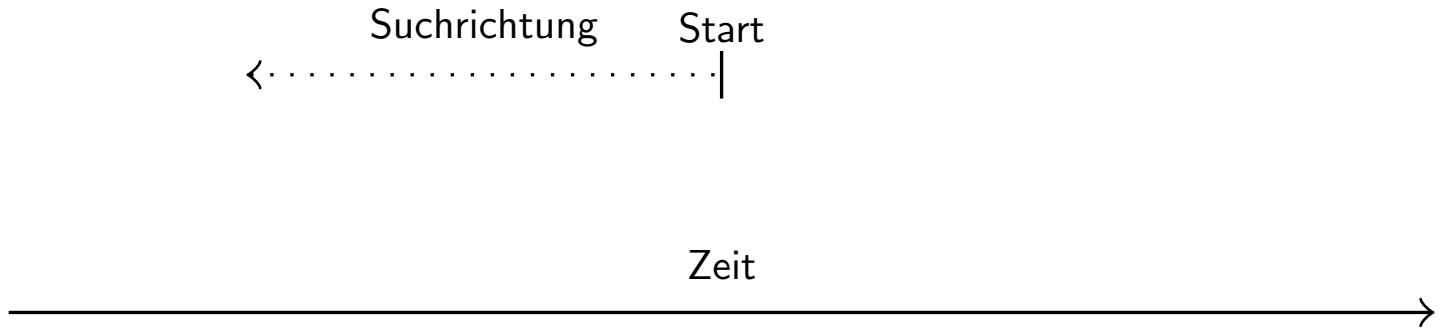
Module - Daemon - Suchstrategie

- Suche zu Suchbegriff immer rückwärts
- Suche: ältester Tweet ist Beschränkung für nächste Anfrage
- Tweets sind zeitlich sortiert

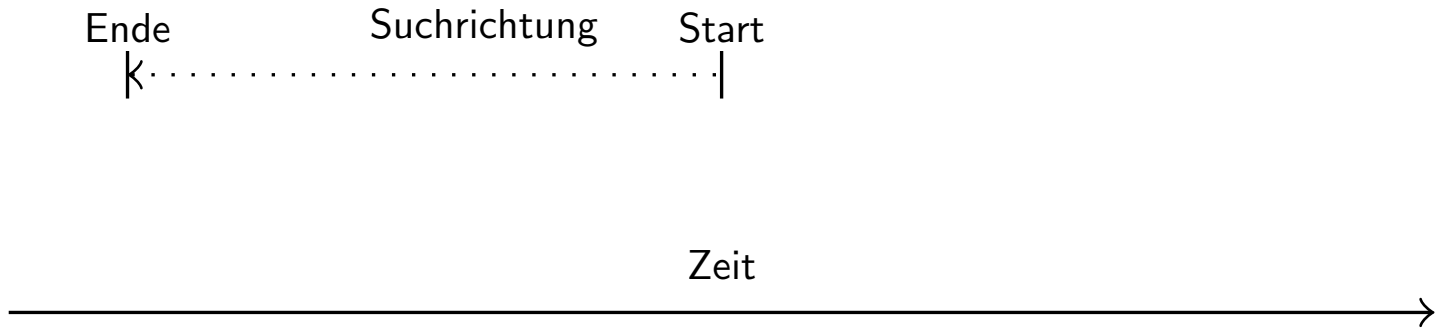
Module - Daemon - Suchstrategie

- Suche zu Suchbegriff immer rückwärts
- Suche: ältester Tweet ist Beschränkung für nächste Anfrage
- Tweets sind zeitlich sortiert
- Keine älteren Tweets mehr: Startzeitpunkt auf jetzt setzen
- Ab diesem Zeitpunkt wird erneut rückwärts gesucht

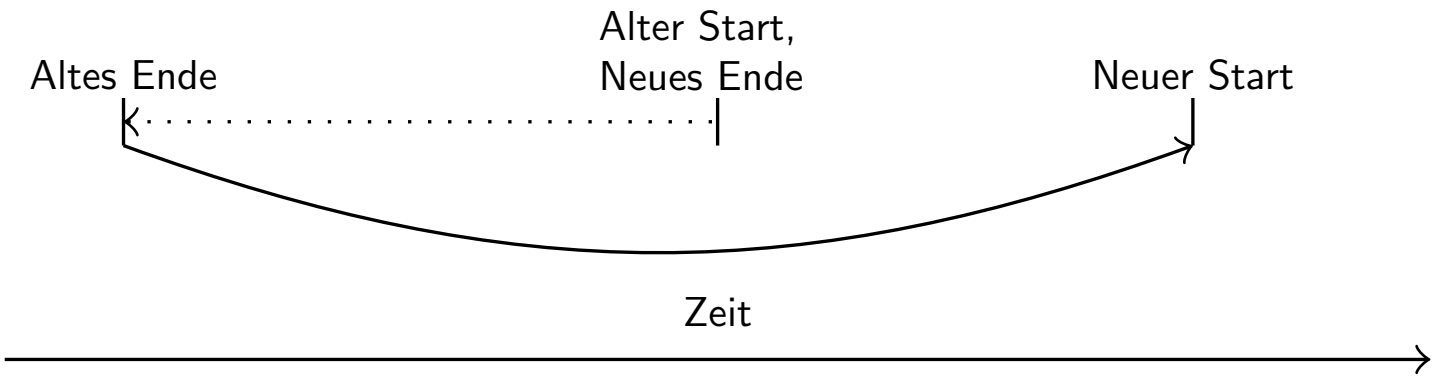
Module - Daemon - Suchstrategie



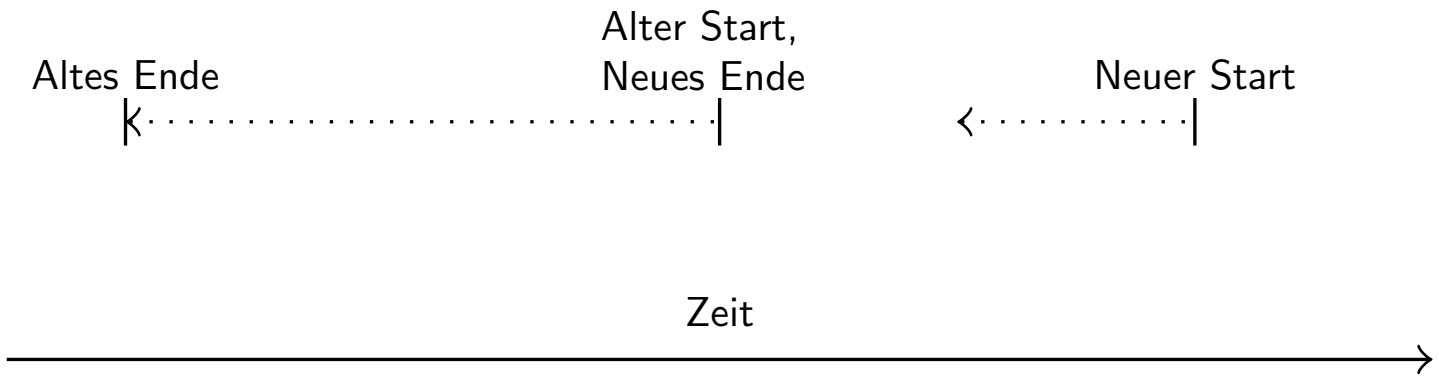
Module - Daemon - Suchstrategie



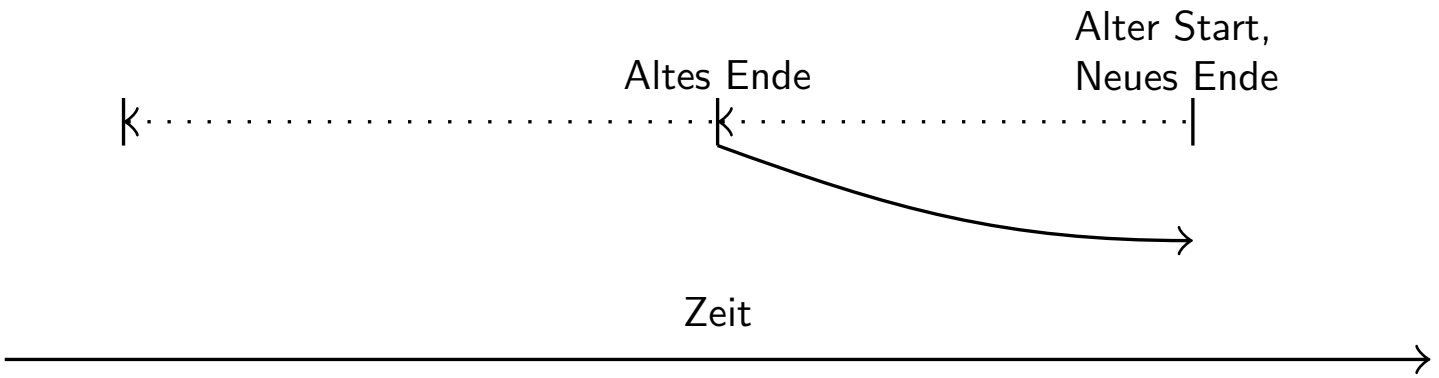
Module - Daemon - Suchstrategie



Module - Daemon - Suchstrategie



Module - Daemon - Suchstrategie



Module - Daemon - Parallele Suche

- Parallele Suche: mehr Tweets in kürzer Zeit finden
- Idee: mehrere Profile nutzen (Multi-Threading)
- Daemon als Master-Worker-Architektur realisieren

Module - Daemon - Scheduling der Suchbegriffe

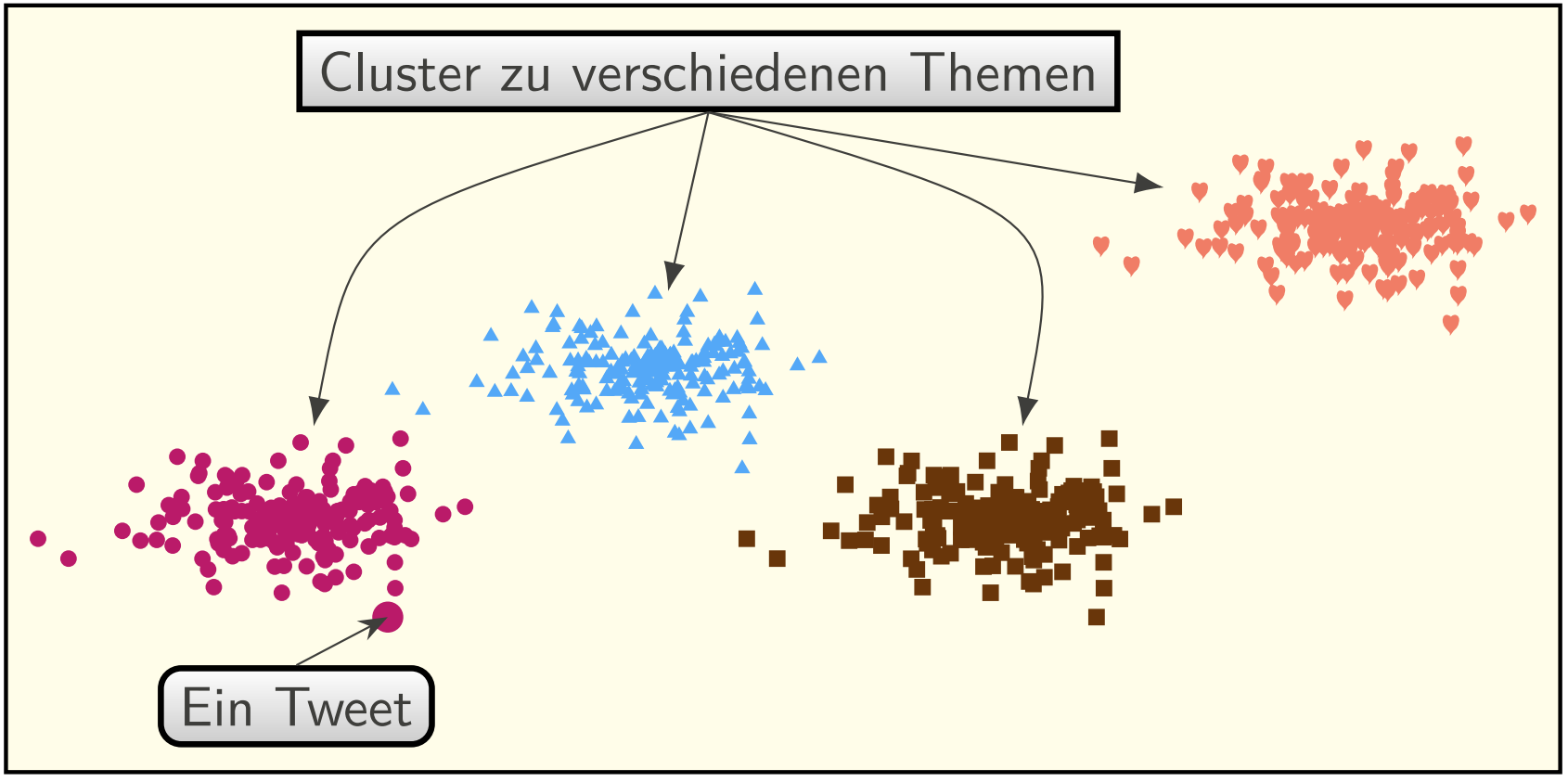
- Wie teilt man die Suchbegriffe auf?
- Short-Terms, kaum neue Tweets
- Long-Terms, viele neuen Tweets
- Worker erhält sowohl Short- als auch Long-Terms

Module - Daemon - Erfahrungen

- Multi-Threading ist komplex
- Probleme mit dem Speicherverbrauch der JVM
- Konsistenz der verschiedenen Teile komplex

Gliederung

Module - Clustering - Grundidee

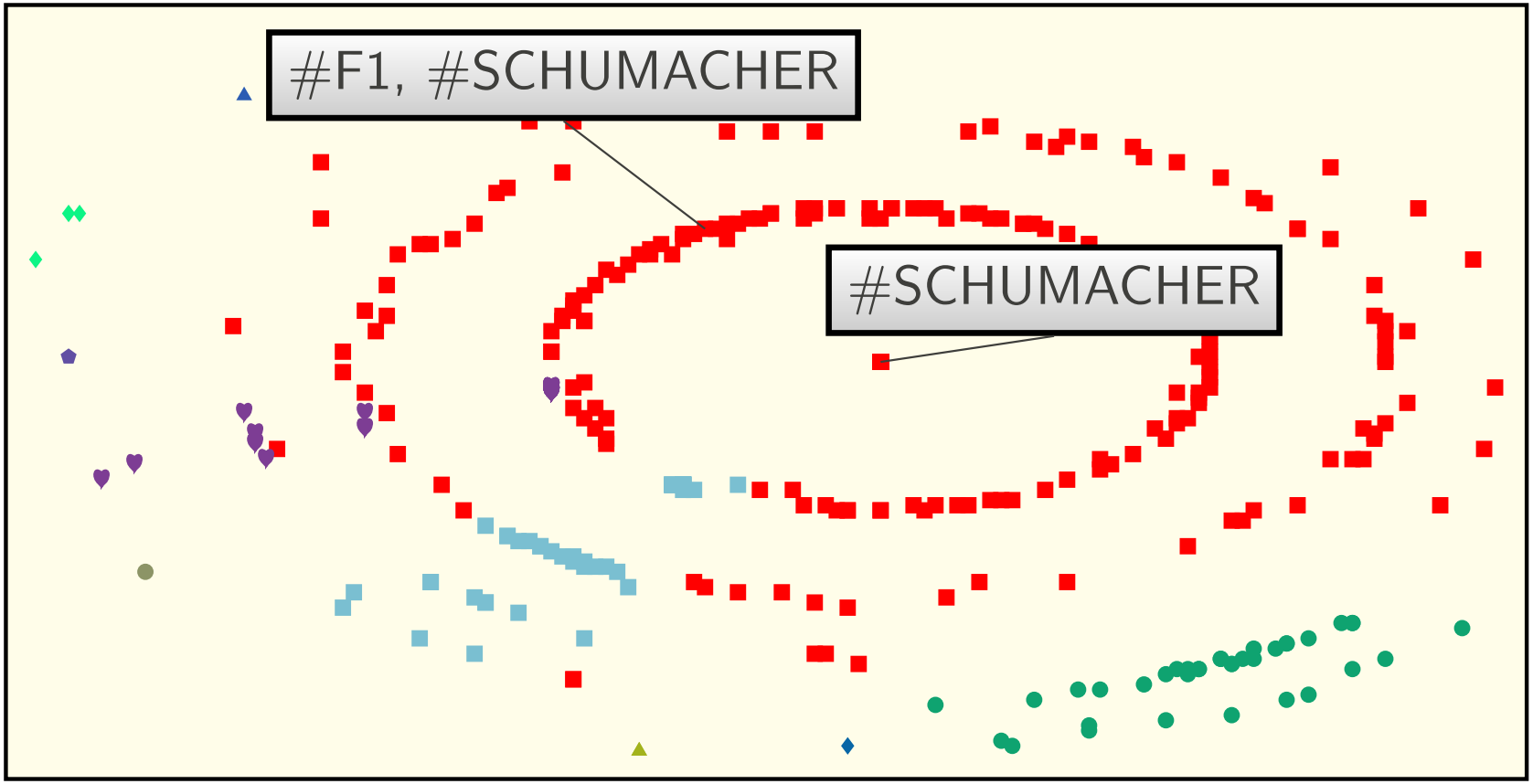


Module - Clustering - Umsetzungsidee

- Features für Tweets/Hashtags berechnen.
- Ähnlichkeiten zwischen allen Tweet/Hashtag-Paaren berechnen.
- Tweets clustern, um festzulegen welche Tweets zusammengehören.
- Tweets in der xy-Ebene Positionen zuordnen.

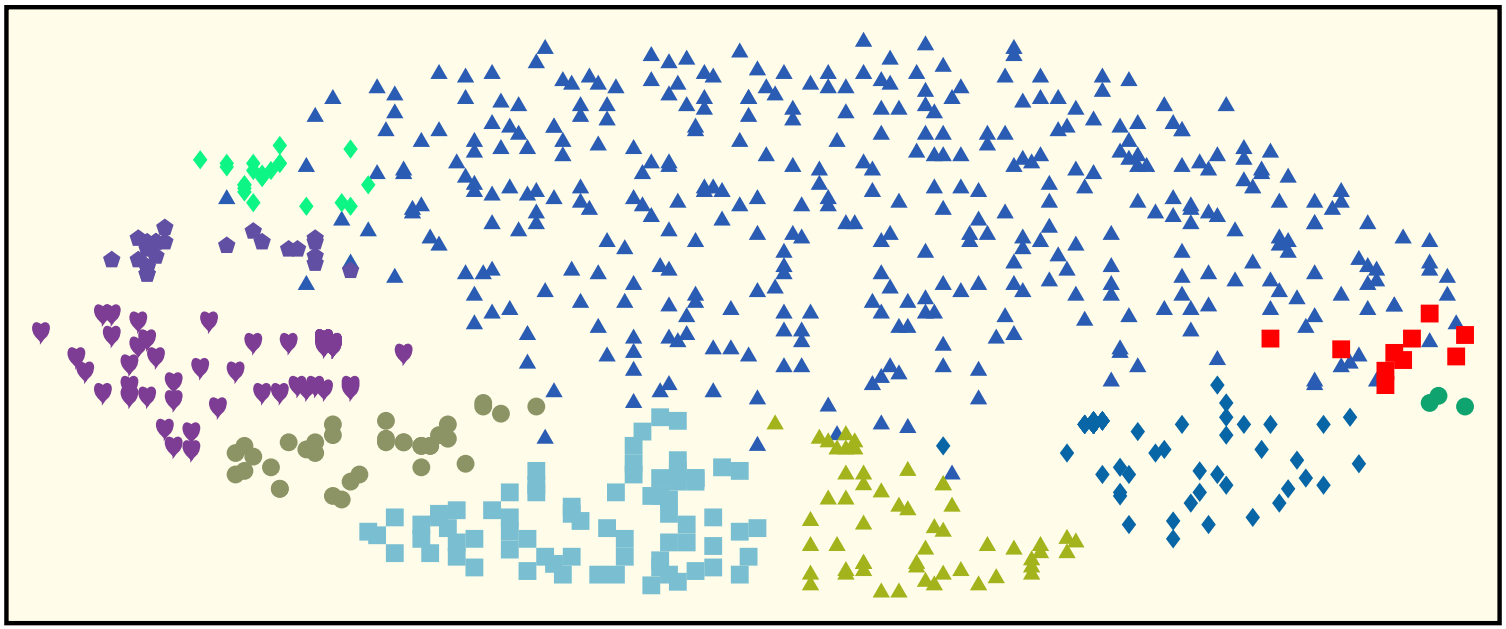
Module - Clustering - Umsetzungsidee

Ergebnis:



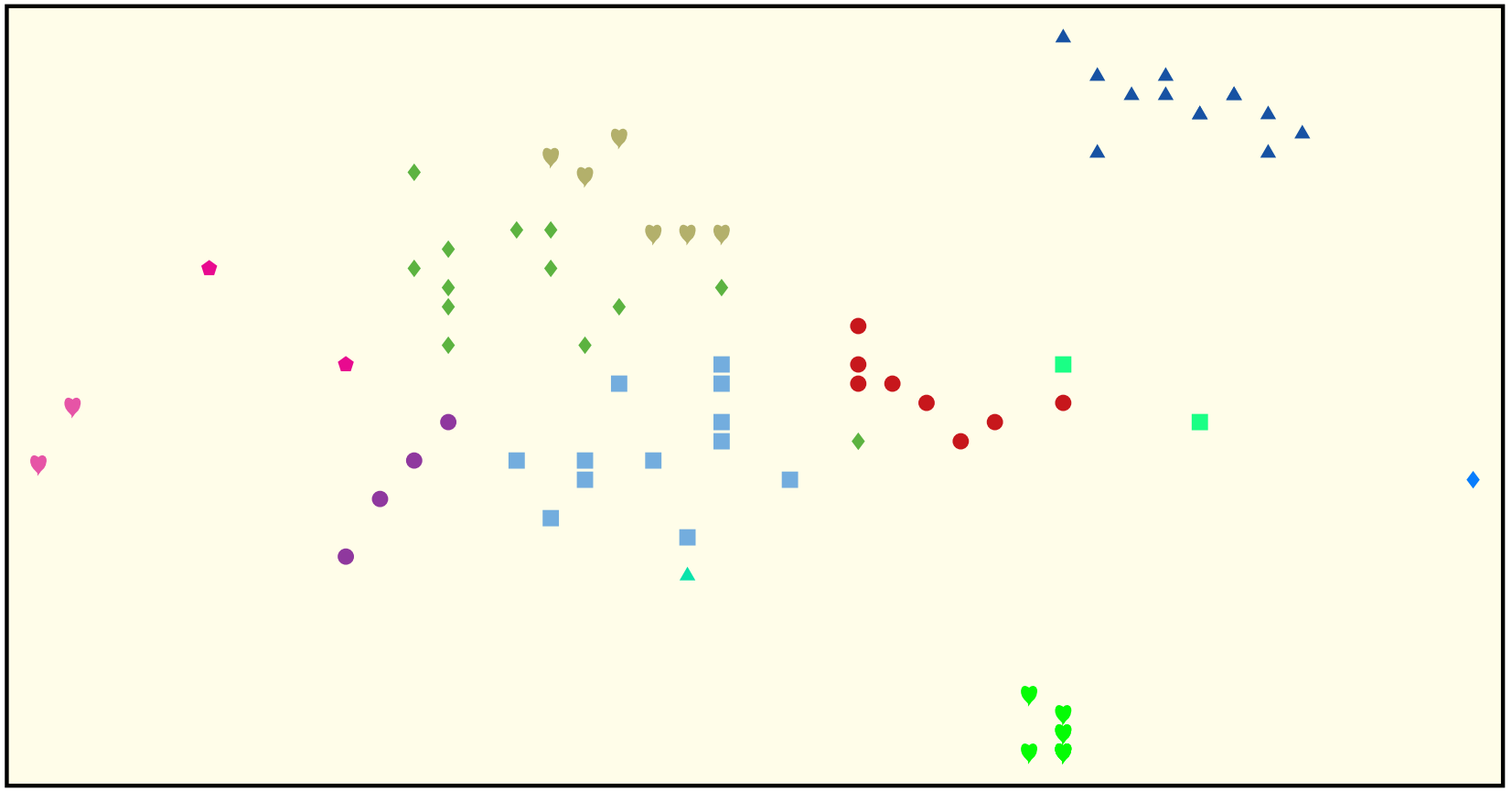
Module - Clustering - Alternative Umsetzungen

- Statt Hashtags alle vorkommenden Wörter in den Tweets verwenden.



Module - Clustering - Alternative Umsetzungen

- Statt Tweets Hashtags clustern und visuell darstellen.

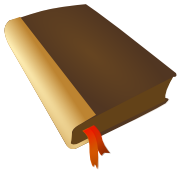


Gliederung

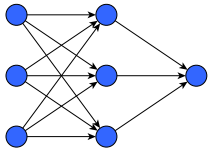
Module - Sentiment - Ansätze

Emoticon-Liste

Wörterbuch



Machine Learning



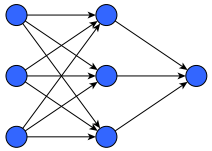
- I love puppies :-)

Module - Sentiment - Ansätze

Emoticon-Liste

Wörterbuch

Machine Learning

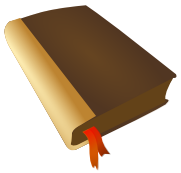


- I love puppies :-)
- I **love** puppies :-)

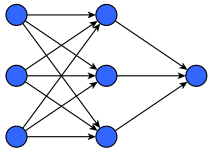
Module - Sentiment - Ansätze

Emoticon-Liste

Wörterbuch



Machine Learning



- I love puppies :-)
- I love puppies :-)
- I love puppies :-)

Module - Sentiment - Machine Learning mit Regression

- Trainingsdaten:
  - “I love puppies”
  - “I hate puppies”

Module - Sentiment - Machine Learning mit Regression

- Trainingsdaten:
  - “I love puppies”
  - “I hate puppies”
  - Merkmalsmatrix:

I	love	hate	puppies	Sentiment
1	1	0	1	$+1$
1	0	1	1	-1

Module - Sentiment - Machine Learning mit Regression

- Trainingsdaten:
  - "I love puppies"
  - "I hate puppies"
  - Merkmalsmatrix:

I	love	hate	puppies	Sentiment
1	1	0	1	+1
1	0	1	1	−1

- Regressionsmodell:

I	love	hate	puppies
0	+1	−1	0

Module - Sentiment - Machine Learning mit Regression

- Trainingsdaten:
  - "I love puppies"
  - "I hate puppies"
  - Merkmalsmatrix:

I	love	hate	puppies	Sentiment
1	1	0	1	$+1$
1	0	1	1	-1

- Regressionsmodell:

I	love	hate	puppies
0	$+1$	-1	0

- Neue Tweets: z. B. "I love kitties"

I	love	kitties

Module - Sentiment - Machine Learning mit Regression

- Trainingsdaten:
  - "I love puppies"
  - "I hate puppies"
  - Merkmalsmatrix:

I	love	hate	puppies	Sentiment
1	1	0	1	$+1$
1	0	1	1	-1

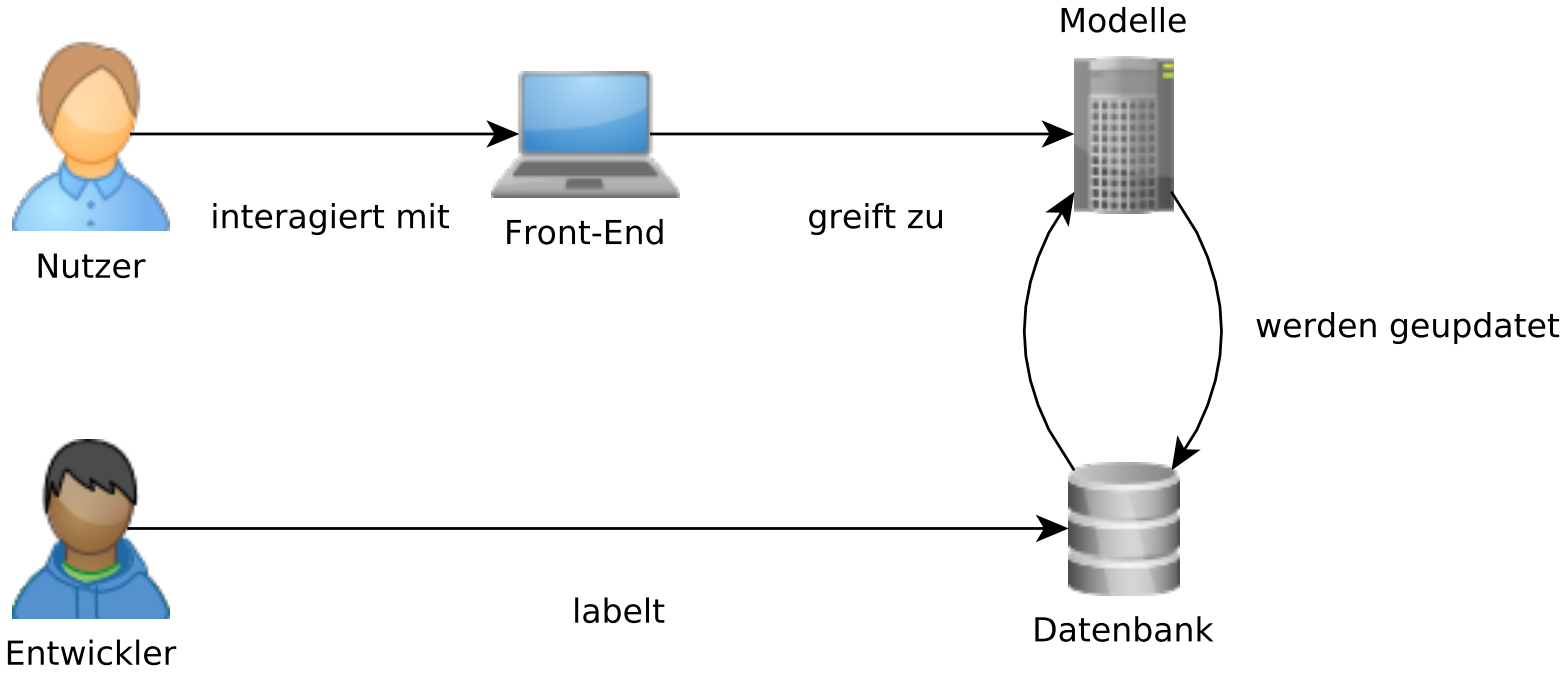
- Regressionsmodell:

I	love	hate	puppies
0	$+1$	-1	0

- Neue Tweets: z. B. "I love kitties"

I		love	kitties	
0	$+$	1		$= 1$

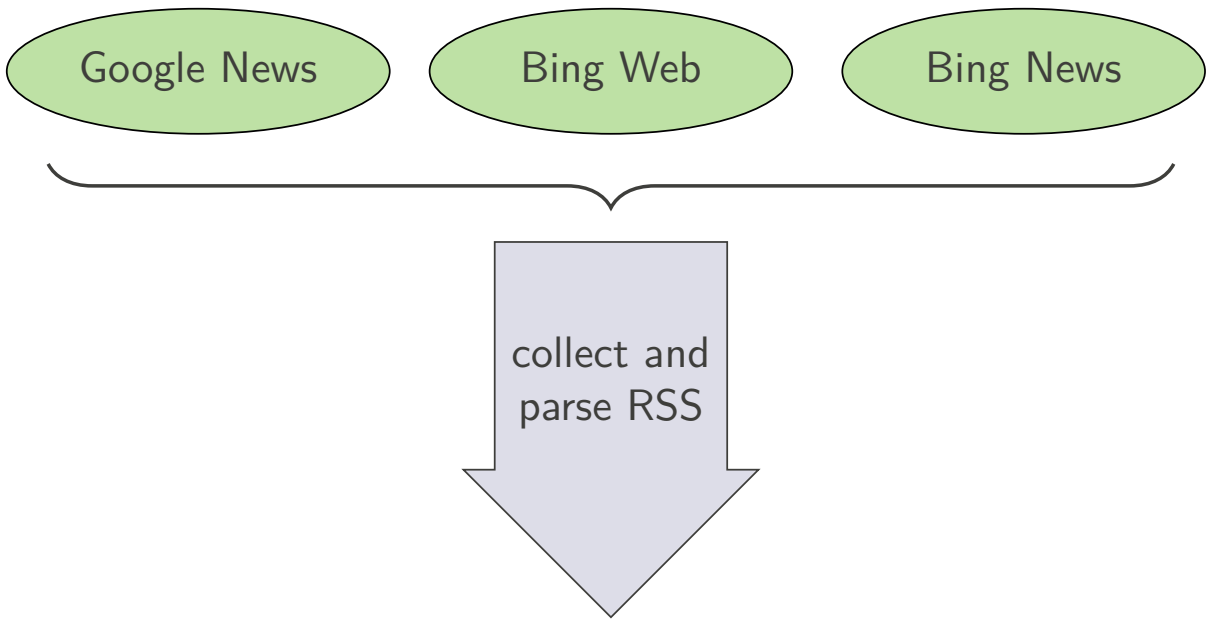
Module - Sentiment - Architektur



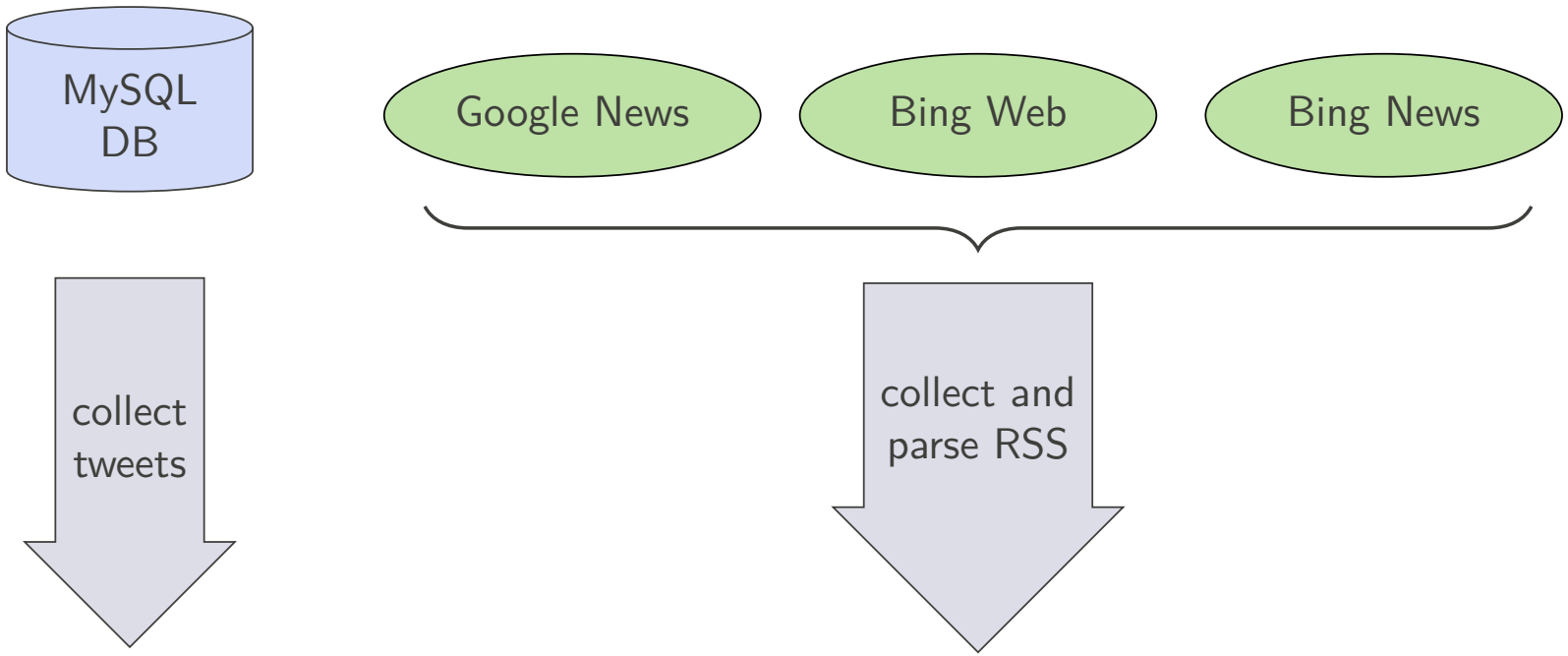
Gliederung

Module - News - Datenquelle

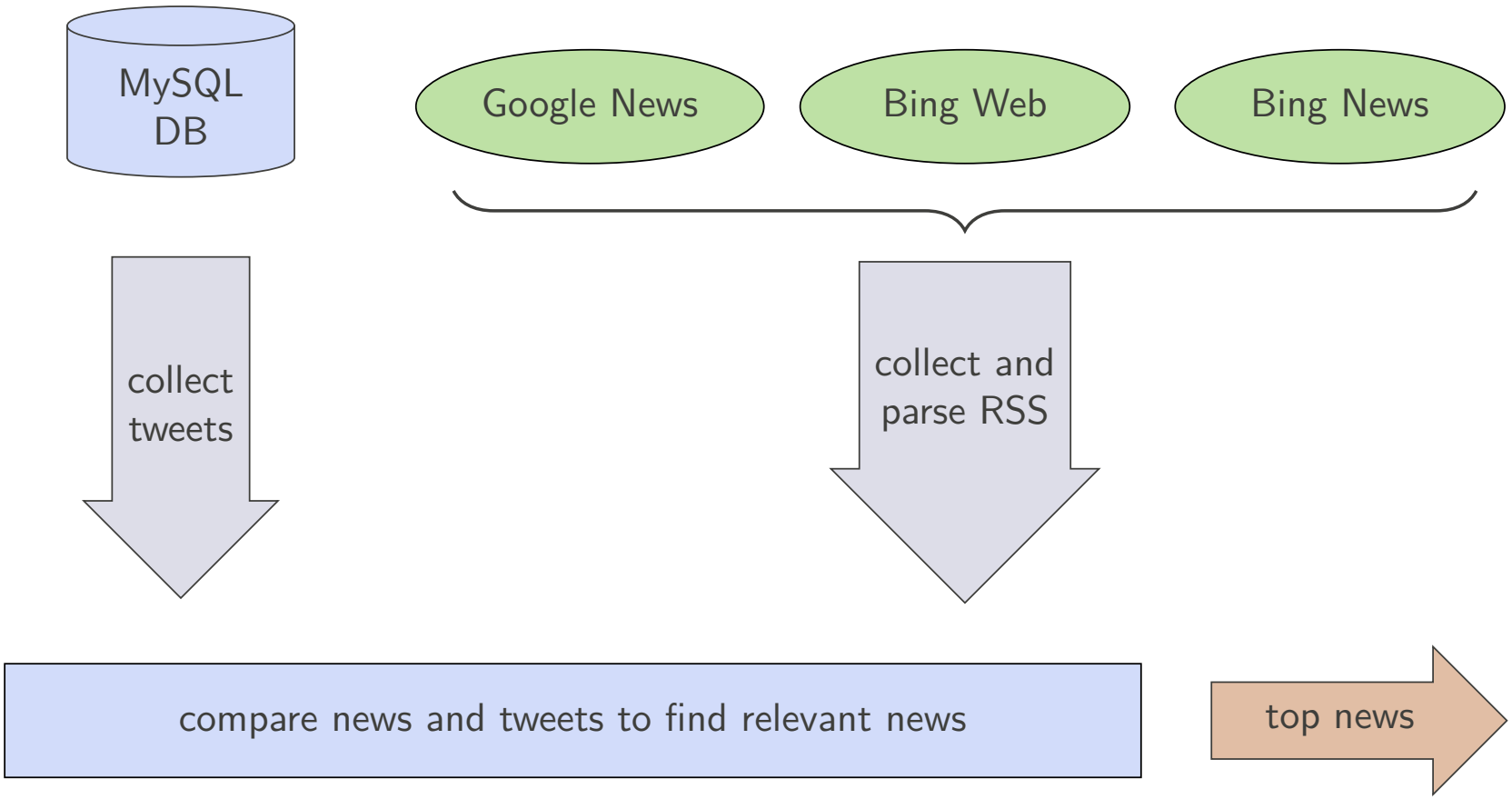
Module - News - Datenquelle



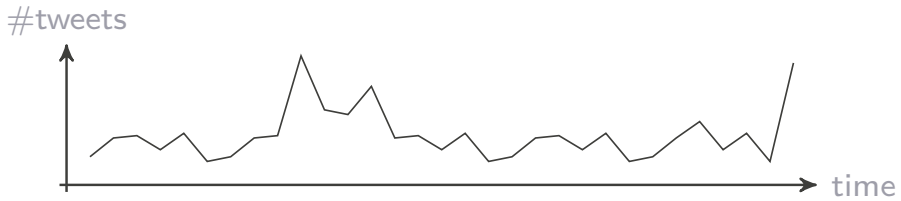
Module - News - Datenquelle



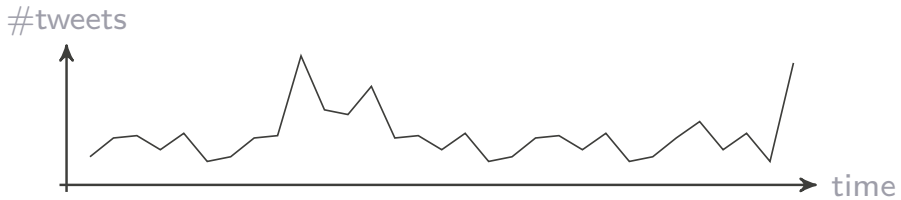
Module - News - Datenquelle

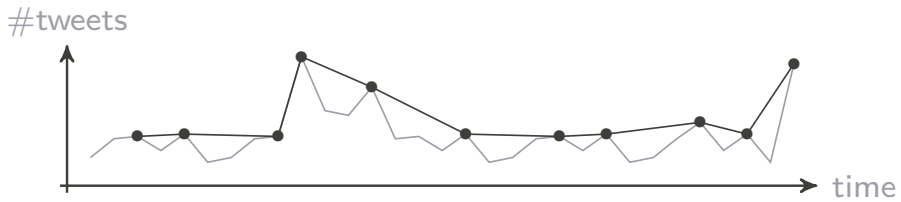


Module - News - Peaksuche

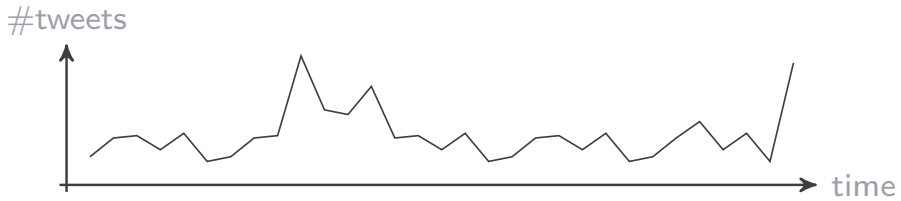


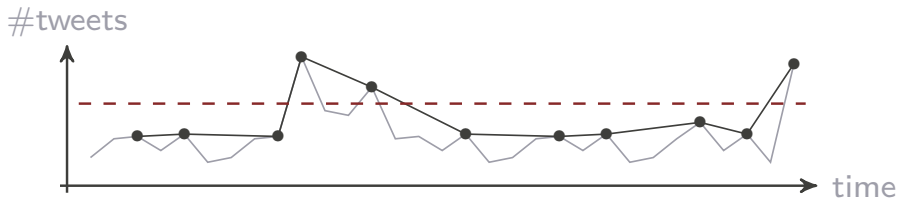
Module - News - Peaksuche



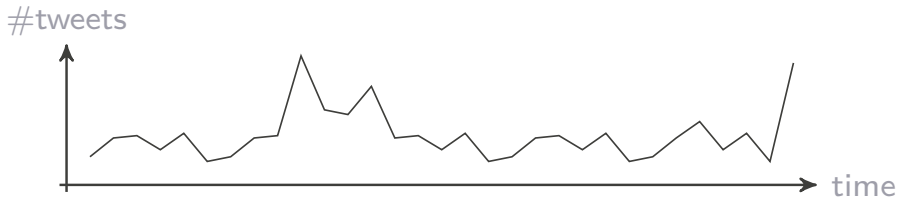


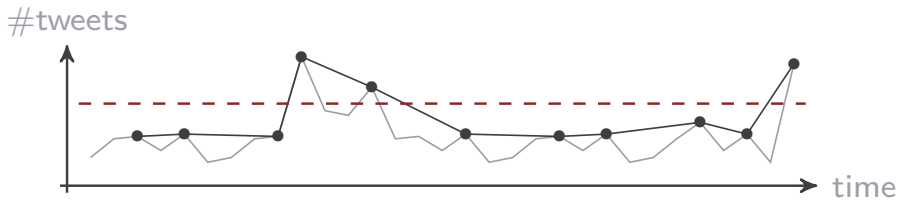
Module - News - Peaksuche

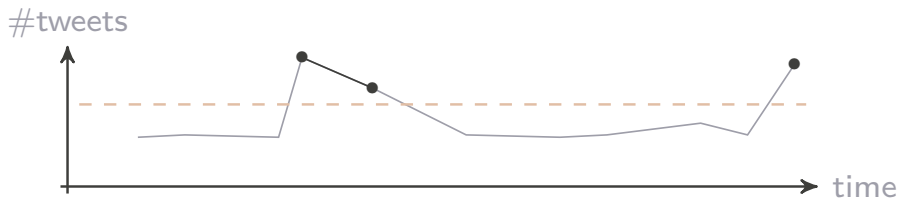




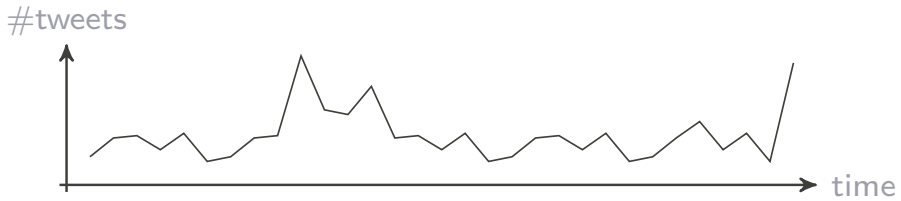
Module - News - Peaksuche

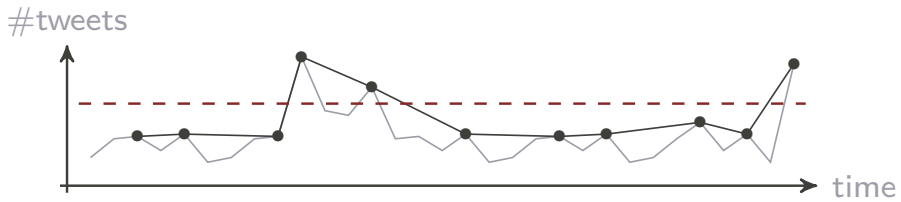


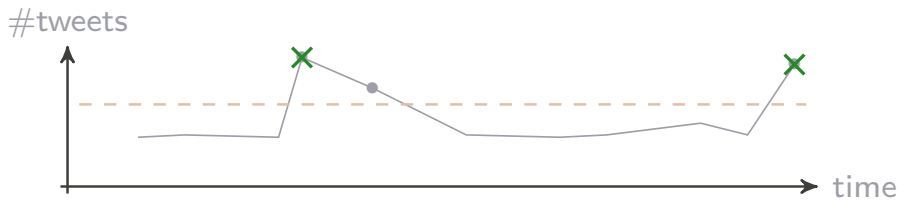




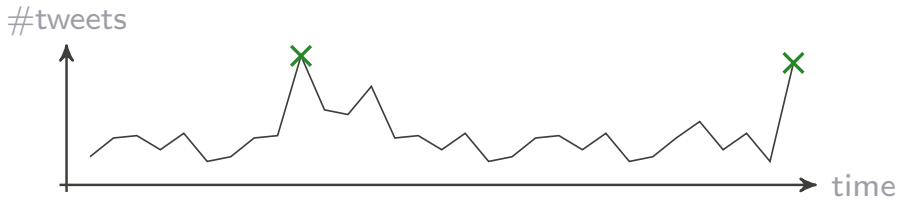
Module - News - Peaksuche

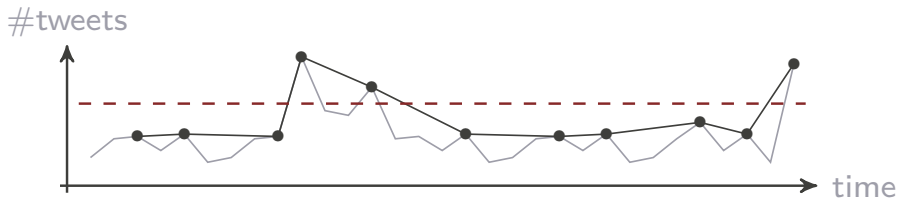


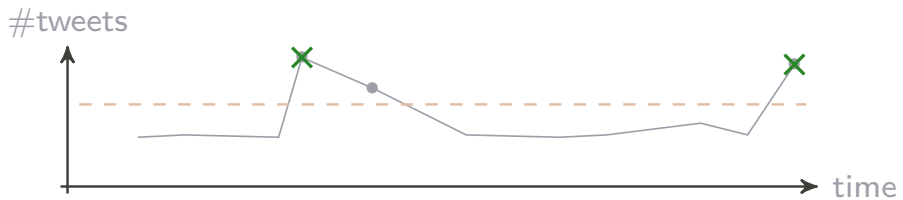




Module - News - Peaksuche







Gliederung

Ausblick

Was wäre noch möglich gewesen? Was wurde nicht umgesetzt?

- Kino-Modul
- Heatmap über Landkarte
- Prognosemöglichkeiten
- weitere Performance-Optimierungen
- ...

Vorgehen: Scrum ... but

Scrum:

- Planning Poker, Definition of Done
- Priorisierung durch Kunden
- Kundentreffen
- selbstorganisierendes Team

But:

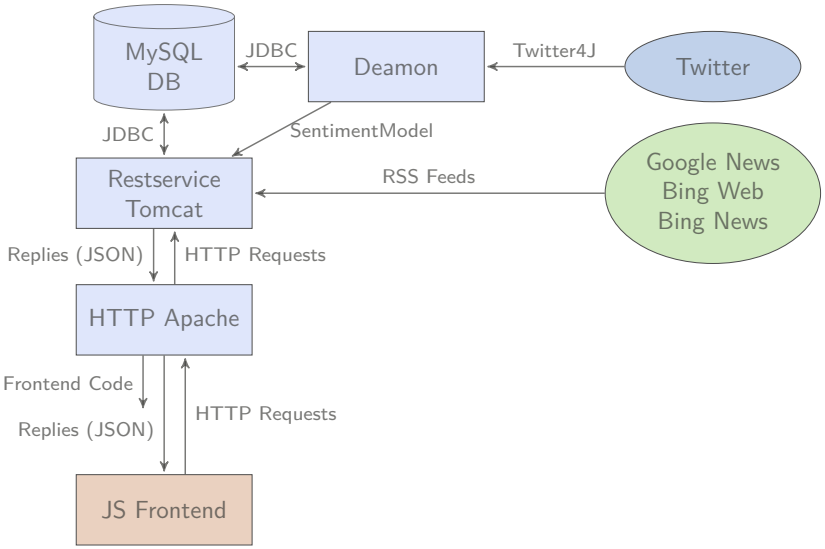
- Daily Scrum
- wöchentlich wechselnde Scrum Master
- Feature und Code Freeze

Vielen Dank für Ihre Aufmerksamkeit!

TMetrics

Datenanalyse auf **Twitterbeiträgen** zur Erkennung und visuellen Darstellung von Meinungstrends

- **Machine Learning**
- **Clustering**
- **Sentiment-Analyse**



Gliederung