Thema 2.3: Apache Spark

Skalierbare verteilte Datenanalyse

Lukas Wappler

15. Mai 2017

Inhaltsverzeichnis

"Statistical thinking will one day be as necessary for efficient citizenship as the ability to read and write."

H.G. WELLS (1866-1946)
Science-Fiction-Roman-Autor
Der Krieg der Welten

Einleitung

Heutige Probleme:

- Immer mehr Daten
- Datenanalysen werden immer schneller benötigt
- Systeme sind nicht skalierbar
- Großrechner sind teuer

Einleitung

Heutige Probleme:

- Immer mehr Daten
- Datenanalysen werden immer schneller benötigt
- Systeme sind nicht skalierbar
- Großrechner sind teuer

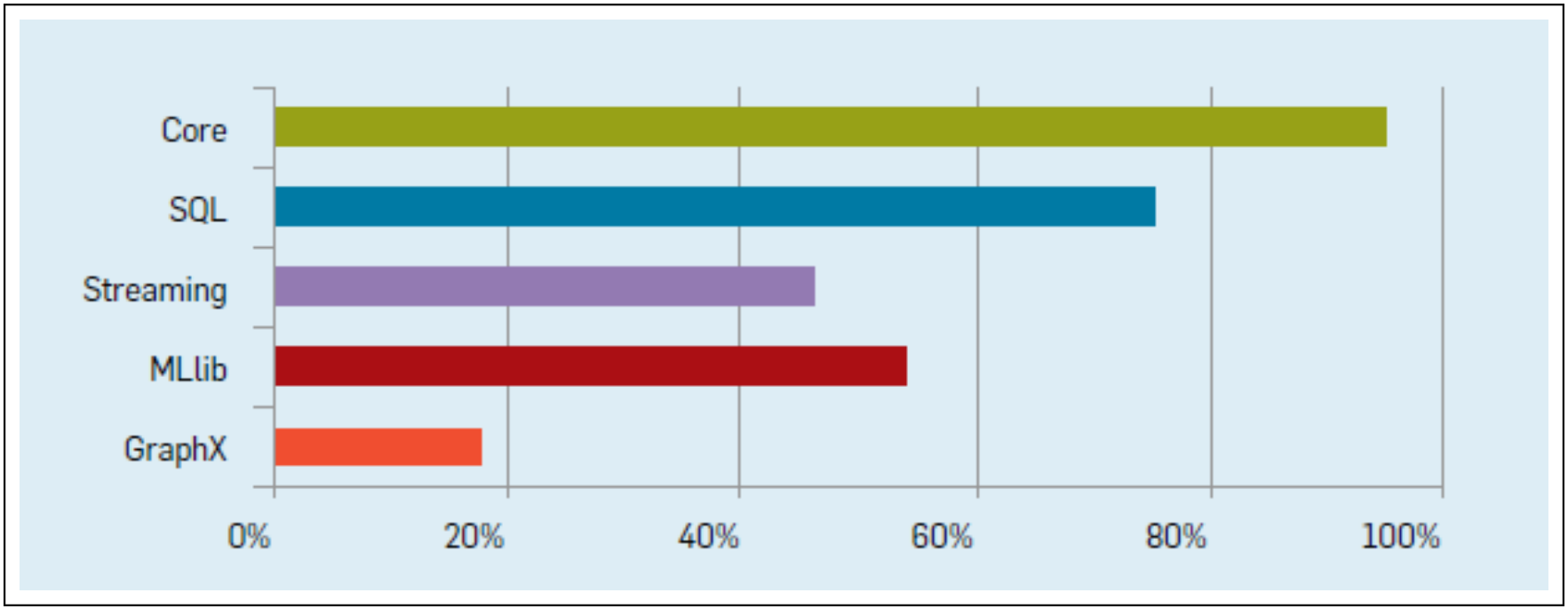
Ist Apache Spark die Lösung?

Apache Spark

- Cluster-System
- 2009 im AMPLab ins Leben gerufen
- 2010 Open Source
- 2013 von der Apache Software Foundation übernommen
- 2014 Top Level Projekt

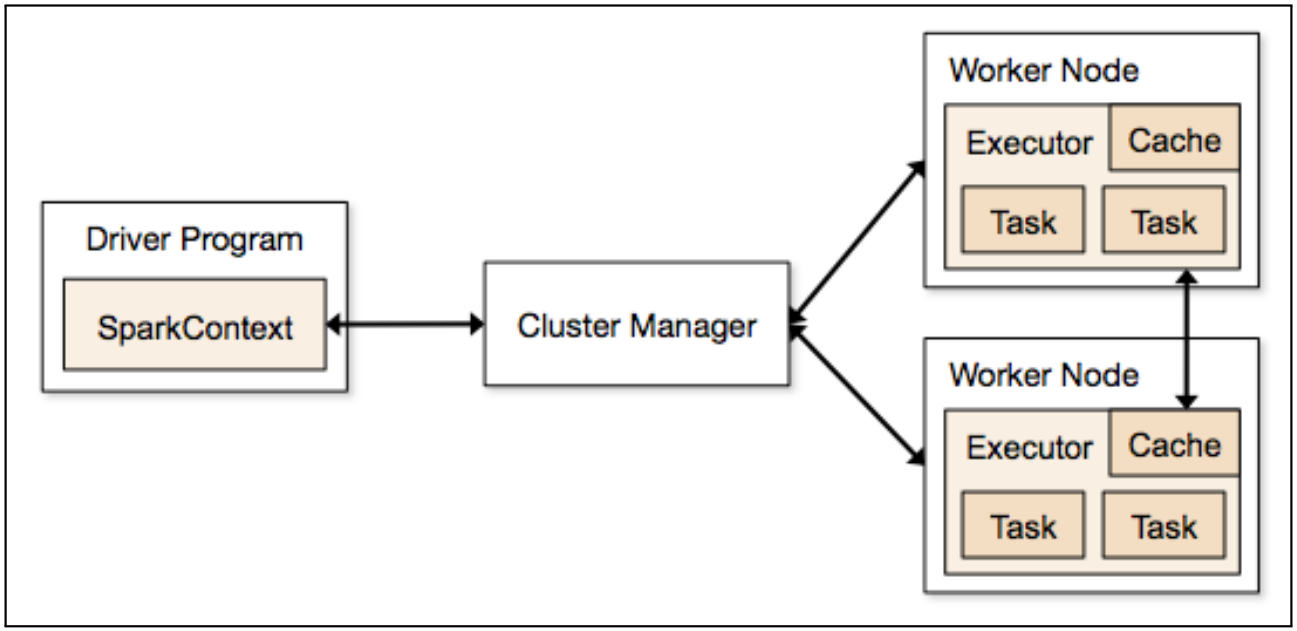
Kern-Bibliotheken / Komponenten

Nutzung der Komponenten



Spark-Core

Aufbau & Architektur



RDD's

Resilient Distributed Datasets:

- belastbar (fehlertolerant / ausfallsicher)
- verteilt (Daten sind in Blöcke unterteilt)
- nach Erstellung nur lesbar

Operationen

- Transformationen (filter, join, ...)
- Aktionen (reduce, count, ...)

Spark-SQL & Dataframes

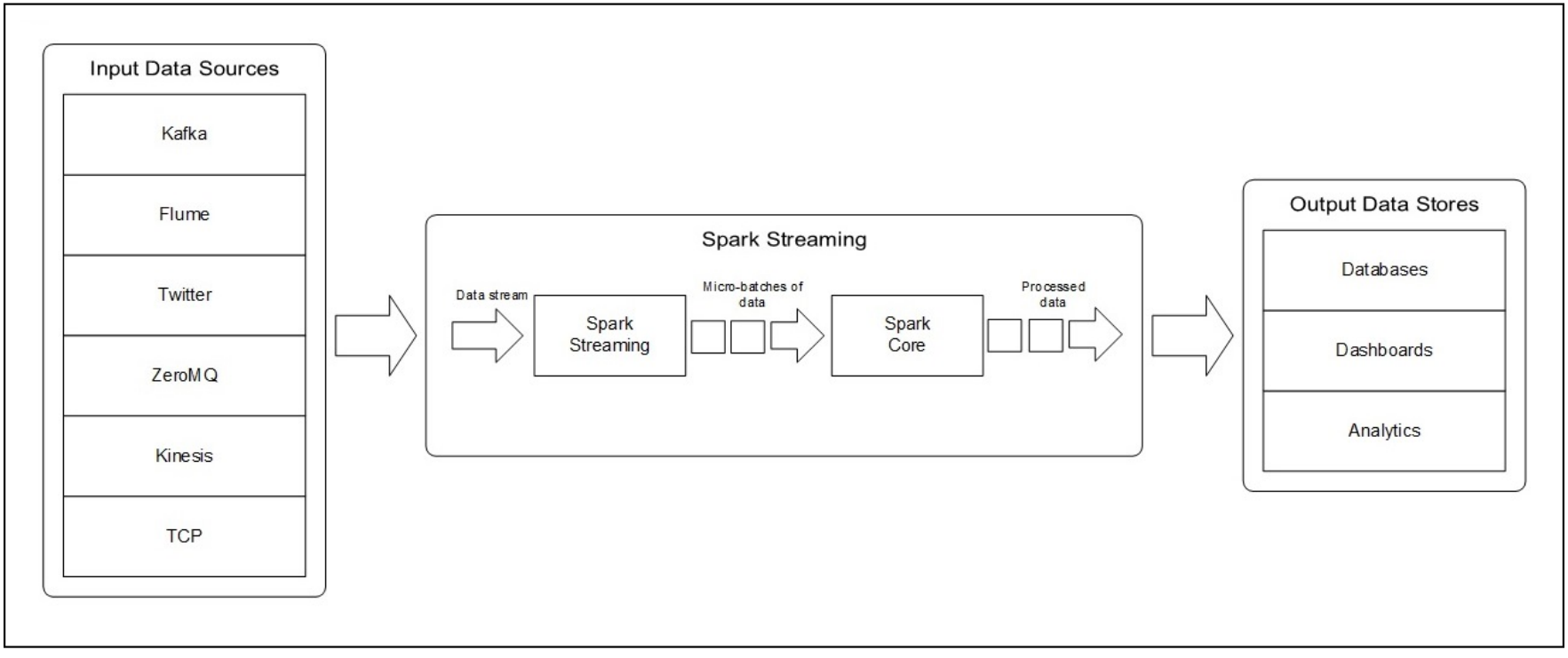
- eine Verbesserung von Shark
- kombiniert prozedurale Algorithmen & relationale Datenbankabfragen
- neben RDD's werden auch DataFrames verwendet.
- Anbindung unterschiedlichster Datenquellen:
  - JSON
  - JDBC
  - Hive
  - ...

Verarbeitung von Datenströmen (Spark-Streaming)

- RDD's werden zu DStreams erweitert
- Daten werden in einzelne Pakete unterteilt
- Transformationen können ausgeführt werden

Verarbeitung von Datenströmen (Spark-Streaming)

- RDD's werden zu DStreams erweitert
- Daten werden in einzelne Pakete unterteilt
- Transformationen können ausgeführt werden

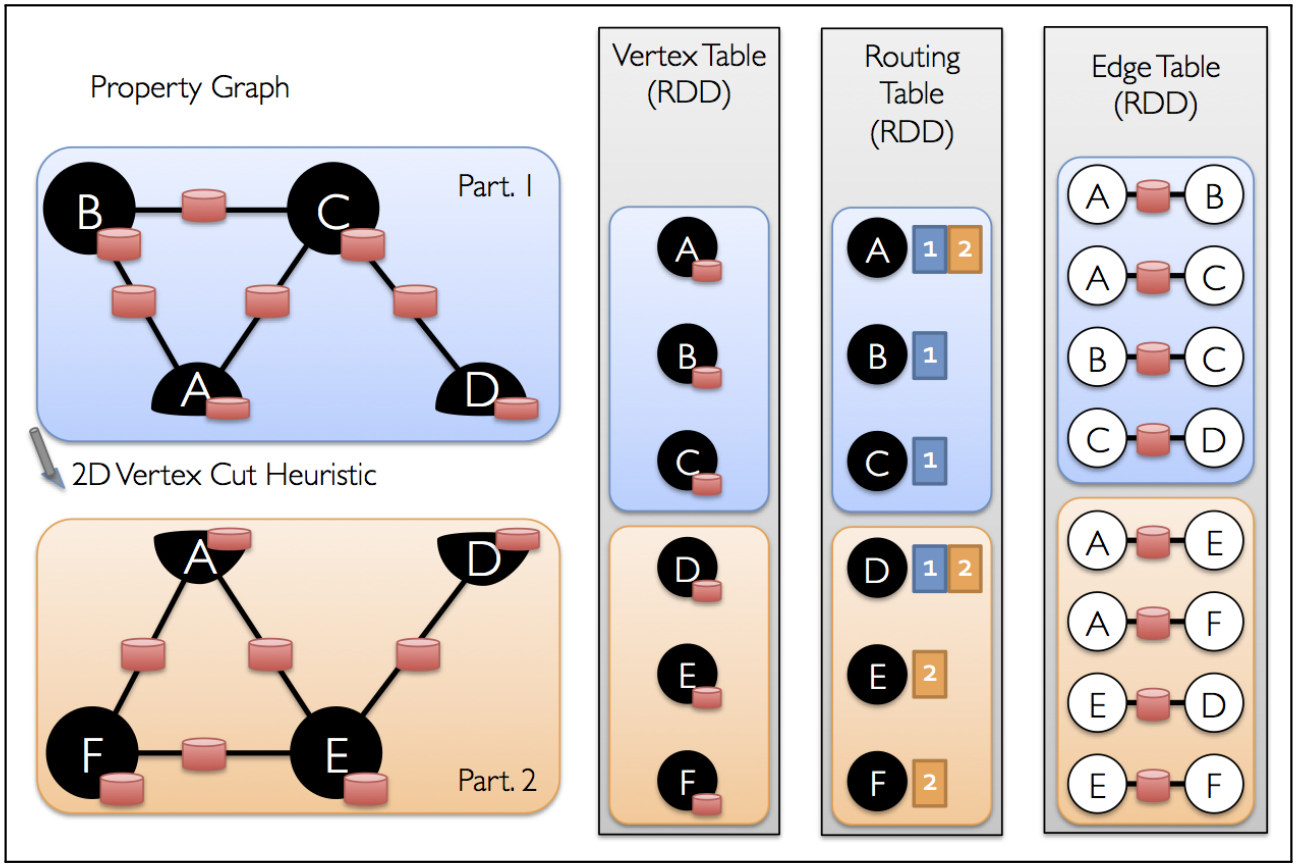


Berechnungen auf Graphen (GraphX)

- Property-Graphen
- Abbildung der Graphen in RDD-Tupeln
- 1. RDD enthält Ecken
- 2. RDD enthält Kanten

Berechnungen auf Graphen (GraphX)

- Property-Graphen
- Abbildung der Graphen in RDD-Tupeln
- 1. RDD enthält Ecken
- 2. RDD enthält Kanten

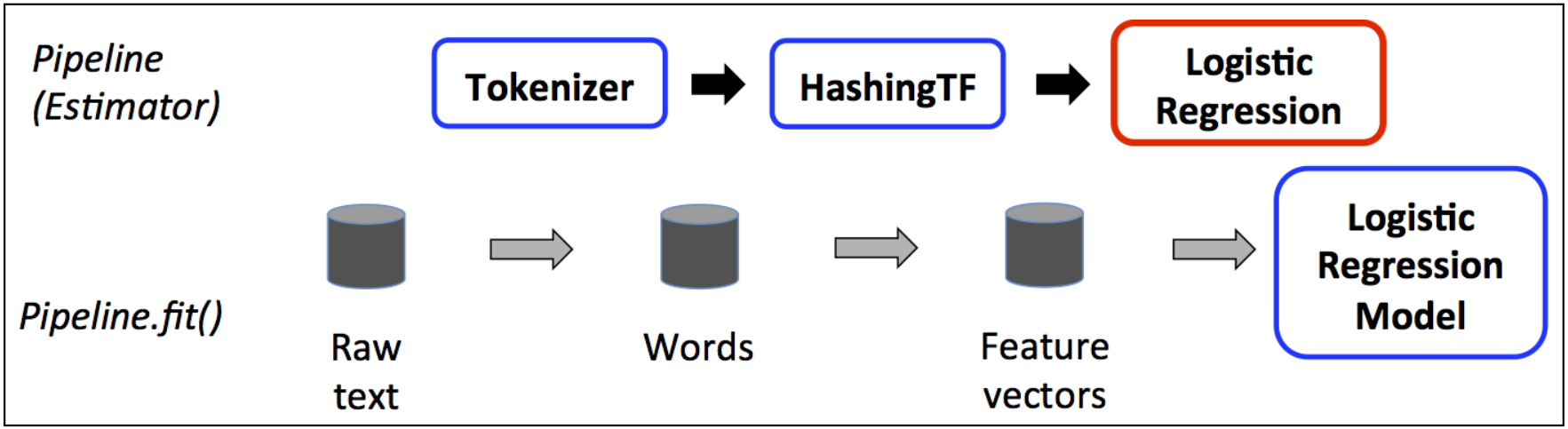


Maschinelles Lernen (MLlib)

- Nutzt DataFrames
- Transformator (verändert die Daten)
- Estimator (Abstraktionen des Lernalgorithmus)
- Pipeline

Maschinelles Lernen (MLlib)

- Nutzt DataFrames
- Transformator (verändert die Daten)
- Estimator (Abstraktionen des Lernalgorithmus)
- Pipeline



Skalierung von R Programmen (SparkR)

Was ist R?

- freie Programmiersprache
- statistische Berechnungen & Grafiken
- R läuft nur in einem Thread

Wie wird das Problem gelöst:

- R-JVM Brücke von R zu Java
- Kommunikation über Sockets

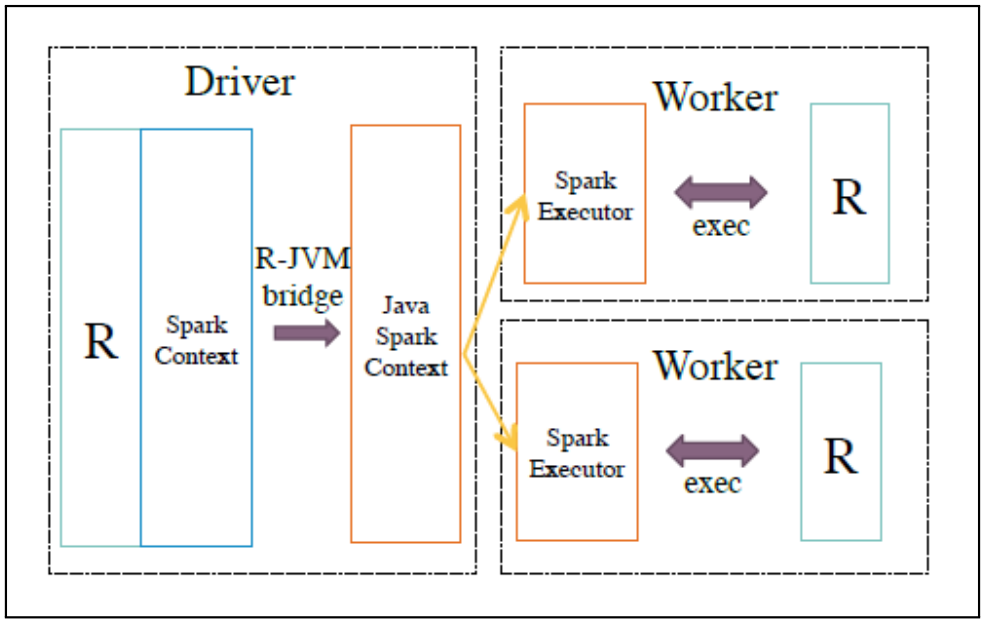
Skalierung von R Programmen (SparkR)

Was ist R?

- freie Programmiersprache
- statistische Berechnungen & Grafiken
- R läuft nur in einem Thread

Wie wird das Problem gelöst:

- R-JVM Brücke von R zu Java
- Kommunikation über Sockets



Demonstration

Apache Spark im Einsatz

(Demo: Live oder Video.)

Mehrere Komponenten im Verbund

SparkCore, SparkSQL und MLlib im Verbund:

- 100.000 Mitarbeiter Datensätze
- JSON-einlesen
- Suchbegriffe trainieren
- Vorhersagen treffen
- Daten reduzieren
- Sortieren
- Ausgabe der Ergebnisse

Performance

- webbasierte Übersicht über Cluster
- schwer zu messen
- Fehler sind schwer zu lokalisieren
- Seiteneffekte bei verteilten Operationen

Besonderheiten bei der Speichernutzung

- Nutzung des Arbeitsspeichers
- speicherplatzeffiziente Datenstrukturen
- komprimierte Daten können Blockgrößen überschreiten

Netzwerk und I/O-Traffic

Analysen mit:

- 8.000 Nodes
- 1PB Daten

Netzwerk und I/O-Traffic

Analysen mit:

- 8.000 Nodes
- 1PB Daten

Optimierungen:

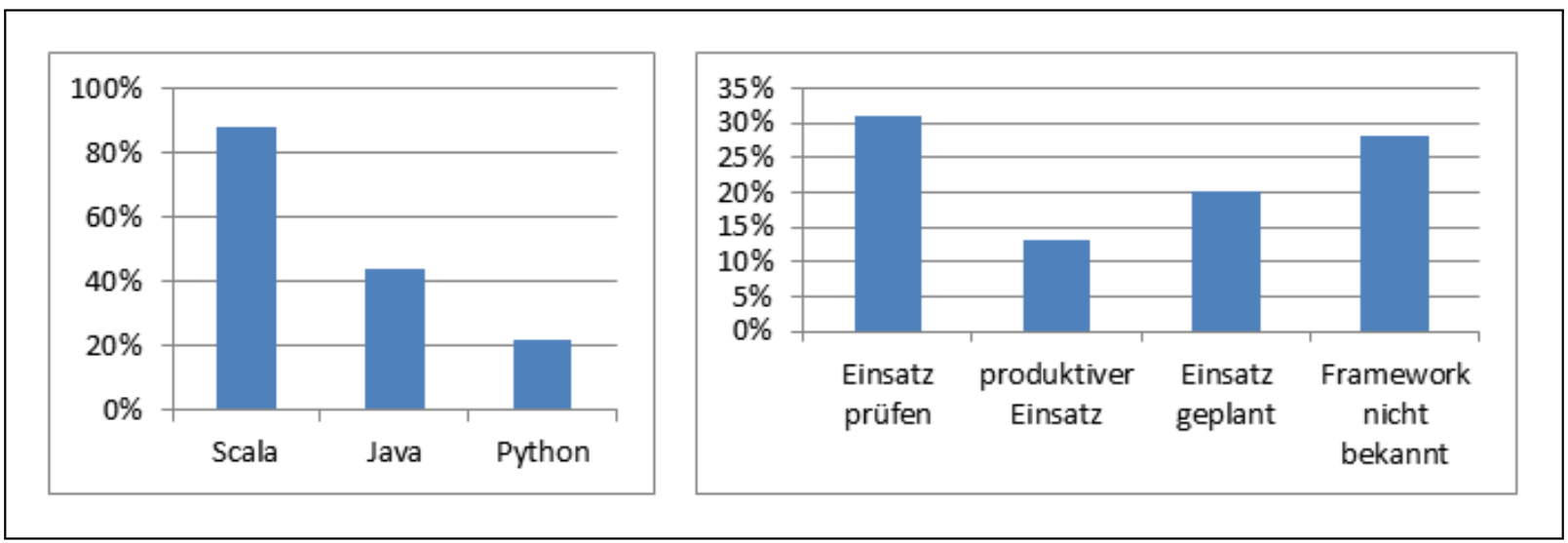
- Daten von Festplatte zum Socket direkt kopieren
- Speichertabellen außerhalb des Java Heaps verwalten
- parallele Verarbeitung über mehrere Verbindungen

Nutzung und Verbreitung

- Skala, Python, Java, (R)
- viele Datenquellen
- viele Dateiformate
- über 400 Mitwirkende (git contributors)
- aus über 100 Unternehmen
- Konferenzen (Spark Summit)
- 12,779 Sterne (Github)

Nutzung und Verbreitung

- Skala, Python, Java, (R)
- viele Datenquellen
- viele Dateiformate
- über 400 Mitwirkende (git contributors)
- aus über 100 Unternehmen
- Konferenzen (Spark Summit)
- 12,779 Sterne (Github)



Fazit

Vorteile:

- vielfältig einsetzbar
- kann mit Speziallösungen mithalten
- gute Dokumentation & Literatur
- kostengünstig
- flexibel

Nachteile:

- Mischung verschiedener Datenstrukturen schwierig
- veraltete News, Blogs oder Foren-Beiträge

Ausblick und Weiterentwicklung

- Code ist über Github verfügbar
- ständige Weiterentwicklung
- jeder kann daran mitarbeiten
- 51 Releases (bzw. RC's)
- über 19.000 commits
- immer wieder Performancesteigerungen

Vielen Dank

Vielen Dank für Ihre Aufmerksamkeit!