18. Oktober 2010

(Advanced) Cloud Computing

Teamprojekt & Projekt

Veranstalter: Prof. Dr. Georg Lausen

Betreuer: Thomas Hordnung,
Alexander Schätzle,
Martin Przjyaciel-Zablocki

dbis

Anforderungen

- **Studienordnung**
  - **Master:** 16 ECTS
    → 480 Semesterstunden ~ 34h/Woche p.P.
  - **Bachelor:** 6 ECTS
    → 180 Semesterstunden ~ 13h/Woche p.P

  - Teamgröße: 3-4 Studenten
  - Projektbericht: ca. 15-25 Seiten p.P.
  - Abschlusspräsentation: ca. 15min p.P.

  - Arbeitsleistung einzelner Teilnehmer muss klar voneinander abzugrenzen sein

Organisation

- **Zeit und Ort:**
  - Montag 14–17 Uhr (c.t.)
  - Raum: SR 00 007 (MMR), Geb. 106
- **Nächstes Treffen:**
  - Dienstag, 2. November 2010 14–17 Uhr (c.t.)
  - Raum: 01–029, Geb. 51
- **Weiterer Ablauf:**
  - Treffen mit Kurzpräsentationen aller Teams
  - Weitere individuelle Termine auf Anfrage

Ziele

- Gemeinsame Arbeit an einem großen Projekt
- Eigenständiges Recherchieren und Arbeiten
- Verbesserung der individuellen Programmierfähigkeiten (hier: Java)
- Einarbeiten in neue Themen (hier: RDF und MapReduce)
- Probleme bei größeren Projekten Kennen und Lösen lernen
- Erfahrungen im Umgang mit MapReduce sammeln

Schlüsselfaktoren zum Erfolg

- Gute Team-interne Organisation
  - Aufteilung von Verantwortung
  - Erfüllen von Verantwortung
  - Einhalten von Fristen
  - Gegenseitige Hilfe und Unterstützung
  - Saubere Definition von Schnittstellen
- Nutzen entsprechender Software (z.B. SVN)
- Einbringen individueller Fähigkeiten
- Spezialisierung auf Teilgebiete, ohne den Blick für das Ganze zu verlieren

Benotungsgrundlage

- Insbesondere (aber nicht ausschließlich)
  - Umfang und Schwierigkeit der geleisteten Arbeit/Implementierung
  - Teamleistung: ein gelungenes Projekt wirkt sich in der Regel positiv auf die Noten einzelner Teammitglieder aus
  - Rolle und Mitarbeit im Team (Koordination etc.)
  - Qualität des Codes (Formatierung, Dokumentation)
  - Individuelle Ausarbeitung (Projektbericht)
  - Mündlicher Vortrag (Abschlusspräsentation)

Projektablauf

- **Einarbeitungsphase**
  - Bis Dienstag, 2. November 2010
  - Endgültige Themenvergabe
- **Kurzpräsentation**
  - 8. November 2010
  - Projektvorstellung
  - Eigene Milestones
  - Interne Arbeitsaufteilung
- **Implementierungsphase**
  - Programmierung & Dokumentation
  - 10. / 17. Januar 2011: Zwischenbericht zu den Milestones (Treffen oder Präsentation)
- **Abschlusspräsentation**
  - 7. Februar 2011
  - Abgabe Projektbericht (14. Februar 2011)

Aufgabenstellung

- **Teamprojekt (Master)**
  - Entwurf und Implementierung eines verteilten RDF-Stores auf Basis von Hadoop (MapReduce)
- **Projekt (Bachelor)**
  - Implementierung eines Graphenproblems auf Basis von Hadoop (MapReduce)
  - Beispiel: Die Suche nach kürzesten Pfaden innerhalb eines RDF-Graphen

2. MapReduce

- Principles & Basic Concepts

MapReduce

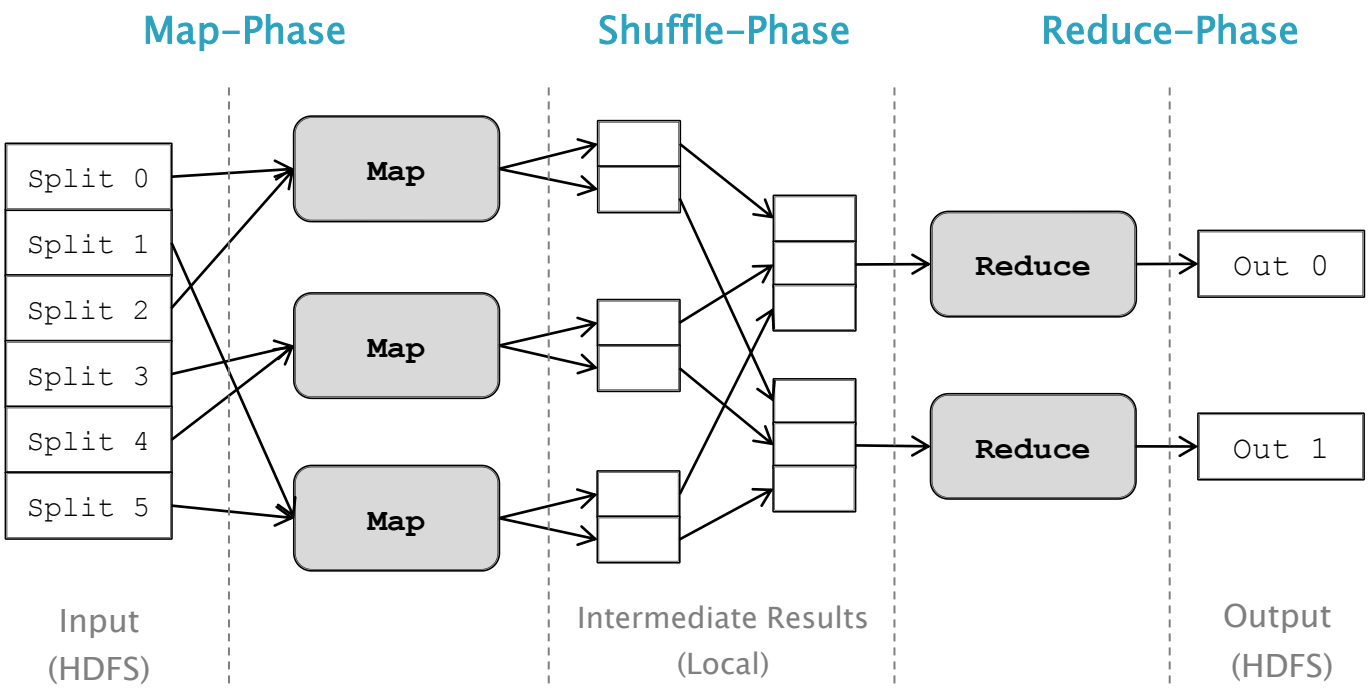
- **Google's MapReduce**
  - Automatic parallelization of computations
  - Fix and simple level of abstraction: Map & Reduce
- **Distributed File System**
  - Clusters of commodity hardware
    → Fault tolerance by replication
  - Very large files / write-once, read-many-times
- **Hadoop**
  - Open Source implementation (Apache project)
  - Used by Yahoo, Facebook, Amazon, IBM, Last.fm, …

MapReduce (2)

- **Map**
  - `map(in_key, in_value) -> (out_key, intermediate_value) list`
  - inputs: records from data source as `(key, value)` pairs, e.g. `(filename, line)`
  - outputs: one or more intermediate values with an output key

- **Reduce**
  - `reduce(out_key, intermediate_value list) -> out_value list`
  - After map phase all intermediate values for **one** output key are combined together in a list
  - reduce combines those intermediate values into one or more final values for the **same** output key

MapReduce (3)


Map-Phase
Shuffle-Phase
Reduce-Phase
Split 0
Split 1
Split 2
Split 3
Split 4
Split 5
Map
Map
Map
Reduce
Reduce
Out 0
Out 1
Input
(HDFS)
Intermediate Results
(Local)
Output
(HDFS)

Empfehlungen

- Hadoop Training
  - http://www.cloudera.com/hadoop/
- Cloudera's Distribution For Hadoop
  - Virtual Machine (→VMware Image)
  - http://www.cloudera.com/downloads/
  - CDH 3 (beta)
- Buch: "Hadoop: The Definitive Guide" von Tom White

3. RDF

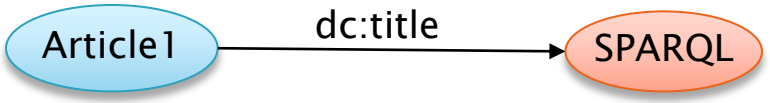
- Principles & Basic Concepts

Resource Description Framework (RDF)

- Datenformat
- Grundpfeiler des Semantic Web
- Sprache um Aussagen und Informationen über Ressourcen formal zu beschreiben
- Vom W3C standardisiert
  - ausführliche Informationen unter http://www.w3.org/RDF
- Datenformat basiert auf Tripeln der Form
  - (Subjekt, Prädikat, Objekt)

RDF Triple

- Jedes Tripel repräsentiert einen Wissensfakt, z.B.
  - (Article1, dc:title, „SPARQL“)
- **„Subjekt“** hat eine Eigenschaft **„Prädikat“** mit dem Wert **„Objekt“**
  - „Article1“ hat den „Titel“ „SPARQL“
- Darstellung eines Tripels als eine gerichtete Kante in einem Graphen möglich
  - Subjekt, Objekt → Knoten
  - Prädikat → Kanten



RDF Triple (2)

- Formale Definition:
  - Gegeben drei Mengen:
    U – Uniform Resource Identifiers (URIs)
    B – Blank Nodes („leere Knoten“)
    L – Literale

- Ein RDF Tripel ist eine Element aus der Menge

$$(B \cup U) \times U \times (B \cup L \cup U)$$

- Beispiel:

(Article1, dc:title, „SPARQL“)

URI URI Literal

RDF Datenbanken

- Eine RDF Datenbank D (auch „Graph“ genannt) ist eine Menge von RDF Tripeln, formal

$$D \subseteq (B \cup U) \times U \times (B \cup L \cup U)$$

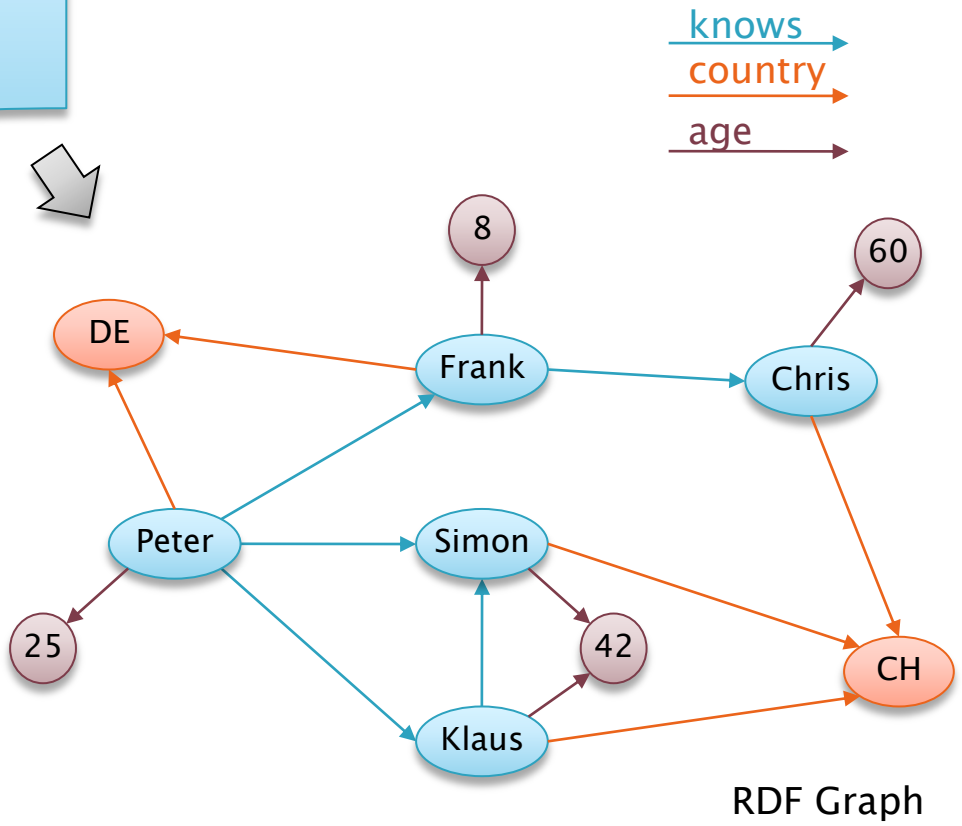
- Serialisierung von RDF Daten in verschiedenen Formaten möglich,
  - z.B. NTriples, N3, RDF/XML, …

Beispiel

```
@prefix ex: <http://example.com/> .
@prefix foaf: <http://xmlns.com/foaf/> .

ex:Chris    foaf:knows      ex:Simon .
ex:Chris    foaf:knows      ex:Sarah .
ex:Sarah    foaf:country    ex:DE .
                  ...
```

RDF Dokument in N3



RDF Graph

Empfehlungen

- RDF Einführung
  - siehe Homepage
- W3C RDF Primer
  - http://www.w3.org/TR/rdf-primer/
- Vorlesungsfolien FGIS
  - http://dbis/index.php?file=fruehereVeranstaltungen.html
  - (SS 2009 / WS 2007)
- RDF Tutorial
  - http://www.w3schools.com/rdf/default.asp

Agenda

- Einarbeitungsphase in MapReduce, RDF und Aufgabenstellung
- Cloudera‘s Distribution for Hadoop installieren
- Team Mitglieder kennenlernen
- **Nächstes Treffen:**
  - Dienstag, 2. November 2010 14-17 Uhr (c.t.)
  - Raum: 01-029, Geb. 51