# Seminar 1912
# MapReduce und Datenbanken

Prof. Dr. Ralf Hartmut Güting
Simone Jandt

Sommersemester 2011

## 1 Allgemeines

Wie angekündigt wollen wir uns im Seminar „MapReduce und Datenbanken“ mit der parallelen Verarbeitung großer Datenmengen mit MapReduce insbesondere im Datenbankumfeld beschäftigen.

Im Abschnitt 2 stellen wir Ihnen die für das Seminar vorgesehen Themen kurz vor. Die in der Literaturliste aufgeführten Quelltexte finden Sie im Internet unter

`http://dna.fernuni-hagen.de/Lehre-offen/Seminare/1912-SS11/LitListespl.html`.

Der Zugriff auf die Seite ist aus urheberrechtlichen Gründen auf Seminarteilnehmer beschränkt. Der Benutzer lautet *Sem1912ss11* und das Kennwort *H1g1uH5l1u*. Wir weisen Sie ausdrücklich darauf hin, dass die zur Verfügung gestellten Quelltexte von Ihnen nur zur Erstellung der Seminararbeit im Rahmen Ihres Studiums genutzt werden dürfen.

Wir bitten Sie, die Themenliste für sich zu priorisieren, d.h. alle in der Tabelle 1 aufgeführten Themen mit einer laufenden Nummer entsprechend Ihren Wünschen zu versehen. Das Thema, das Sie im Seminar am liebsten vorstellen wollen, bekommt die Nummer eins, das erste Ausweichthema die zwei, usw. bis alle Themen durchnummeriert sind. Die entsprechend priorisierte Themenliste senden Sie bitte bis zum 9.3.2011 24 Uhr formlos per E-Mail an `Simone.Jandt@FernUni-Hagen.de`.

Die Benachrichtigung über die Themenvergabe erfolgt am 10.03.2011.

Wir werden uns bemühen, bei der Themenvergabe Ihre Wünsche soweit wie möglich zu berücksichtigen. Aber natürlich können nicht alle das gleiche Thema behandeln. Die Themen 1 - 15 werden auf jeden Fall vergeben. Aus den restlichen Themen werden, unter Berücksichtigung Ihrer Wünsche, zwei ausgewählt und vergeben.

Sie haben dann bis zum 13.04.2011 Zeit, uns Ihre Gliederung einzureichen.

Die endgültige Fassung Ihrer Ausarbeitung und die Folien für die Präsentation erwarten wir dann spätestens am 15.6.2011.

Die Einsendungen können jeweils in Form von PDF-Dateien per E-Mail an Frau Jandt `Simone.Jandt@FernUni-Hagen.de` gesendet werden.

Ursprünglich war für die Präsenzphase des Seminars nur der 15. + 16.7.2011 vorgesehen. Auf Grund des unerwartet hohen Interesses an diesem Seminar haben wir mehr Teilnehmer als ursprünglich geplant akzeptiert. Es kann deshalb sein, dass wir den 14.7.2011 zusätzlich für Seminarvorträge nutzen müssen.

Beachten Sie bei der Erstellung Ihrer Ausarbeitung und der Folien bitte die Hinweise im Abschnitt 3.

| **Thema** | **Titel** | **Priorität** |
|---|---|---|
| 1 | Parallele Datenbanken | |
| 2 | Parallele Anfrageauswertung in Volcano | |
| 3 | MapReduce und Hadoop | |
| 4 | Parallele Datenbanken vs. MapReduce | |
| 5 | HadoopDB | |
| 6 | Dryad | |
| 7 | DryadLINQ | |
| 8 | PigLatin | |
| 9 | SCOPE | |
| 10 | Osprey | |
| 11 | Map-Reduce-Merge | |
| 12 | Optimierung von Joins | |
| 13 | Spatial Join | |
| 14 | MapReduce für Multicore Rechner | |
| 15 | Strom- bzw. Onlineverarbeitung mit MapReduce | |
| 16 | Ähnlichkeitssuche auf Mengen | |
| 17 | Clustern von dynamischen Datenmengen | |
| 18 | Mustererkennung in Netzwerken | |
| 19 | Effiziente Graphanalyse | |
| 20 | Spezielle Anwendungen | |
| 21 | Performance in gemischten Umgebungen | |
| 22 | Sicherheit verteilter Datenbanken | |

Tabelle 1: Prioritätenliste

# 2 Themen

## 2.1 Klassische Parallele Datenbanken

Der erste Vortragsblock bietet einen Überblick über die vor MapReduce existierenden Mechanismen für parallele Datenbanken und Datenverarbeitung.

### 2.1.1 Thema 1: Parallele Datenbanken

[DG92] beschreibt die grundlegenden Ziele und Techniken der parallelen Datenverarbeitung.

### 2.1.2 Thema 2: Parallele Anfrageauswertung in Volcano

Volcano [Gra90, GD93] beschreibt eine Schnittstelle, die die eigentliche Hardware-Architektur kapselt und sich um die Details der parallelen Ausführung von Datenbankoperationen kümmert. So können Operatoren, die diese Schnittstelle bedienen und die ursprünglich für Ein-Prozessor-Systeme konzipiert und optimiert wurden auch in parallelen Datenbankumfeld eingesetzt werden.

## 2.2 Thema 3: MapReduce und Hadoop

Das von Google entwickelte MapReduce Framework [DG04] basiert auf dem Google File System [GGL03]. Seine Open-Source-Implementierung Hadoop [Had11] bringt mit HDFS ein eigenes verteiltes Filesystem mit. [1] Die Grundidee des MapReduce ist in allen weiteren Vorträgen enthalten, deshalb soll es hier mehr um die technischen Hintergründe des MapReduce-Verfahrens und die Sicherstellung seiner Zusicherungen der Fehlertoleranz und der Korrektheit gehen.

## 2.3 Thema 4: Parallele Datenbanken vs. MapReduce

Wie in der Seminarankündigung erwähnt, war der Einsatz von MapReduce im Datenbankumfeld nicht unumstritten. Während die eine Seite die Vorteile der MapReduce-Techniken feierte und alles über MapReduce lösen wollte, beriefen sich die Verfechter herkömmlicher paralleler Datenbanksysteme auf die Vorteile des Einsatzes von Indexen und andere Stärken, paralleler Datenbanksysteme, die von MapReduce nicht unterstützt werden. Eine Diskussion der Stärken und Schwächen beider Systeme bei der Verarbeitung großer Datenmengen findet sich u.a. in [PPR+09, SAD+10, DS08a] und [DS08b].

## 2.4 Datenbanken und MapReduce

In diesem Vortragsblock werden verschiedene auf dem MapReduce Ansatz basierende bzw. von ihm beeinflusste Datenbanksysteme und ihre Anfragesprachen vorgestellt.

### 2.4.1 Thema 5: HadoopDB

Hadoop selbst ist keine Datenbank, sondern nur eine Open-Source-Implementierung des MapReduce-Frameworks. [ABPA+09] beschreibt die Verknüpfung von PostgreSQL und Hadoop mittels Hive [TSJ+09] zu einer kompletten Open-Source-Lösung für die verteilte Speicherung und parallele Auswertung großer Datenmengen mittels MapReduce-Techniken in Rechnerclustern.

[1] Wer für die Ausarbeitung seines Themas vorab tiefere Informationen zu MapReduce benötigt, als in seinem Originaltext vorhanden sind, kann eine Kurzbeschreibung des MapReduce Frameworks in [DG08] finden.

### 2.4.2 Thema 6: Dryad

Microsoft entwickelte mit Dryad [IBY+07] ein mit MapReduce vergleichbares System zur verteilten parallelen Abwicklung von Datenbankabfragen in Clustern. Neben der Vorstellung des Dryad-Systems soll es hier auch um die Gemeinsamkeiten und Unterschiede der beiden Systeme gehen.

### 2.4.3 Thema 7: DryadLINQ

DryadLINQ [YIF+08, IY09] stellt auf der Basis von .NET-Objekten eine Menge von Spracherweiterungen zur Verfügung, die ein neues Programmiermodell für verteilte Anwendungen bilden.

### 2.4.4 Thema 8: PigLatin

Grundlage für den sinnvollen Einsatz von MapReduce im Datenbankumfeld ist die Schaffung entsprechender Schnittstellen zwischen der Datenbankanfragesprache und dem prozeduralen MapReduce-Framework. PigLatin von Yahoo! [ORS+08] ist ein Ansatz die Lücke zwischen SQL und MapReduce zu schließen.

### 2.4.5 Thema 9: SCOPE

SCOPE von Microsoft [CJL+08] versucht analog zu PigLatin die Lücke zwischen SQL und MapReduce zu schließen.

### 2.4.6 Thema 10: Osprey

[YYTM10] ist ein von der MapReduce Idee beeinflusstes System, das in verteilten Datenbanken die fehlertolerante Ausführung von Anfragen, wie Sie bei MapReduce gegeben ist, sicherstellen soll.

## 2.5 Joins

Ein großes Thema in der Datenbankwelt ist der Verbund (Join) mehrerer Eingangsrelationen zu einer Ausgangsrelation. In den letzten Jahren wurden diverse Ansätze veröffentlicht, die den Verbund von Datenmengen mittels MapReduce ermöglichen sollen.

### 2.5.1 Thema 11: Map-Reduce-Merge

[YDHP07] erweitert das MapReduce-Framework um eine Merge-Komponente, die die parallele Ausführung relationaler Operationen, insbesondere auch von Joins, ermöglicht.

### 2.5.2 Thema 12: Optimierung von Joins

Die Optimierung der Ausführung von Operationen ist im Datenbankumfeld immer ein heißes Thema gewesen. So beschäftigt sich [AU10] mit der Optimierung von Join-Operationen im MapReduce-Umfeld.

### 2.5.3 Thema 13: Spatial Join

Insbesondere im Umfeld geographischer Datenbanken ist die Zusammenführung von Daten nach räumlicher Nähe ein Thema. [ZHL+09] beschreibt den Einsatz von MapReduce-Techniken für die effiziente parallele Durchführung von Spatial-Join-Operationen.

## 2.6 Thema 14: MapReduce für Multicore Rechner

[CCZ10] verfeinert die MapReduce-Techniken, um die Datenanalyse auf Rechnern mit vielen Prozessorkernen, die über eine gemeinsame Datenbasis verfügen, zu optimieren.

## 2.7 Thema 15: Strom- bzw. Onlineverarbeitung mit MapReduce

Das Grundkonzept von MapReduce ermöglicht zunächst keine Strom- bzw. laufende Online-Verarbeitung von Daten. [CCA+10, KAGW10] und [BAH10] sind drei Erweiterungen des MapReduce Frameworks, die eine Strom- bzw. Online-Verabeitung auf der Basis von MapReduce-Techniken ermöglichen. Hier reicht es, ein Modell ausführlich darzustellen und mit den anderen kurz zu vergleichen.

## 2.8 Anwendungen

Inzwischen finden MapReduce-Techniken vielfältige Anwendungen im Datenbankumfeld. Ein paar davon sollen im Folgenden vorgestellt werden.

### 2.8.1 Thema 16: Ähnlichkeitssuche auf Mengen

[VCL10] ermöglicht die Ermittlung und Gruppierung ähnlicher Datensätze mittels erweiterter MapReduce-Techniken.

### 2.8.2 Thema 17: Clustern von dynamischen Datenmengen

Bei Google News [DDGR07] werden MapReduce-Verfahren eingesetzt, um ständig die aktuellen Nachrichten verschiedener Quellen zu gleichen Themenkomplexen für Newsportale zusammenzuführen.

### 2.8.3 Thema 18: Mustererkennung in Netzwerken

Bei der Analyse großer Netzwerke treten wiederkehrende Muster auf. Diese können laut [LJC+09] auch mit MapReduce-Techniken ermittelt werden.

### 2.8.4 Thema 19: Effiziente Graphanalyse

[LS10] beschäftigt sich mit der Optimierung der Analyse komplexer Graphen mit MapReduce-Techniken.

### 2.8.5 Thema 20: Spezielle Anwendungen

Zum Schluß beschäftigen wir uns mit drei konkreten auf MapReduce basierenden Anwendungen, die zum Teil auch über graphische Benutzerschnittstellen verfügen. Eins der Systeme sollte ausführlich dargestellt werden, bei den anderen reicht eine kurze Beschreibung.

[WPR+08] stellt eine einfach zu bedienende Benutzerschnittstelle zur Verfügung, die es einfachen Anwendern ermöglicht große in einem Web Archiv abgelegte Dokumentmengen zu analysieren.

[Haz10] beschreibt den Einsatz von MapReduce-Techniken im Zusammenhang mit der Abfrage von Protein Datenbanken.

[JVB09] beschreibt die Parallelisierung Genetischer Algorithmen mit MapReduce-Techniken.

## 2.9 Weitere mögliche Themen

### 2.9.1 Thema 21: Performance in gemischten Umgebungen

Die Fehlertoleranz von Hadoop ist auf Cluster mit identischer Hard- und Software ausgelegt. In der Realität gibt es aber vielfach unterschiedliche Plattformen, die gemischt verwendet werden. [ZKJ+08] beschreibt, wie Hadoop verbessert werden kann, um auch in solchen gemischten Rechnerumgebungen die Fehlertoleranz zu gewährleisten.

### 2.9.2 Thema 22: Sicherheit verteilter Datenbanken

MapReduce wird nicht zuletzt auch auf großen teilweise öffentlichen Clustern eingesetzt. Mit der Frage, wie in diesen öffentlichen Umgebungen der Schutz der Daten und der Privatsphäre sichergestellt werden kann beschäftigt sich [RSK+10].

# 3 Gestaltungshinweise

## 3.1 Inhalt von Vortrag und Ausarbeitung

Wiederholen Sie nicht einfach den oder die zugrunde liegenden Texte. Versuchen Sie vielmehr, den wesentlichen Inhalt mit möglichst einfachen, eigenen Worten zu beschreiben. Bewerten Sie den vorgestellten Ansatz kritisch. Beschreiben Sie die Vorteile, zeigen Sie aber auch die Grenzen und Schwächen auf.

Grundsätzlich ist die Erarbeitung eigener Beispiele zu empfehlen. Das bringt Sie selbst dazu, die Thematik besser zu verstehen.

Häufig ist es nötig, weitere Literatur zum Thema zu berücksichtigen. Ein wichtiger Anhaltspunkt sind die Referenzen, auf die der Basistext verweist. Führen Sie die für Ihren Beitrag genutzte Literatur vollständig auf.

## 3.2 Vortrag

Die Vortragsdauer beträgt 45 Minuten. Daran schließt sich eine etwa 15-minütige Diskussion an.

Wir erwarten, dass Sie nicht nur bei Ihrem eigenen Vortragsthema mit diskutieren!

### 3.2.1 Inhalt

Geben Sie zu Beginn Ihres Vortrags eine kurze Übersicht über die Themen, die Sie erläutern werden.

Versuchen Sie nicht, in Ihrem Vortrag alle Einzelheiten der Basistexte zusammenzufassen, sondern konzentrieren Sie sich auf die wichtigsten Ideen und Konzepte: Lassen Sie lieber den einen oder anderen Gesichtspunkt aus und stellen dafür die übrigen Punkte ausführlich und in Ruhe dar, statt von einem Teil zum nächsten zu eilen.

Halten Sie Ihren Vortrag zur Probe vor Bekannten (oder auch alleine, aber in jedem Fall laut) und vergewissern Sie sich so, dass Ihr Zeitplan korrekt ist und dass Ihre Erklärungen ausreichend sind.

Überprüfen Sie nach einzelnen Abschnitten Ihres Vortrags jeweils, ob Sie noch innerhalb des Zeitplans liegen. Lassen Sie ggf. weniger wichtige Teile aus, um Ihren Vortrag in der vorgegebenen Zeit abschließen zu können.

### 3.2.2 Folien

Verwenden Sie ein geeignetes Programm (z.B. OpenOffice Impress oder Microsoft Powerpoint) zur Erstellung Ihrer Präsentationsfolien. Manchmal ist auch eine „einfache“ Textverarbeitung hilfreich; wenn Sie daraus ein PDF erzeugen, eignet sich der Acrobat-Reader im Vollbildmodus gut zur Präsentation.

Achten Sie auf gute *Lesbarkeit*:

- Benutzen Sie einen genügend großen (16 oder 18 Punkt) und gut lesbaren Zeichensatz.
- Bedenken Sie auch, dass manche *Farben* von einem Projektor nicht so gut dargestellt werden.
- *Hervorhebungen* erreichen Sie durch Farbwechsel, Kursiv- bzw. Fettschrift und verschiedene Schriftarten. Gestalten Sie Hervorhebungen *einheitlich*. Verwenden Sie so wenig Hervorhebungen, wie nötig!
- Vermeiden Sie ganze Sätze! Geben Sie stattdessen Stichpunkte an und verwenden Sie Skizzen, Diagramme, Bilder oder Formeln, die Sie dann in Ihrem Vortrag erläutern.
- Überladen Sie einzelne Folien nicht mit Informationen! Psychologen haben herausgefunden, dass 3–5 Punkte pro Folie optimal sind (allerhöchstens 7!).
- Die Zuhörer sollen in erster Linie Ihren Ausführungen folgen. Die Folien dienen als *zusätzliche Stütze* für das Publikum.
- Wenn Sie auf jede Folie 3 Minuten verwenden, können Sie in Ihrem Vortrag maximal 15 Folien präsentieren. Planen Sie die Folien und deren Inhalt deshalb vorher genau.

Die *Titelseite* nennt den Seminartitel, das Thema (Nr. und Titel) und Ihren Namen.

Eine *Übersicht* (Gliederung Ihres Vortrags) ist auf der zweiten Seite zu finden.

Nach der Titelseite sind alle Seiten mit einer laufenden *Kopfzeile* (mit Unterstrich), die die Informationen Thema (Nr. + Titel), Ihren Namen und die Seitennummer (Inhaltsverzeichnis = Seite 1) enthält, versehen.

## 3.3 Ausarbeitung

Es handelt sich hier um eine Ausarbeitung des Vortrags, also keine Zusammenfassung oder Abschrift/Übersetzung der Literatur und auch keine gedruckte Version der Folien. Einen Vorschlag für die Titelseite finden Sie in Abschnitt 3.3.1.

Typischerweise haben Seminarausarbeitungen einen Umfang von ungefähr 15 DIN A4-Seiten bezogen auf die folgenden Rahmenbedingungen:

- Basis Absatzschriftart: Times-Roman (12 pt)
- Seitenformat Doppelseitig: Abstand zur Bindekante (2,5cm), Abstand Außen (2,0 cm), Unten (2,0 cm), Oben (1,5 cm bis zur Kopfzeile und 2,3 cm bis zum Textrahmen).
- Die Schriftgröße der laufenden Kopfzeile beträgt 9 pt.

Führen Sie am Schluss eine komplette Literaturliste auf. Innerhalb Ihrer Ausarbeitung sollten dann auch Verweise auf die Quellen in Form von Seitennummern und/oder Kapiteln zu finden sein.

Bitte schreiben Sie nirgends in Ihrer Ausarbeitung Ihre Adresse und Matrikelnummer, da es sich hierbei um sensible Informationen handelt, wir aber beabsichtigen einen Seminarband zu erstellen, der auch über die Web-Seiten des Lehrgebietes abrufbar sein wird.

### 3.3.1 Gestaltung der Titelseite

FernUniversität in Hagen
-
Seminar 01912
im Sommersemester 2011

„MapReduce und Datenbanken“

Thema 3

Name des Themas

Referentin: Petra Mustermann

# Literatur


[ABPA+09] Abouzeid, A. ; Bajda-Pawlikowski, K. ; Abadi, D. ; Silberschatz, A. ; Rasin, A.: HadoopDB: An architectural hybrid of MapReduce and DBMS technologies for analytical workloads. In: *Proceedings of the VLDB Endowment* 2 (2009), Nr. 1, S. 922–933. – ISSN 2150–8097

[AU10] Afrati, F.N. ; Ullman, J.D.: Optimizing Joins in a Map-Reduce Environment. In: *Proceedings of the 13th International Conference on Extending Database Technology* ACM, 2010, S. 99–110

[BAH10] Böse, J.H. ; Andrzejak, A. ; Högqvist, M.: Beyond Online Aggregation: Parallel and Incremental Data Mining With Online Map-Reduce. In: *Proceedings of the 2010 Workshop on Massive Data Analytics on the Cloud* ACM, 2010, S. 1–6

[CCA+10] Condie, T. ; Conway, N. ; Alvaro, P. ; Hellerstein, J.M. ; Elmeleegy, K. ; Sears, R.: MapReduce Online. In: *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation* USENIX Association, 2010, S. 21

[CCZ10] Chen, R. ; Chen, H. ; Zang, B.: Tiled-MapReduce: Optimizing Resource Usages of Data-Parallel Applications on Multicore with Tiling. In: *Proceedings of the 19th international conference on Parallel architectures and compilation techniques* ACM, 2010, S. 523–534

[CJL+08] Chaiken, R. ; Jenkins, B. ; Larson, P.Å. ; Ramsey, B. ; Shakib, D. ; Weaver, S. ; Zhou, J.: SCOPE: Easy and efficient parallel processing of massive data sets. In: *Proceedings of the VLDB Endowment* 1 (2008), Nr. 2, S. 1265–1276. – ISSN 2150–8097

[DDGR07] Das, A.S. ; Datar, M. ; Garg, A. ; Rajaram, S.: Google News Personalization: Scalable Online Collaborative Filtering. In: *Proceedings of the 16th international conference on World Wide Web* ACM, 2007, S. 271–280

[DG92] DeWitt, D. ; Gray, J.: Parallel Database Systems: The Future of High Performance Database Systems. In: *Communications of the ACM* 35 (1992), Nr. 6, S. 85–98. – ISSN 0001–0782

[DG04] Dean, J. ; Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters. In: *Proceedings of Operating Systems Design and Implementation (OSDI)*. San Francisco, CA, 2004, S. 137 – 150

[DG08] Dean, J. ; Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters. In: *Communications of the ACM* 51 (2008), January, Nr. 1, S. 107 – 113

[DS08a] DeWitt, J. ; Stonebraker, M.: *MapReduce A Major Step Backwards*. Web Blog. `http://databasecolumn.vertica.com/database-innovation/mapreduce-a-major-step-backwards/`. Version: 2008

[DS08b] DeWitt, J. ; Stonebraker, M.: *MapReduce II*. Web Blog. `http://databasecolumn.vertica.com/database-innovation/mapreduce-ii/`. Version: 2008

[GD93] Graefe, G. ; Davison, DL: Encapsulation of Parallelism and Architecture-Independence in Extensible Database Query Execution. In: *IEEE Transactions on Software Engineering* 19 (1993), Nr. 8, S. 749–764. – ISSN 0098–5589

[GGL03] Ghemawat, S. ; Gobioff, H. ; Leung, S.-T.: The Google File System. In: *19th Symposium on Operating Systems Principles*. Lake George, New York, 2003, S. 29 –43

[Gra90] Graefe, Goetz: Encapsulation of parallelism in the Volcano Query Processing System. In: *Proceedings of the 1990 ACM SIGMOD international conference on Management of data*. New York, NY, USA : ACM, 1990 (SIGMOD '90). – ISBN 0–89791–365–5, 102–111

[Had11] *Hadoop*. Web Page. `http://hadoop.apache.org`. Version: 2011

[Haz10] Hazelhurst, S.: PH2: An Hadoop-Based Framework for Mining Structural Properties from the PDB Database. In: *Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists* ACM, 2010, S. 104–112

[IBY+07] Isard, M. ; Budiu, M. ; Yu, Y. ; Birrell, A. ; Fetterly, D.: Dryad: distributed data-parallel programs from sequential building blocks. In: *ACM SIGOPS Operating Systems Review* 41 (2007), Nr. 3, S. 59–72. – ISSN 0163–5980

[IY09] Isard, M. ; Yu, Y.: Distributed data-parallel computing using a high-level programming language. In: *Proceedings of the 35th SIGMOD international conference on Management of data* ACM, 2009, S. 987–994

[JVB09] Jin, C. ; Vecchiola, C. ; Buyya, R.: MRPGA: An Extension of MapReduce for Parallelizing Genetic Algorithms. In: *eScience, 2008. eScience'08. IEEE Fourth International Conference on* IEEE, 2009, S. 214–221

[KAGW10] Kumar, V. ; Andrade, H. ; Gedik, B. ; Wu, K.L.: DEDUCE: At the Intersection of MapReduce and Stream Processing. In: *Proceedings of the 13th International Conference on Extending Database Technology* ACM, 2010, S. 657–662

[LJC+09] Liu, Y. ; Jiang, X. ; Chen, H. ; Ma, J. ; Zhang, X.: Mapreduce-based pattern finding algorithm applied in motif detection for prescription compatibility network. In: *Advanced Parallel Processing Technologies* (2009), S. 341–355

[LS10] Lin, J. ; Schatz, M.: Design Patterns for Efficient Graph Algorithms in MapReduce. In: *Proceedings of the Eighth Workshop on Mining and Learning with Graphs* ACM, 2010, S. 78–85

[ORS+08] Olston, C. ; Reed, B. ; Srivastava, U. ; Kumar, R. ; Tomkins, A.: Pig latin: a not-so-foreign language for data processing. In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data* ACM, 2008, S. 1099–1110

[PPR+09] Pavlo, A. ; Paulson, E. ; Rasin, A. ; Abadi, D.J. ; DeWitt, D.J. ; Madden, S. ; Stonebraker, M.: A Comparison of Approaches to Large-Scale Data Analysis. In: *Proceedings of the 35th SIGMOD International Conference on Management of Data* ACM, 2009, S. 165–178

[RSK+10] Roy, I. ; Setty, S.T.V. ; Kilzer, A. ; Shmatikov, V. ; Witchel, E.: Airavat: Security and Privacy for MapReduce. In: *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation* USENIX Association, 2010, S. 20

[SAD+10] Stonebraker, M. ; Abadi, D. ; DeWitt, D.J. ; S.Madden ; Paulson, E. ; Pavlo, A. ; Rasin, A.: MapReduce and Parallel DBMSs: Friends or Foes? In: *Communications of the ACM* 53 (2010), January, Nr. 1, S. 64 – 71. `http://dx.doi.org/10.1145/1629175.1629197`. – DOI 10.1145/1629175.1629197

[TSJ+09] Thusoo, A. ; Sarma, J.S. ; Jain, N. ; Shao, Z. ; Chakka, P. ; Anthony, S. ; Liu, H. ; Wyckoff, P. ; Murthy, R.: Hive: a warehousing solution over a map-reduce framework. In: *Proceedings of the VLDB Endowment* 2 (2009), Nr. 2, S. 1626–1629. – ISSN 2150–8097

[VCL10] Vernica, R. ; Carey, M.J. ; Li, C.: Efficient parallel set-similarity joins using MapReduce. In: *Proceedings of the 2010 international conference on Management of data* ACM, 2010, S. 495–506

[WPR+08] Weigel, F. ; Panda, B. ; Riedewald, M. ; Gehrke, J. ; Calimlim, M.: Large-scale collaborative analysis and extraction of web data. In: *Proceedings of the VLDB Endowment* 1 (2008), Nr. 2, S. 1476–1479. – ISSN 2150–8097

[YDHP07] Yang, H. ; Dasdan, A. ; Hsiao, R.L. ; Parker, D.S.: Map-reduce-merge: simplified relational data processing on large clusters. In: *Proceedings of the 2007 ACM SIGMOD international conference on Management of data* ACM, 2007, S. 1029–1040

[YIF+08] Yu, Y. ; Isard, M. ; Fetterly, D. ; Budiu, M. ; Erlingsson, Ú. ; Gunda, P.K. ; Currey, J.: DryadLINQ: A System for General-Purpose Distributed Data-Parallel Computing Using a High-Level Language. In: *Proceedings of the 8th USENIX conference on Operating systems design and implementation* USENIX Association, 2008, S. 1–14

[YYTM10] Yang, C. ; Yen, C. ; Tan, C. ; Madden, S.R.: Osprey: Implementing MapReduce-Style Fault Tolerance in a Shared-Nothing Distributed Database. In: *Data Engineering (ICDE), 2010 IEEE 26th International Conference on* IEEE, 2010, S. 657–668

[ZHL+09] Zhang, S. ; Han, J. ; Liu, Z. ; Wang, K. ; Xu, Z.: SJMR: Parallelizing Spatial Join with MapReduce on Clusters. In: *Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on* IEEE, 2009. – ISSN 1552–5244, S. 1–8

[ZKJ+08] Zaharia, M. ; Konwinski, A. ; Joseph, A.D. ; Katz, R. ; Stoica, I.: Improving MapReduce Performance in Heterogeneous Environments. In: *Proceedings of the 8th USENIX conference on Operating systems design and implementation* USENIX Association, 2008, S. 29–42