Database as a Service

Neue Technologien aus dem Bereich Cloud Computing, NoSQL und Big Data

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Themenblöcke

- Basistechnologien
- Cloud Computing und DBaaS Plattformen
- Big Data Plattformen
- NoSQL Systeme

Ziele des Seminars

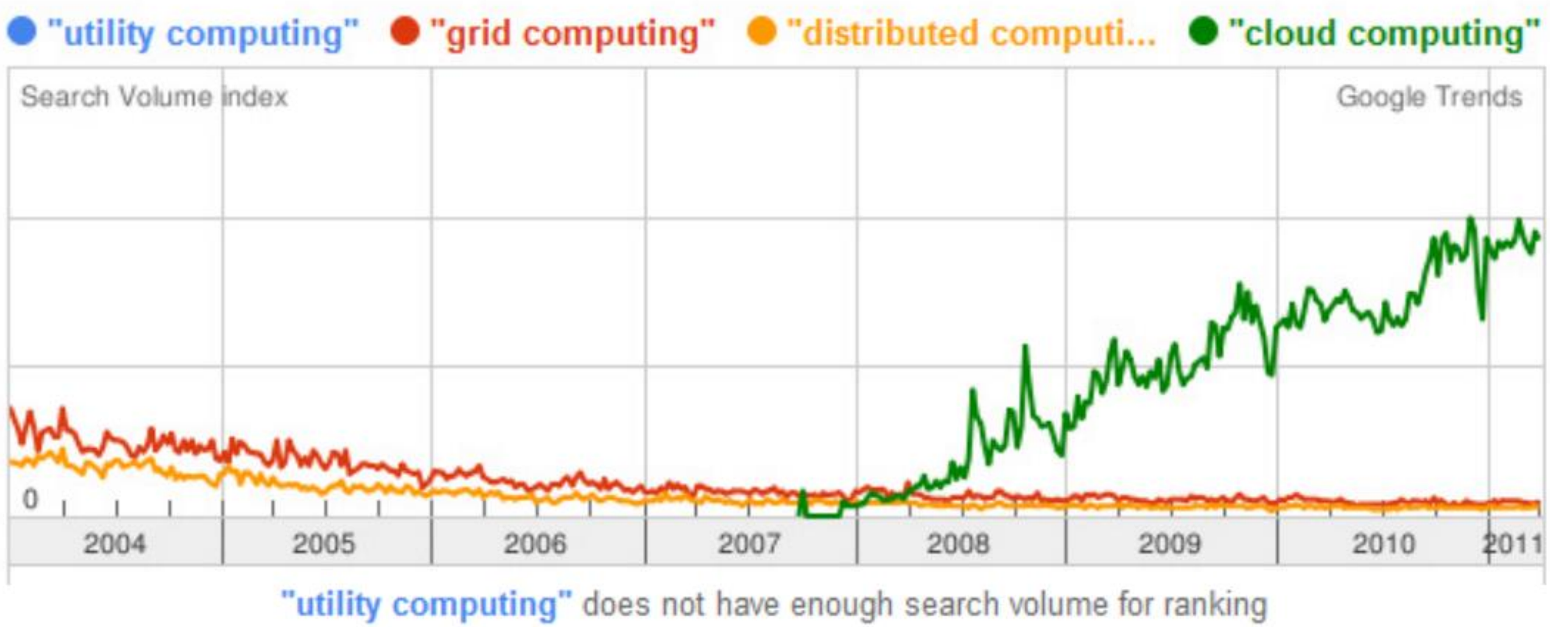
Cloud Computing | Big Data | NoSQL

↓ ↓ ↓

Database-as-a-Service & Cloud Data Management

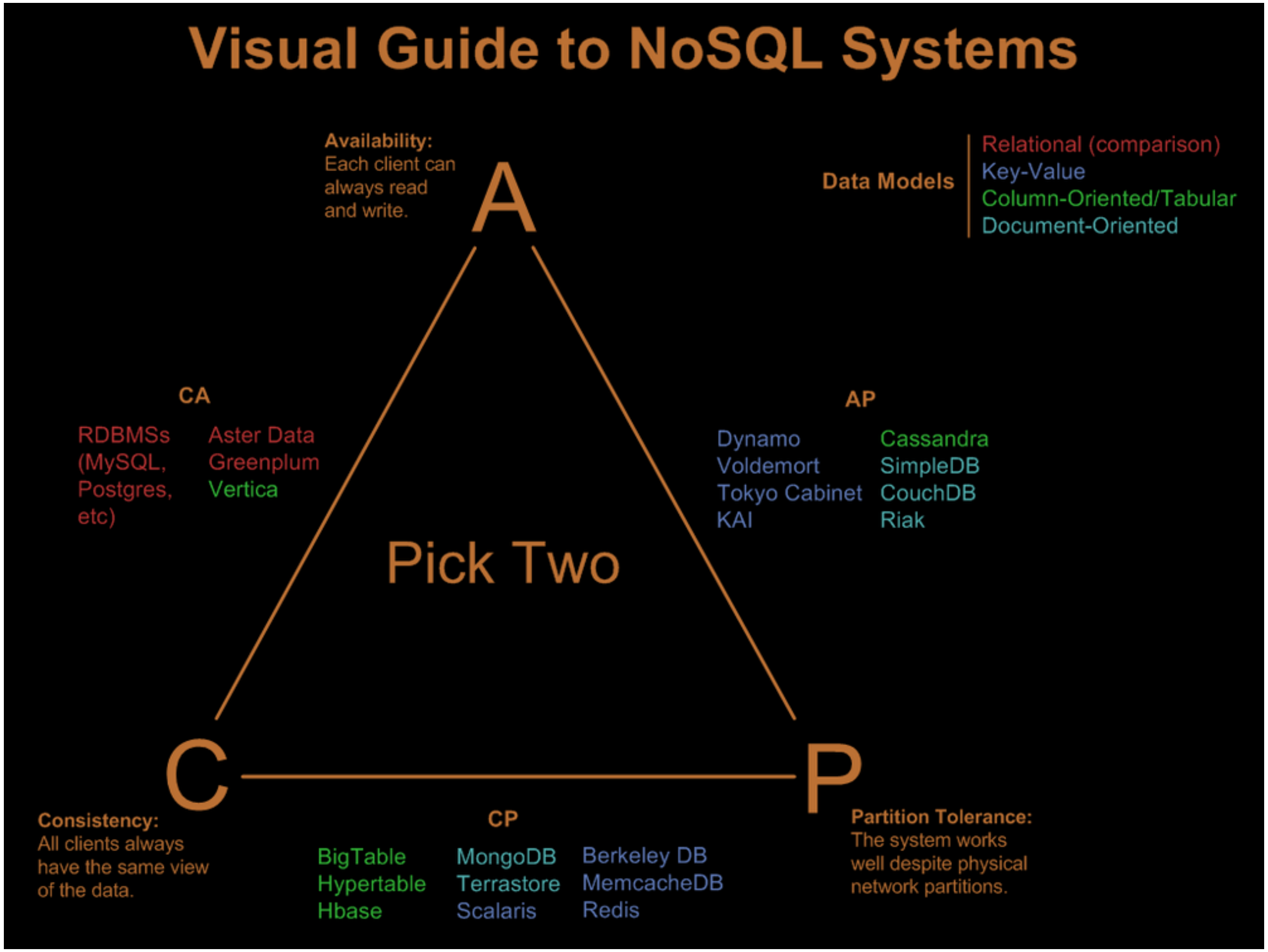
Umbrüche in der Datenbankwelt

1. Cloud Computing:



Umbrüche in der Datenbankwelt

2. NoSQL

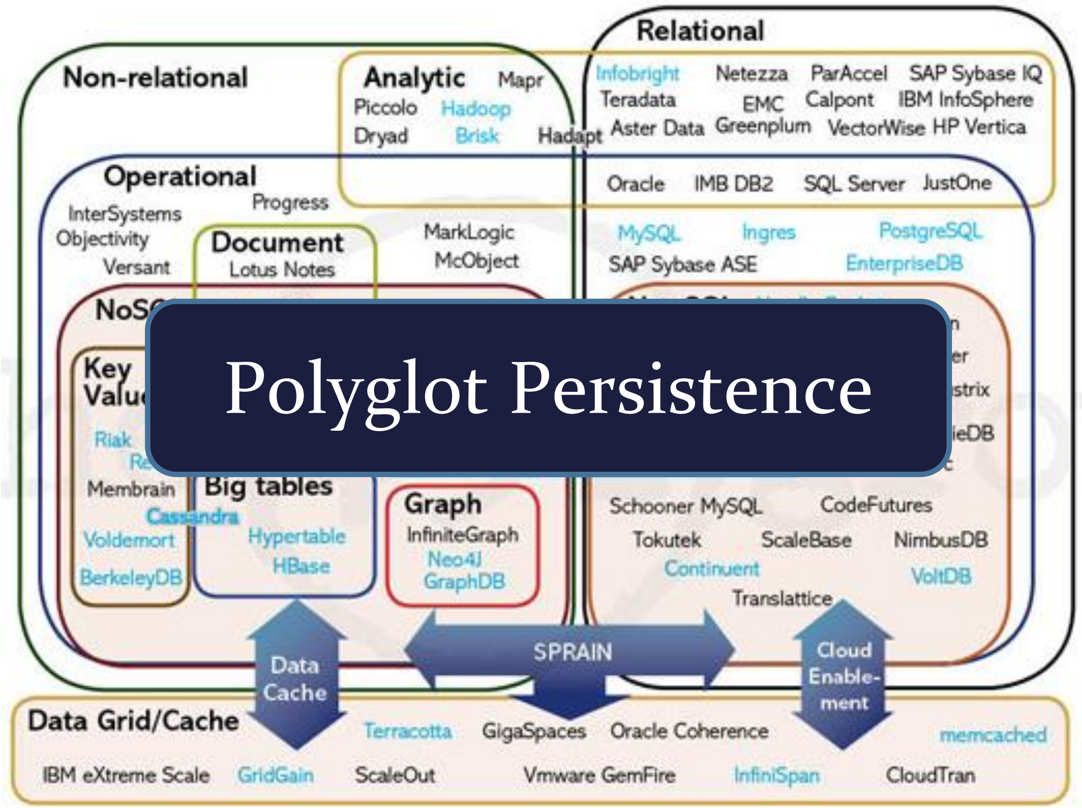


[Nathan Hurst, Visual Guide to NoSQL Systems]

Umbrüche in der Datenbankwelt

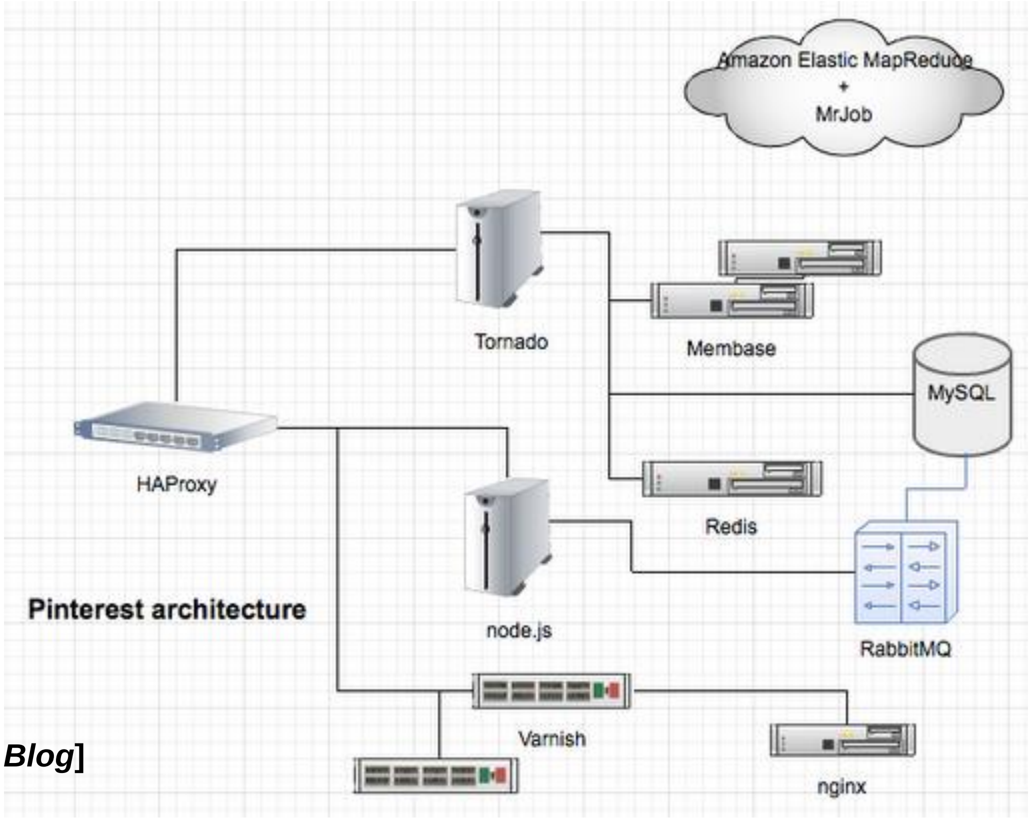
[451 Group Report, 2011]

2. NoSQL



Umbrüche in der Datenbankwelt

3. Big Data

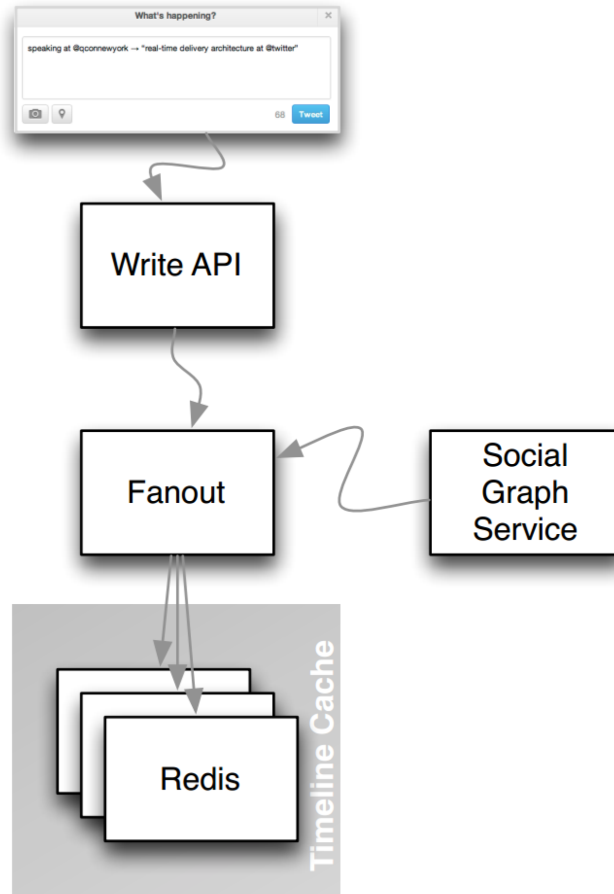


[Poepescu: *myNoSQL Blog*]

Umbrüche in der Datenbankwelt

3. Big Data

>150 Millionen Nutzer
~300k Timeline Querys pro Sekunde

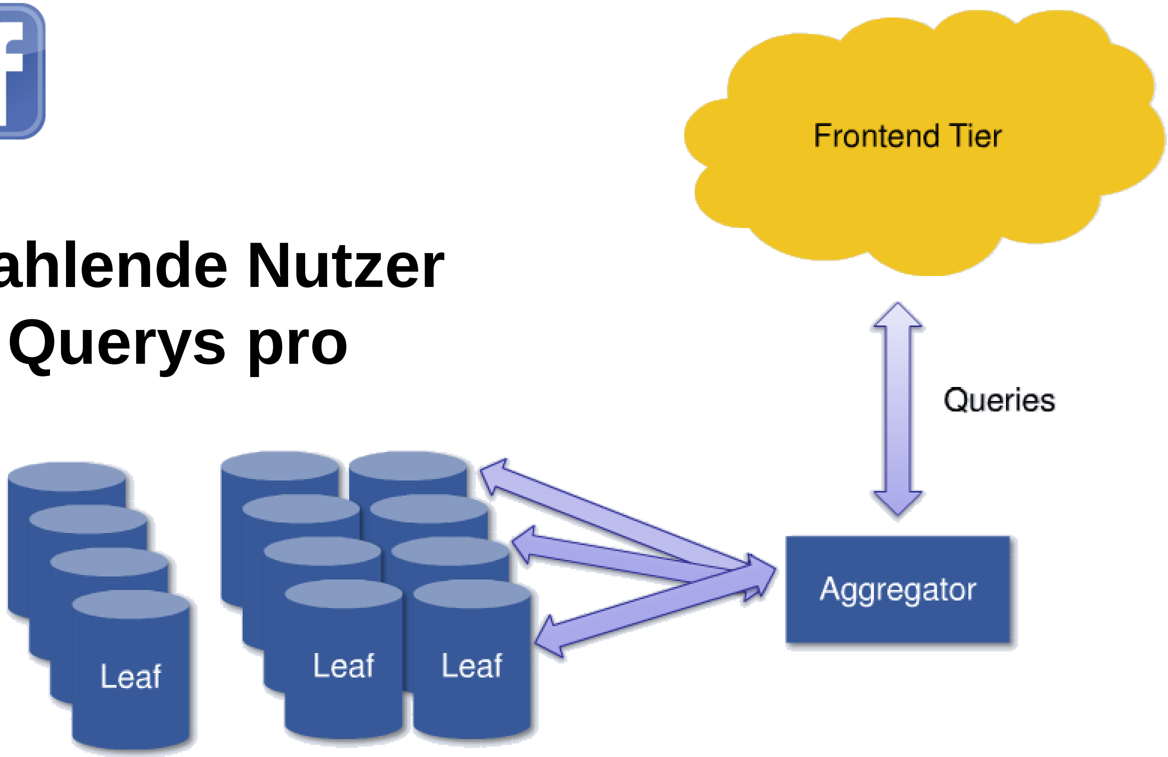


[http://www.infoq.com/presentations/Real-Time-Delivery-Twitter]

Umbrüche in der Datenbankwelt

3. Big Data

>1 Milliarde zahlende Nutzer
~1M Timeline Querys pro Sekunde

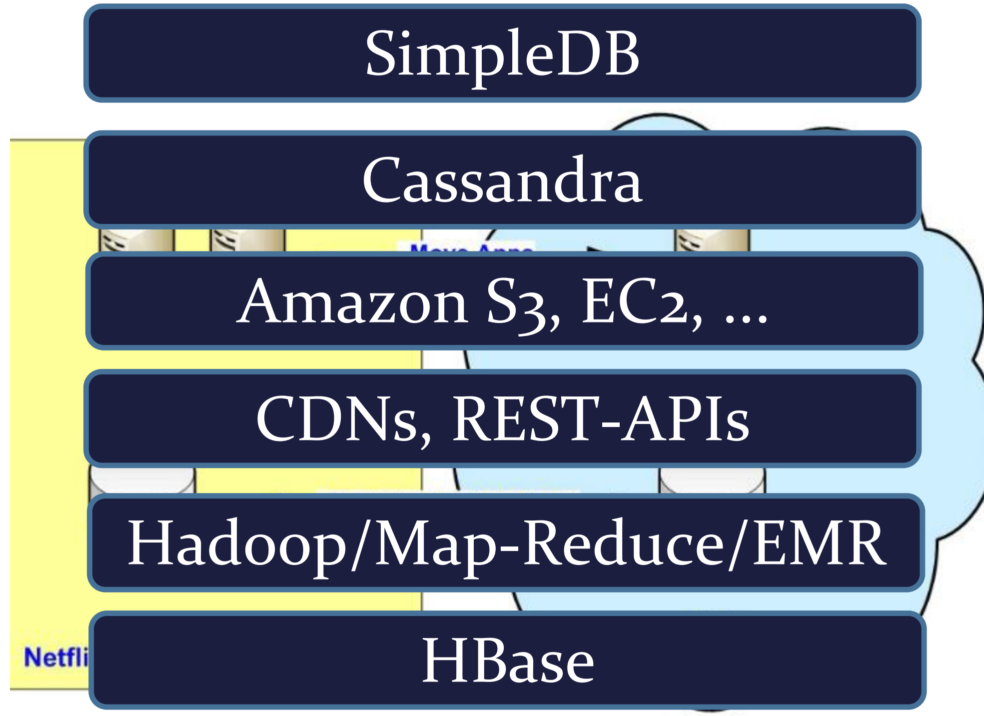


[http://www.infoq.com/presentations/Facebook-News-Feed]

Umbrüche in der Datenbankwelt

→ **NoSQL + Cloud + Big Data**

>20M zahlende Nutzer
~1 Milliarde Stunden gestreamte Videos pro Monat



[http://www.infoq.com/presentations/NoSQL-Netflix]

Database as a Service

mongoHQ

bietet an

mongoDB

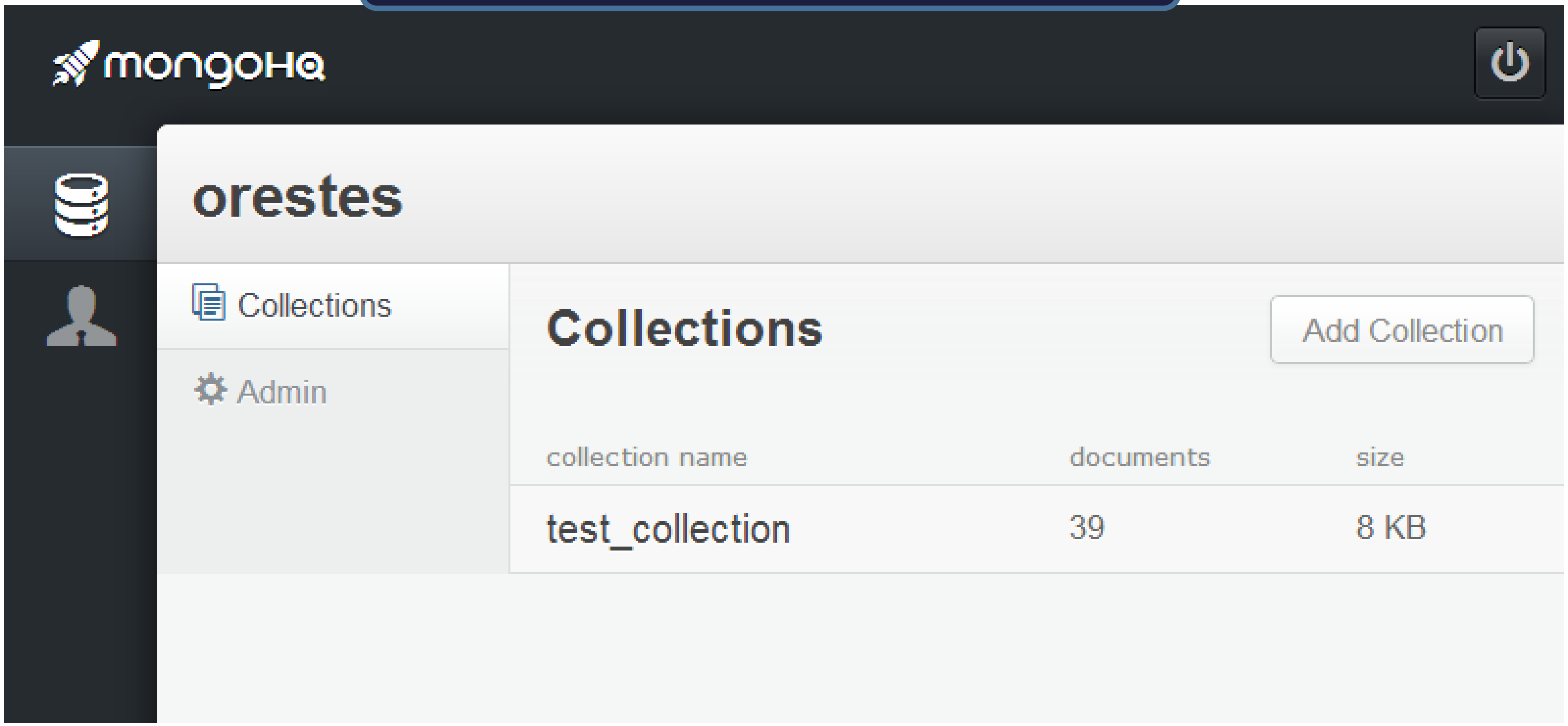
as-a-Service

Database as a Service

Schritt 1: Anmelden

Database as a Service

Schritt 2: DB anlegen



Database as a Service

Schritt 3: Killer-App

```
c = Connection('mongodb://orestes:BrotBrot@alex.mongohq.com:10008/orestes')
obj = { "User" : "Felix", "message" : "Testing MongoDB", "test" : [1,2,3,4] }
c.orestes.test_collection.insert(obj)

with timer:
    c.orestes.test_collection.find_one( {"User":"Felix" })
timer.show()
```

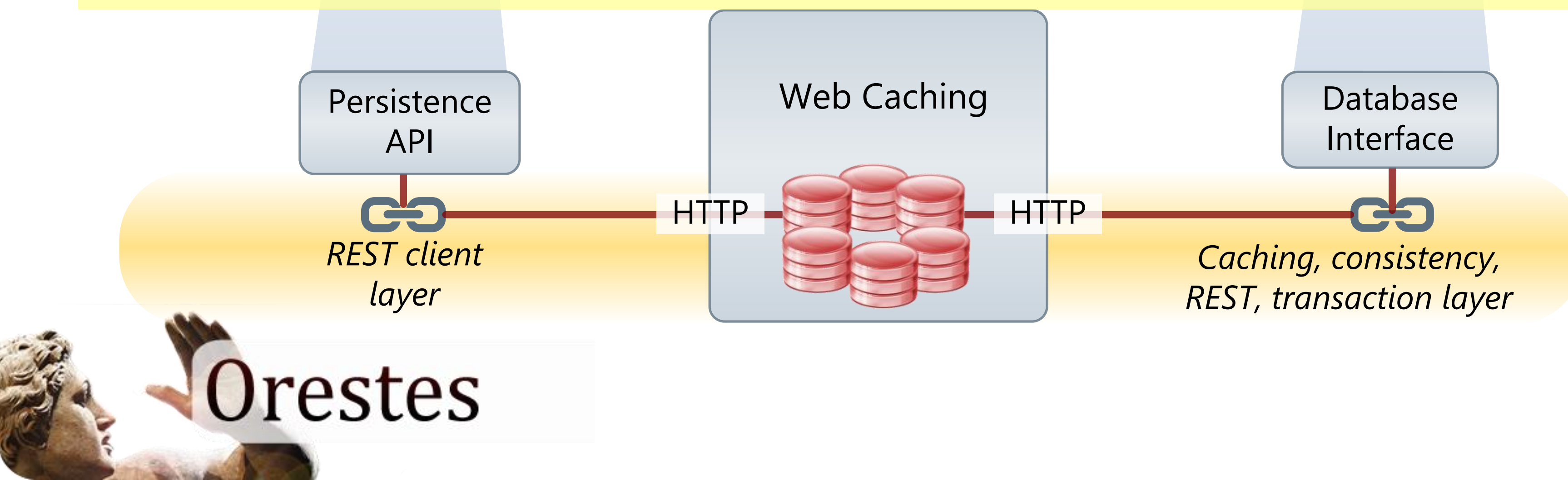
Ziele des Seminars

Cloud Computing	Big Data	NoSQL

Ziel: Die drei neuen Trends und ihren Zusammenhang verstehen

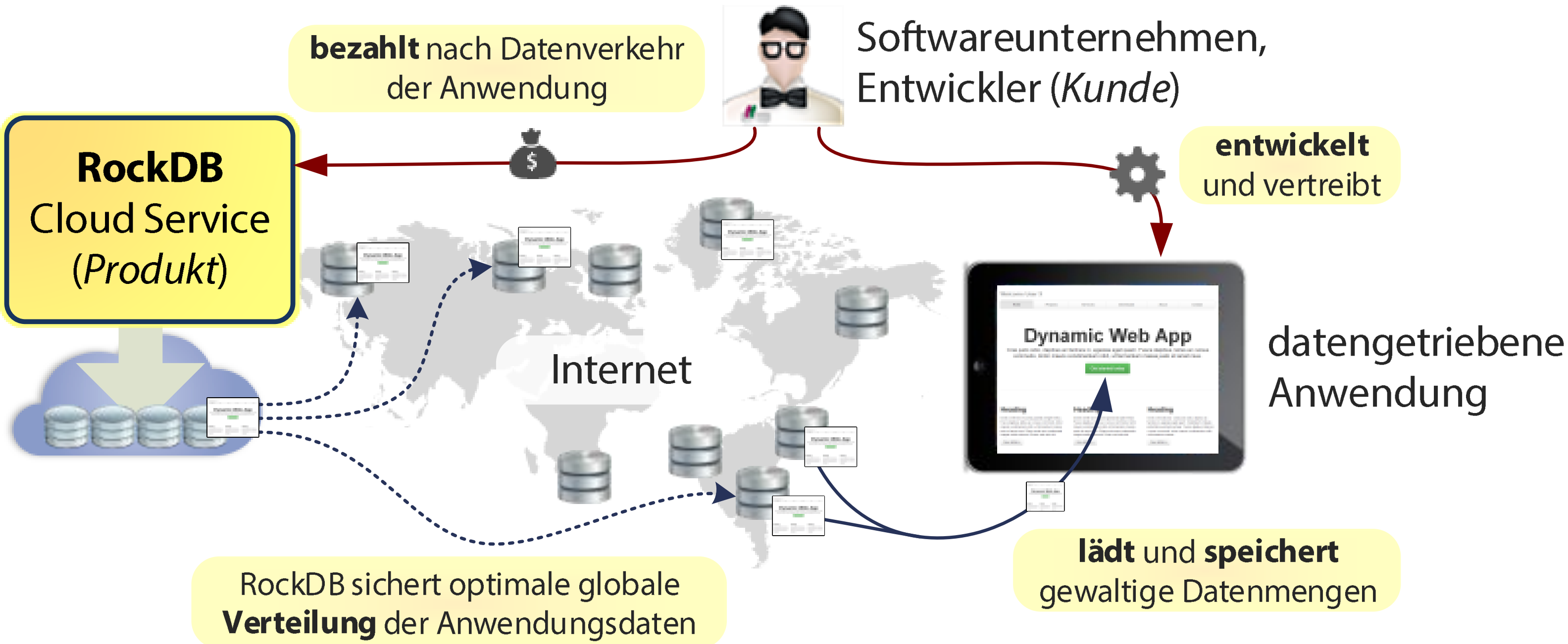
Unsere Forschung

- REST API für NoSQL Backends
- ACID Transaktionen
- Aggressives Caching für geringe Latenz



RockDB

- Database-as-a-Service **Startup** aus der Uni
- Verwertet Forschungsergebnisse



Themen

Basistechnologien

Cloud Computing - Terminologie

Inhalt:

- Unterscheidung zwischen
  - Software as a Service
  - Platform as a Service
  - Infrastrucutre as a Service
- Public Clouds, Private Clouds, Hybrid Clouds
- Praxisbeispiele
- Abgrenzung zu Grid Computing – Ersetzt die Cloud das Grid?

Basistechnologien

Cloud Computing - Terminologie

Referenzen:

- http://www.infoq.com/presentations/Cloud-Introduction
- P. Mell and T. Grance, “The NIST definition of cloud computing,” *National Institute of Standards and Technology*, vol. 53, no. 6, p. 50, 2009.
- Lehner, Wolfgang, and Kai-Uwe Sattler. *Web-scale Data Management for the Cloud*. Springer, 2013.
- M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and others, “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.
- I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud computing and grid computing 360-degree compared,” in *Grid Computing Environments Workshop, 2008. GCE’08*, 2008, pp. 1–10.
- I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, “The physiology of the grid,” in *Grid computing*, 2003, pp. 217–249.
- C. Baun, M. Kunze, J. Nimis, and S. Tai, *Cloud Computing: Web-basierte dynamische IT-Services (Informatik im Fokus)*, 2nd ed. Springer, 2011.

Basistechnologien

DBaaS - Grundlagen

Inhalt:

- Mandantenfähigkeit: Shared machine, Shared Process, Shared Tables
- Privacy/Verschlüsselung
- Skalierbarkeit und Elastizität
- Quality of Service und Workload Management
- DBaaS-Konzepte für relationale Datenbanken: „Relational Cloud“

Basistechnologien

DBaaS - Grundlagen

Referenzen:

- Vortrag „Creating Scalable Multitenant Architectures for the Cloud“: http://dbaas.wordpress.com/2011/05/14/creating-scalable-multitenant-architectures-for-the-cloud/
- Beispiele: http://dbaas.wordpress.com/database-as-a-service-dbaas-product-directory/
- M. Seibold and A. Kemper, “Database as a Service,” *Datenbank-Spektrum*, pp. 1–4, 2012.
- C. A. Curino, E. P. C. Jones, R. A. Popa, N. Malviya, E. Wu, S. R. Madden, H. Balakrishnan, N. Zeldovich, and others, “Relational cloud: A database-as-a-service for the cloud,” 2011.
- R. A. Popa, C. Redfield, N. Zeldovich, and H. Balakrishnan, “CryptDB: protecting confidentiality with encrypted query processing,” in *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, 2011, pp. 85–100.
- S. Krompass, D. Gmach, A. Scholz, S. Seltzsam, and A. Kemper, “Quality of service enabled database applications,” *Service-Oriented Computing–ICSOC 2006*, pp. 215–226, 2006.
- S. Sakr, A. Liu, D. M. Batista, and M. Alomari, “A survey of large scale data management approaches in cloud environments,” *Communications Surveys & Tutorials, IEEE*, vol. 13, no. 3, pp. 311–336, 2011.
- Lehner, Wolfgang, and Kai-Uwe Sattler. *Web-scale Data Management for the Cloud*. Springer, 2013.

Basistechnologien

Virtualisierung

Inhalt:

- Virtualisierung als Grundlage für isoliert laufende Systeme
- Virtuelle Maschinen – Grundlagen:
  - Hardware Virtualisierung und Paravirtualisierung
  - Typ I und Typ II Hypervisor
- Ein Beispiel-Hypervisor erklären, z.B: KVM, Xen, VmWare, HyperV
- Ist Virtualisierung transparent für DBs?

Basistechnologien

Virtualisierung

Referenzen:

- Vorlesungsskript: http://pages.cs.wisc.edu/~remzi/OSFEP/
- Podcast: http://cre.fm/cre092
- P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, “Xen and the art of virtualization,” in *ACM SIGOPS Operating Systems Review*, 2003, vol. 37, pp. 164–177.
- K. Adams and O. Agesen, “A comparison of software and hardware techniques for x86 virtualization,” in *ACM SIGOPS Operating Systems Review*, 2006, vol. 40, pp. 2–13.
- A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori, “kvm: the Linux virtual machine monitor,” in *Proceedings of the Linux Symposium*, 2007, vol. 1, pp. 225–230.
- Potentielles Problem für Durability: fsync Verhalten - [illegible]alization-is-bad-for-database-integrity/

Basistechnologien

REST Services

Inhalt:

- Representational State Transfer – HTTP als Protokoll für serviceorientierte Architekturen
- Grundlagen von HTTP
- REST Constraints, z.B. Statuslosigkeit
- Praxisbeispiel: eine gelungene REST-API erläutern, z.B. Twitter, Facebook, Atom Publishing Protocol, Microsoft OData, Google Gdata, etc.

Basistechnologien

REST Services

Referenzen:

- Vortrag: http://www.infoq.com/presentations/REST-And-Now-for-Something-Completely-Different
- R. T. Fielding, “Architectural styles and the design of network-based software architectures,” University of California, 2000.
- S. Tilkov, *REST und HTTP: Einsatz der Architektur des Web für Integrationsszenarien*. dpunkt, 2011.
- How REST replaced SOAP on the web: http://www.infoq.com/articles/rest-soap

Basistechnologien

Content Delivery Networks (CDNs)

Inhalt:

- Der Schlüssel für schnelle Cloud Services und Websites
- Ziel: Minimierung der Latenz zum Client und Entlastung des Services
- Beispiele: Cloudfront, Akamai
- Techniken (u.a.):
  - DNS-Anycasting, URL-Rewriting, Edge-Site Includes

Basistechnologien

Content Delivery Networks (CDNs)

Referenzen:

- Amazon Cloudfront Doc: http://aws.amazon.com/de/documentation/cloudfront/
- J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, and B. Weihl, "Globally distributed content delivery," *Internet Computing, IEEE*, vol. 6, no. 5, pp. 50–58, 2002.
- M. Pathan and R. Buyya, "A taxonomy of CDNs," *Content delivery networks*, pp. 33–77, 2008.
- Gilbert and Held, *A Practical Guide to Content Delivery Networks, Second Edition*, 2nd ed. CRC Press, 2012.
- R. Buyya, M. Pathan, and A. Vakali, *Content Delivery Networks*, 1st ed. Springer Berlin Heidelberg, 2008.

Basistechnologien

Cloud Computing Plattformen

Amazon Web Services

Inhalt:

- Derzeit größte Cloud Plattform mit diveresen Diensteen (u.a.):
  - Elastic Compute Cloud (EC2)
  - Simple Storage Service (S3)
  - SimpleDB & DynamoDB & RDS
  - Elastic Load Balancer (ELB)
  - Beanstalk
  - Route 53

Cloud Computing Plattformen

Amazon Web Services

Referenzen:

- Online Dokumentation: http://aws.amazon.com/de/
- Präsentation: http://www.infoq.com/presentations/amazon-web-services und http://www.infoq.com/presentations/Deploying-on-Amazon-EC2
- J. van Vliet and F. Paganelli, *Programming Amazon EC2*. O'Reilly Media, 2011.
- C. Baun, M. Kunze, J. Nimis, and S. Tai, *Cloud Computing: Web-basierte dynamische IT-Services (Informatik im Fokus)*, 2nd ed. Springer, 2011.

Cloud Computing Plattformen

Amazon SimpleDB, DynamoDB, S3, RDS

Inhalt:

- Database as a Service in der Amazon Cloud
- SimpleDB:
  - Eingeschränkte aber schnelle Querys
  - Eventually Consistent, Pay-as-you-go, REST-API
- DynamoDB:
  - Automatische Partionierung, SSDs statt HDDs
- Simple Storage Service (S3): Blobs/Dateien
- Relational Data Service (RDS): klassisches Oracle, MS-SQL, MySQL Datenbanksystem

Cloud Computing Plattformen

Amazon SimpleDB, DynamoDB, S3, RDS

Referenzen:

- AWS Online Dokumentation: http://aws.amazon.com/de/
- SimpleDB: http://aws.amazon.com/de/documentation/simpledb/
- DynamoDB: http://aws.amazon.com/de/dynamodb
- RDS: http://aws.amazon.com/de/documentation/rds/
- S3: http://aws.amazon.com/de/documentation/s3/
- J. van Vliet and F. Paganelli, *Programming Amazon EC2*. O'Reilly Media, 2011.

Google App Engine

Inhalt:

- Erläuterung des Google PaaS
- Grundlagen der Anwendungsarchitektur in der GAE
- GAE DataStore – die Schnittstelle zum Google BigTable Backend (die größte DB der Welt)

Cloud Computing Plattformen

Google App Engine

Referenzen:

- Präsentation auf der Google IO 2012: http://www.youtube.com/watch?feature=player_embedded&v=uy0nALQEAM4
- Vortrag „SQL vs NoSQL“ auf der Google IO: http://www.youtube.com/watch?v=rRoy6I4gKWU
- Docs: https://developers.google.com/appengine/docs/
- App Engine Datastore: https://developers.google.com/appengine/docs/java/datastore/?hl=de
- M. C. Chu-Carroll, *Code in the Cloud: Programming Google AppEngine*. Pragmatic Programmers, 2011.
- D. Sanderson, *Programming Google App Engine*, 2. Auflage. O'Reilly

Cloud Computing Plattformen

Windows Azure

Inhalt:

- Überblick über die Services:
  - Anwendungsentwicklung – Programmiersprachen und SDKs
  - Datenspeicherung
  - Integration mit In-House Lösungen
- Diverse Möglichkeiten zur Persistenz: Azure Blobs, Tables, Queues und SQL Azure
- Architektur und Aufbau der Azure Cloud

Cloud Computing Plattformen

Windows Azure

Referenzen:

- Vortrag „Getting started with Windows Azure“: http://channel9.msdn.com/Events/windowsazure/learn/Keynote-Getting-Started-with-Windows-Azure
- David Chappell, Whitepapers: http://www.davidchappell.com/writing/white_papers.php
- Windows Azure Storage http://www.davidchappell.com/writing/white_papers/Windows_Azure_Data_v1.0.pdf
- S. Krishnan, *Programming Windows Azure: Programming the Microsoft Cloud*, 1st ed. O’Reilly Media, 2010.

Cloud Computing Plattformen

Eucalyptus

Inhalt:

- *Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems*
- Open Source Implementierung von Amazon EC2
- Verteilte Server-Architektur
- Grundidee: Abstraktionslayer für Anfragen wie: „Stelle eine neue virtuelle Maschine bereit“
- Abgrenzung zu OpenNebula und OpenStack

Cloud Computing Plattformen

Eucalyptus

Referenzen:

- Vortag Velocity Conference 2008: http://blip.tv/oreilly-velocity-conference/rich-wolski-eucalyptus-elastic-utility-computing-architecture-for-linking-your-programs-to-useful-systems-1026899
- P. Sempolinski and D. Thain, “A comparison and critique of eucalyptus, opennebula and nimbus,” in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, 2010, pp. 417–426.
- D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, “The eucalyptus open-source cloud-computing system,” in *Cluster Computing and the Grid, 2009. CCGRID’09. 9th IEEE/ACM International Symposium on*, 2009, pp. 124–131.

Cloud Computing Plattformen

OpenStack

Inhalt:

- Private IaaS Cloud Plattform (begründet von Rackspace und NASA)
- Compute (Nova)
- Object Storage (Swift), Block Storage (Cinder)
- Networking (Neutron)

Cloud Computing Plattformen

Openstack

Referenzen:

- Videos von der Konferenz OpenStack Summit: http://www.openstack.org/summit/portland-2013/session-videos/?day=1
- Documentation: http://docs.openstack.org/
- Operating OpenStack (frei verfügbares Buch): http://docs.openstack.org/trunk/openstack-ops/openstack-ops-manual-trunk.pdf
- Einführender Blog-Artikel: http://ken.pepple.info/openstack/2012/09/25/openstack-folsom-architecture/
- Wen, Xiaolong, et al. "Comparison of open-source cloud management platforms: OpenStack and OpenNebula." *Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on*. IEEE, 2012.

Cloud Computing Plattformen

Big Data Processing

GFS und HDFS

Inhalt:

- Verteiltes hochskalierbares Dateisystem
- HDFS (Hadoop Projekt) als Implementierung von GFS (Google)
- Grundlage für Map-Reduce
- Namenode-Datanode Architektur

Big Data Processing

GFS und HDFS

Referenzen:

- Intro to HDFS talk: http://www.youtube.com/watch?v=ziqx2hJY8Hg
- Ghemawat, Sanjay, Howard Gobioff, and Shun-Tak Leung. "The Google file system." ACM SIGOPS Operating Systems Review. Vol. 37. No. 5. ACM, 2003.
- Vorlesung Uni Leipzig zu Cloud Data Management: http://dbs.uni-leipzig.de/de/stud/2012ss/cdm
- T. White, *Hadoop: The definitive guide*. Yahoo Press, 2010.
- D. Borthakur, “The hadoop distributed file system: Architecture and design,” *Hadoop Project Website*, vol. 11, p. 21, 2007.

MapReduce und Hadoop

Inhalt:

- Revolution der verteilten Datenverarbeitung durch Google
- Grundlagen:
  - Map-Phase, Shuffle-Phase, Reduce-Phase
- Beispielalgorithmen, z.B. Word-Count
- Hadoop die Open-Source Implementierung:
  - Job-Tracker, Worker, Hadoop Distributed Filesystem
- As-a-Service: Amazon Elastic Map Reduce

Big Data Processing

MapReduce und Hadoop

Referenzen:

- Vortrag über das Hadoop Ökosystem: http://www.infoq.com/presentations/Hadoop-Introduction
- Vorlesung Uni Leipzig zu Cloud Data Management: http://dbs.uni-leipzig.de/de/stud/2012ss/cdm
- Lehner, Wolfgang, and Kai-Uwe Sattler. *Web-scale Data Management for the Cloud*. Springer, 2013.
- J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.
- T. White, *Hadoop: The definitive guide*. Yahoo Press, 2010.
- D. Borthakur, "The hadoop distributed file system: Architecture and design," *Hadoop Project Website*, vol. 11, p. 21, 2007.

Big Data Processing

Pig & Hive

Inhalt:

- Anfragesprachen als Layer über MapReduce
- Hive:
  - HiveQL – eine SQL ähnliche Anfragesprache
  - Queryverarbeitung und Übersetzung in MapReduce
- Pig:
  - Pig Latin - Prozedurale Anfragesprache
- Vergleich, Vor- und Nachteile, Amazon EMR

Big Data Processing

Pig & Hive

Referenzen:

- Vorlesung Uni Leipzig zu Cloud Data Management: http://dbs.uni-leipzig.de/de/stud/2012ss/cdm
- C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins, “Pig latin: a not-so-foreign language for data processing,” in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, 2008, pp. 1099–1110.
- A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, and R. Murthy, “Hive: a warehousing solution over a map-reduce framework,” *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1626–1629, 2009.
- Hive: http://hive.apache.org/
- Pig: http://pig.apache.org/

Big Data Processing

Dremel (Google BigQuery)

Inhalt:

- Idee: wenn man den berüchtigten „Full Table Scan“ nicht vermeiden kann muss man ihn optimieren
- Ziel: SQL Querys auf GigaBytes bis PetaBytes von Daten in wenigen Sekunden
- Interne Realisierung, Verteilungsarchitektur
- As-a-Service: BigQuery

Big Data Processing

Dremel (Google BigQuery)

Referenzen:

- Google IO Vortrag 2012 zu BigQuery: http://www.youtube.com/watch?v=QI8623HIYd4
- S. Melnik, A. Gubarev, J. J. Long, G. Romer, S. Shivakumar, M. Tolton, and T. Vassilakis, “Dremel: interactive analysis of web-scale datasets,” *Proceedings of the VLDB Endowment*, vol. 3, no. 1–2, pp. 330–339, 2010.
- Apache Drill: http://bigdatacraft.com/archives/374

Big Data Processing

Google Pregel

Inhalt:

- Programmiermodell für sehr große Graphenprobleme
- Message-Passing zwischen Knoten des Graphens
- Jeder Knoten hat lokale Funktionen, die definiert werden können
- Implementierung von Googles PageRank: 15 Zeilen Code

Big Data Processing

Google Pregel

Referenzen:

- Einstieg: http://www.quora.com/Pregel/What-are-the-main-concepts-behind-Googles-Pregel
- G. Malewicz, M. H. Austern, A. J. C. Bik, J. C. Dehnert, I. Horn, N. Leiser, and G. Czajkowski, “Pregel: a system for large-scale graph processing,” in *Proceedings of the 2010 international conference on Management of data*, 2010, pp. 135–146.
- Ricky Ho, Blogpost: http://horicky.blogspot.de/2010/07/google-pregel-graph-processing.html

Big Data Processing

Berkley Stack

Inhalt:

- BDAS ([Badass]), *the Berkeley Data Analytics Stack*
- Spark: MapReduce mit Optimierungen für iterative Ausführungen
- Shark: SQL-to-MapReduce
- BlinkDB: SQL mit beschränkten Fehlerraten und Antwortzeiten
- MLBase: Deklaratives Machine-Learning Framework
- 2 davon aussuchen und detailliert behandeln

Big Data Processing

Berkley Stack

Referenzen:

- http://de.slideshare.net/AmazonWebServices/bdt305-tranformingbigdata
- Agarwal, Sameer, et al. "BlinkDB: queries with bounded errors and bounded response times on very large data." *Proceedings of the 8th ACM European Conference on Computer Systems*. ACM, 2013.
- Shenker, Scott, et al. "Shark: SQL and Rich Analytics at Scale." (2012)
- Kraska, Tim, et al. "MLbase: A Distributed Machine-learning System." *CIDR*. 2013.
- Engle, Cliff, et al. "Shark: fast data analysis using coarse-grained distributed memory." *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*. ACM, 2012.

Big Data Processing

NoSQL Systeme

NoSQL – Terminologie und Historie

Inhalt:

- Was bedeutet NoSQL und warum ist die Bezeichnung schlecht?
- Kategorien von NoSQL Systemen (unterschiedliche Taxonomien):
  - Key-Value Stores
  - Dokumentendatenbanken
  - Graphendatenbanken
  - Wide Column Stores

NoSQL Systeme

NoSQL – Terminologie und Historie

Referenzen:

- Vortrag: NoSQL Database Technology: A Survey and Comparison of Systems http://www.infoq.com/presentations/NoSQL-Survey-Comparison
- C. Strauch, U. L. S. Sites, and W. Kriha, "NoSQL databases," *Lecture Notes, Stuttgart Media University*, 2011.
- R. Cattell, "Scalable sql and nosql data stores," *ACM SIGMOD Record*, vol. 39, no. 4, pp. 12–27, 2011.
- P. J. Sadalage, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison-Wesley Professional.
- S. Edlich, *NoSQL Einstieg in die Welt nichtrelationaler Web-2.0-Datenbanken*. München: Hanser, 2010.
- E. Redmond, J. Wilson, and J. Carter, *Seven databases in seven weeks : a guide to modern databases and the NoSQL movement*. Lewisville, Tex.; Farnham: Pragmatic Bookshelf ; O'Reilly [distributor], 2012.
- M. Stonebraker, "SQL databases v. NoSQL databases," *Communications of the ACM*, vol. 53, no. 4, pp. 10–11, 2010.

NoSQL Systeme

CAP Theorem und Skalierbarkeit

Inhalt:

- Die Unvereinbarkeit von **C**onsistency, **A**vailability und **P**artition Tolerance
- Unterschiede zu ACID
- Warum skalieren relationale DBs schlecht?
- Konsequenzen: BASE, Eventual Consistency, Abgeschwächte Konsistenz, z.B. Read-Your-Writes

NoSQL Systeme

CAP Theorem und Skalierbarkeit

Referenzen:

- Teaser: http://adam.heroku.com/past/2009/7/6/sql_databases_dont_scale/
- Werner Vogels (Amazon CTO), „Eventually Consistent“: http://www.allthingsdistributed.com/2007/12/eventually_consistent.html
- S. Gilbert and N. Lynch, “Brewer’s conjecture and the feasibility of consistent, available, partition-tolerant web services,” *ACM SIGACT News*, vol. 33, no. 2, pp. 51–59, 2002.
- E. A. Brewer, “Towards robust distributed systems,” in *Proceedings of the Annual ACM Symposium on Principles of Distributed Computing*, 2000, vol. 19, pp. 7–10.
- R. Ramakrishnan, “CAP and Cloud Data Management,” *Computer*, vol. 45, no. 2, pp. 43–49, 2012.
- E. Brewer, “CAP twelve years later: How the,” *Computer*, vol. 45, no. 2, pp. 23–29, 2012.

NoSQL Systeme

Google BigTable

Inhalt:

- Das Datenbankbackend von Google
- Verteilungsarchitektur: Sharding (Partitionierung)
- Schnittstelle: nichttransaktionaler Zugriff
- Konsistenzmodell: Replikation, eventually consistent
- Open-Source Implementierung: HBase
  - MapReduce Querys, Verteilte Datenhaltung in HDFS
  - Anfrage durch REST-API oder RPC (Thrift, Avro)

NoSQL Systeme

Google BigTable

Referenzen:

- Jeff Dean, Vortrag http://video.google.com/videoplay?docid=7278544055668715642
- Ricky Ho, Blog http://horicky.blogspot.de/2010/10/bigtable-model-with-cassandra-and-hbase.html
- F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, “Bigtable: A distributed storage system for structured data,” *ACM Transactions on Computer Systems (TOCS)*, vol. 26, no. 2, p. 4, 2008.
- HBase Vortag http://www.infoq.com/presentations/HBase-at-Facebook
- L. George, *HBase: The Definitive Guide: The Definitive Guide*. O’Reilly Media, 2011.

NoSQL Systeme

Amazon Dynamo

Inhalt:

- Hochperformanter, verteilter Key-Value Store
- Verschmelzung raffinierter Algorithmen:
  - Merkle-Trees, Gossip Protokoll
  - Consistent Hashing, Vector Clocks
- Inspiriert Voldemort, Riak, Dynomite
- Besonders gut: Riak
  - Wählbare Konsistenz, Agnostisch gegenüber Speichertechnik (z.B. Dateisystem, Datenbanksystem)

Amazon Dynamo

Referenzen:

- Einstieg: http://www.infoq.com/presentations/Riak-Core
- Vortrag „Dynamo is not just for Datastores“ http://www.infoq.com/presentations/Dynamo-Is-Not-Just-for-Datastores

- G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, “Dynamo: amazon’s highly available key-value store,” in *ACM SIGOPS Operating Systems Review*, 2007, vol. 41, pp. 205–220.
- C. Strauch, U. L. S. Sites, and W. Kriha, “NoSQL databases,” *Lecture Notes, Stuttgart Media University*, 2011.
- E. Redmond, J. Wilson, and J. Carter, *Seven databases in seven weeks : a guide to modern databases and the NoSQL movement*. Lewisville, Tex.; Farnham: Pragmatic Bookshelf ; O’Reilly [distributor], 2012.

NoSQL Systeme

MongoDB

Inhalt:

- Populäre Dokumentendatenbank
- Funktionalität:
  - Speicherung von JSON Dokumenten
  - Komplexe Queries auf Dokuementen
  - Replikation und Sharding (Kontrollierbare Verfügbarkeit und Performance)
  - Map-Reduce Querys

NoSQL Systeme

MongoDB

Referenzen:

- Teaser-Vortrag http://www.infoq.com/presentations/Why-I-Chose-MongoDB-for-Guardian
- MongoDB Architecture, Ricky Ho, Blog http://horicky.blogspot.de/2012/04/mongodb-architecture.html
- MongoDB Docs: http://www.mongodb.org/display/DOCS/Home
- C. Strauch, U. L. S. Sites, and W. Kriha, “NoSQL databases,” *Lecture Notes, Stuttgart Media University*, 2011.
- K. Chodorow and M. Dirolf, *MongoDB: the definitive guide*. O’Reilly Media, 2010.

NoSQL Systeme

CouchDB

Inhalt:

- Webnahe Dokumentendatenbank
- Eigenschaften:
  - Map-Reduce Views mit JavaScript
  - REST API
  - Replikation und Konflikterkennung
  - Embedded Mode

NoSQL Systeme

CouchDB

Referenzen:

- Damien Katz: http://www.infoq.com/presentations/katz-couchdb-and-me
- Will LeinWeber, Vortrag http://www.infoq.com/presentations/couchdb-Will-Leinweber
- CouchDB Wiki http://wiki.apache.org/couchdb/
- C. Strauch, U. L. S. Sites, and W. Kriha, “NoSQL databases,” *Lecture Notes, Stuttgart Media University*, 2011.
- J. C. Anderson, J. Lehnardt, N. Slater, and Safari Tech Books Online, *CouchDB the definitive guide*. Sebastopol, Calif.: O’Reilly Media, Inc., 2010.
- E. Redmond, J. Wilson, and J. Carter, *Seven databases in seven weeks : a guide to modern databases and the NoSQL movement*. Lewisville, Tex.; Farnham: Pragmatic Bookshelf ; O’Reilly [distributor], 2012.

NoSQL Systeme

Redis

Inhalt:

- Datenstrukturserver: Key-Value Paare, Listen, Sets, Queues, Sorted Sets, Maps
- Extreme Geschwindigkeit
- Replikation und konfigurierbare Persistenzgarantien
- Wahrscheinlicher Erbe von Memcached

Redis

Referenzen:

- Einstieg, „Redis in depth“, Vortrag http://www.infoq.com/presentations/Redis
- Dokumentation: http://redis.io/documentation
- S. Edlich, *NoSQL Einstieg in die Welt nichtrelationaler Web-2.0-Datenbanken*. München: Hanser, 2010.
- C. Strauch, U. L. S. Sites, and W. Kriha, “NoSQL databases,” *Lecture Notes, Stuttgart Media University*, 2011.
- E. Redmond, J. Wilson, and J. Carter, *Seven databases in seven weeks : a guide to modern databases and the NoSQL movement*. Lewisville, Tex.; Farnham: Pragmatic Bookshelf ; O’Reilly [distributor], 2012.

NoSQL Systeme

Cassandra

Inhalt:

- Synthese aus BigTable und Amazon Dynamo
- Entwickelt für die Suche in der Facebook Inbox
- Eigenschaften:
  - Skalierbarkeit und Fehlertoleranz (Dezentralisierung)
  - Query-Sprache CQL
  - Integration mit Hadoop

Cassandra

Referenzen:

- „Adopting Apache Cassandra“, Vortrag http://www.infoq.com/presentations/Adopting-Apache-Cassandra
- A. Lakshman and P. Malik, “Cassandra: a decentralized structured storage system,” *ACM SIGOPS Operating Systems Review*, vol. 44, no. 2, pp. 35–40, 2010.
- Cassandra Wiki: http://wiki.apache.org/cassandra/
- P. J. Sadalage, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison-Wesley Professional.

NoSQL Systeme

CryptDB

Inhalt:

- Verschlüsselte Datenbank in der Cloud
- Problem: Querys auf verschlüsselten Daten
- Lösung: Homomorphe Verschlüsselung
- Praktische Umsetzung durch Onion Encryption
- Teil des DBaaS Systems *Relational Cloud*

NoSQL Systeme

CryptDB

Referenzen:

- Popa, Raluca Ada, et al. "CryptDB: protecting confidentiality with encrypted query processing." *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*. ACM, 2011.
- Curino, Carlo, et al. "Relational cloud: A database-as-a-service for the cloud." (2011).

ElasTras

Inhalt:

- Elastras: An elastic transactional data store in the cloud
- Idee: Durch Begrenzung auf Partitionen sowohl Transaktionen als auch Skalierbarkeit ermöglichen
- Architektur mit Load-Balancing, Transaction Managern und verteilter Speicherung (Amazon S3)

NoSQL Systeme

ElasTras

Referenzen:

- Das, Sudipto, Divyakant Agrawal, and Amr El Abbadi. "Elastras: An elastic transactional data store in the cloud." *USENIX HotCloud* 2 (2009).
- Das, Sudipto, Divyakant Agrawal, and Amr El Abbadi. "ElasTraS: An elastic, scalable, and self-managing transactional database for the cloud." *ACM Transactions on Database Systems (TODS)* 38.1 (2013): 5.

NoSQL Systeme

Hyperdex

- Inhalt
  - Searchable Key-Value Store, entwickelt an der Cornell University
  - Idee: Hypersapce Hashing
  - Hochverfügbar
  - Transaktionslayer: Warp

Hyperdex

Referenzen:

- Foliensätze: http://hyperdex.org/papers/
- Escriva, Robert, Bernard Wong, and Emin Gün Sirer. "HyperDex: A distributed, searchable key-value store." *ACM SIGCOMM Computer Communication Review* 42.4 (2012): 25-36.
- Escriva, Robert, Bernard Wong, and Emin Gün Sirer. "Warp: Multi-Key Transactions for Key-Value Stores."

NoSQL Systeme

H-Store/VoltDB

- Inhalt:
  - NewSQL Bewegung
  - Skalierbare OLTP SQL Datenbank
  - Main-Memory
  - Tabellen werden partitioniert über Server
  - Gesamter Concurrency Control Overhead entfernt: kein Latching, Locking und Logging
  - Entwicklung in Java Stored Procedures

H-Store/VoltDB

- Referenzen:
  - Not your father transaction processing (video): http://www.infoq.com/presentations/NewSQL-VoltDB
  - Kallman, Robert, et al. "H-store: a high-performance, distributed main memory transaction processing system." *Proceedings of the VLDB Endowment* 1.2 (2008): 1496-1499.
  - Stonebraker, Michael, et al. "The end of an architectural era:(it's time for a complete rewrite)." *Proceedings of the 33rd international conference on Very large data bases*. VLDB Endowment, 2007.
  - http://highscalability.com/blog/2010/6/28/voltdb-decapitates-six-sql-urban-myths-and-delivers-internet.html
  - http://www.dbms2.com/2010/05/25/voltdb-finally-launches/

Übersicht - Basistechnologien

Thema	Vortragender
Cloud Computing - Terminologie	
DBaaS - Grundlagen	
Virtualisierung	
REST Services	
Content Delivery Networks (CDNs)	

Cloud Computing Plattformen

Thema	Vortragender
Amazon Web Services	
Amazon SimpleDB, DynamoDB, S3, RDS	
Google App Engine	
Windows Azure	
Eucalyptus	
Openstack	

Big Data Processing

Thema	Vortragender
MapReduce und Hadoop	
Pig & Hive	
Dremel (Google BigQuery)	
Google Pregel	
Der Berkley Stack	

Übersicht – NoSQL

Thema	Vortragender
NoSQL – Terminologie und Historie	
CAP Theorem und Skalierbarkeit	
Google BigTable	
Amazon Dynamo	
MongoDB	
CouchDB	
Redis	
Cassandra	
Hyperdex	
H-Store/VoltDB	
CryptDB	

Email an

gessert@informatik.uni-hamburg.de

Inhalt:

- Wann könnt ihr bei einem Blocktermin (3 Tage, in der Woche, vorlesungsfreie Zeit)
- Top 3 Themenauswahl
- Nicht in Stine Angemeldete werden gemäß Präferenz verteilt, wenn alle Angemeldeten ein Thema haben

Ab Montag 28.10.2013, 23:59, First Come First Serve

Formalien

- Seminararbeit: 15 Seiten netto
- Vortrag:
  - 30 Minuten + 5 min. Diskussion
  - <50% Textfolien