

Database-as-a-Service: Übersicht

Cloud Data Management Seminar
Christian Kötteritzsch

Inhalt

- 1.Motivation
- 2.Cloud Computing
- 3.Database-as-a-Service
- 4.Vergleich zwischen DaaS und lokaler Installation
- 5.Database-as-a-Service Model
- 6.Adaption des Modells an Forschungsprototypen
- 7.Adaption des Modells an Amazon Simple DB
- 8.Zusammenfassung
- 9.Literatur

Motivation

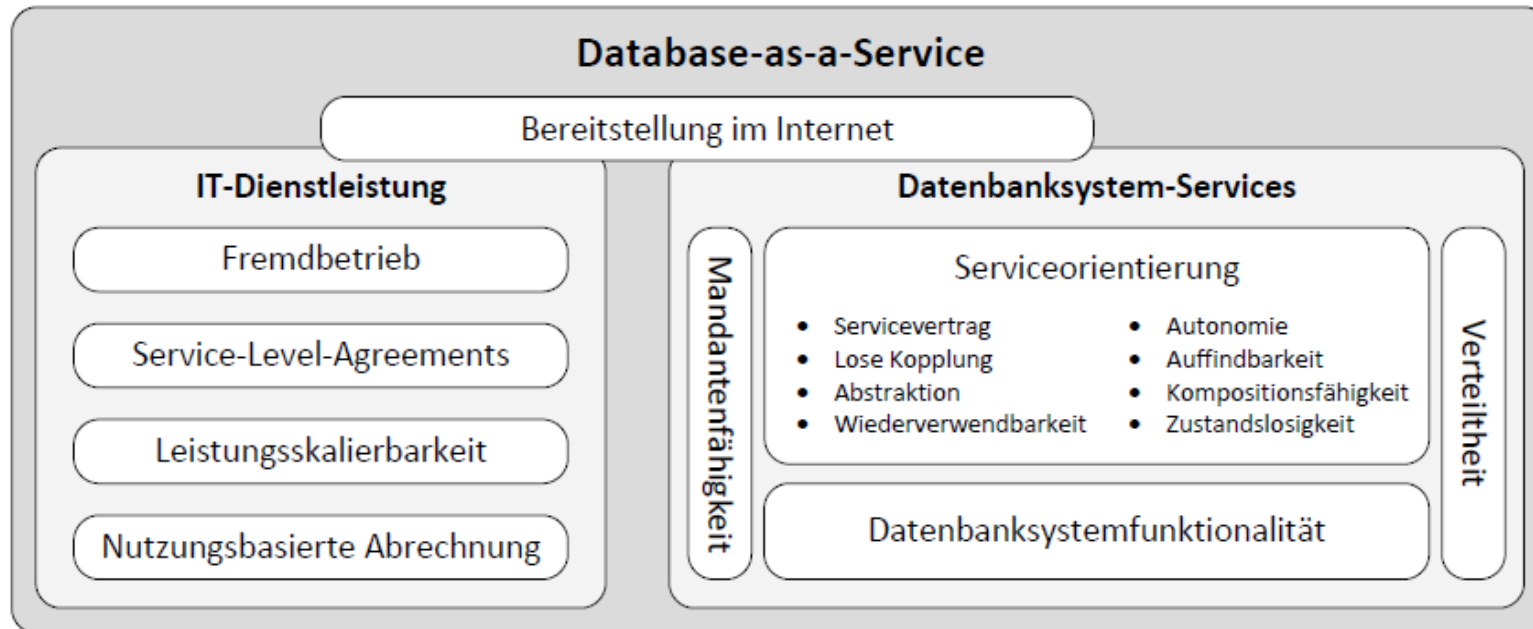
- Datenbanken wichtiger Teil der IT
- aufwändig und kostenintensiv Datenbanksystem zu betreiben
- Trend geht zu Datenbanksysteme als Service
- Basiert auf Clou Computing
- Dadurch keine Kosten für Anschaffung und Betrieb
- Kunde zahlt nur die Reine Leistung die benötigt wird
- Kunde kann sich somit auf Kerngeschäft konzentrieren
- Bsp.: -Amazon Simple Storage Service
-Amazon SimpleDB

Cloud Computing

- Ziel ist es Hardwareressourcen zur Verfügung zu stellen
- Stellt auch Software zur Verfügung
- Nutzer hat Zugriff auf neue Technologien
- Nutzungsbezogene Abrechnung
- Kann auf Bedürfnisse angepasst werden
- Erfüllung der Leistung in Service Level Agreements festgelegt

- Abhängig vom Anbieter
- Langsamer durch höhere Latenzzeit
- Datenschutz und Datensicherheit nicht ausreichend geklärt

Database-as-a-Service



Quelle:

- Es gibt 2 Konzepte bei Database-as-a-Service
- 1.) IT Dienstleistungen
- 2.) Datenbanksystem-Services

Database-as-a-Service

- Bietet entweder komplette Datenbanksysteme oder Teilsysteme
- Können durch Komposition zusammengestellt werden
- Es ist nur Schnittstellenbeschreibung bekannt
- Leistungen durch SLA's festgelegt
- Abrechnung nach angeforderter Leistung
- Energieeinsparungen
- Unternehmen kann sich auf Kerngeschäft konzentrieren

- Sehr große Abhängigkeit vom Anbieter
- Unternehmensdaten liegen in fremden Unternehmen → Datenschutz und Datensicherheit
- Fehleranalyse, Debugging und Reportig eingeschränkt möglich
- Noch nicht alle Datensicherheitsaspekte geklärt

Herausforderungen

- Mandantenfähigkeit
 - schwieriger Aspekt muss zwischen Sicherheit, Kosten und individuellen Anforderungen abwägen
- Verwaltung der Infrastruktur
 - Administration und Wartung von Hard-und Software
 - Hinzufügen und entfernen von Ressourcen
 - Echtzeitüberwachung aller Ressourcen
- Überwachung der SLA's
 - Protokollierung der Leistungen für Servicegarantierung
 - Monitoring-Komponente die Systemleistung nicht beeinflusst

Weitere Herausforderungen

- Caching von Objekten
 - Hohe Kommunikationskosten → Verwednung bestehender Caching verfahren
- Zugriffskontrolle
 - Zugriffskontrolle in 5. Schicht aber 5. Schicht oft nicht genutzt → Zugriffskontrolle auf unterer Schicht umsetzen
- Administration durch den Kunden
 - Kundenseitige Änderung der Dienstparameter → hat Anpassung an bereitgestellten Ressourcen
 - Nicht geklärt ob durch Automatismen realisierbar
- Schaffen von Vertrauen
 - Fehleranalyse nur eingeschränkt möglich
 - Exportfunktionen, Monitoring-Werkzeuge, Analysefunktionen
 - Kunde erhält Kontrolle über Daten

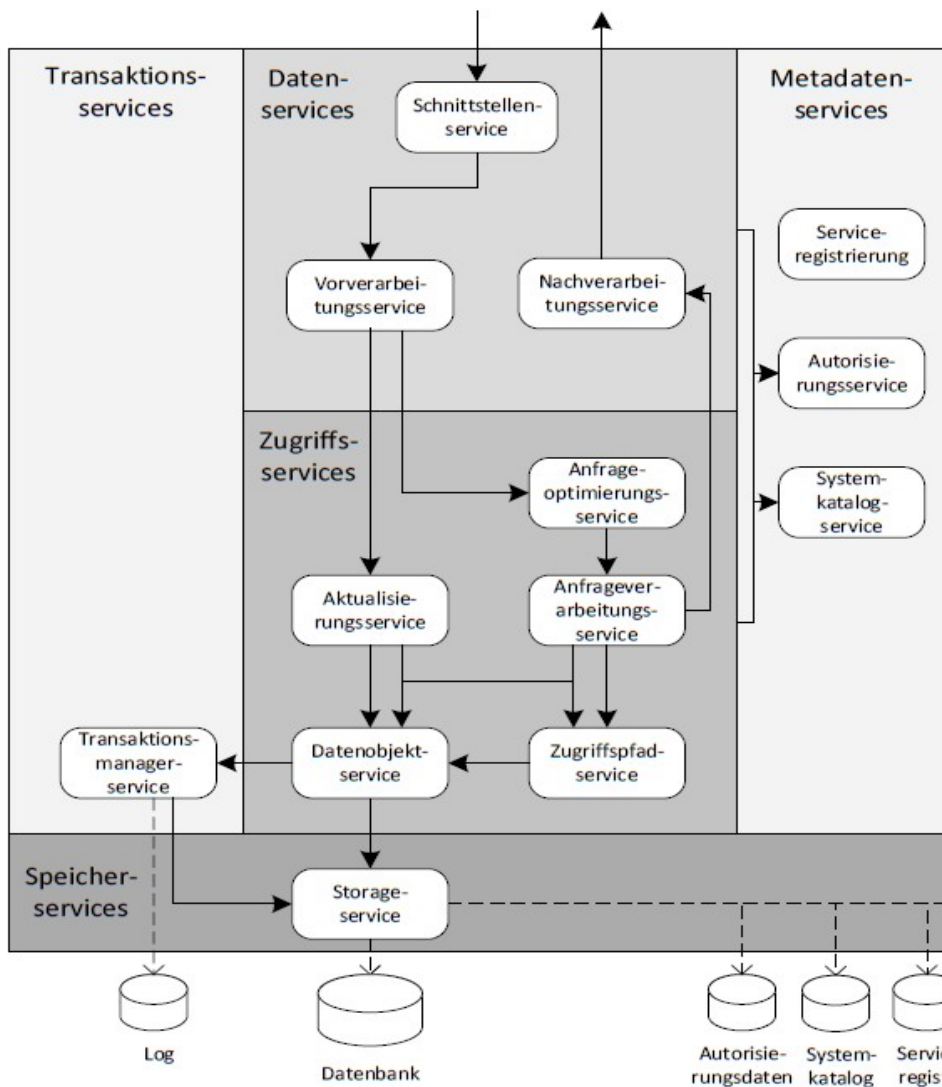
Vergleich

	DaaS	lokale Datenbanl
Kosten für Hard - und Software	+	-
Kosten für Betrieb und Administration	+	-
Skalierbarkeit	+	-
Geschwindigkeit	0	+
Flexibilität	+	0
Datenschutz und Datensicherheit	0	+
Vertrauen zu Fremdfirmen	-	0

Database-as-a-Service Model

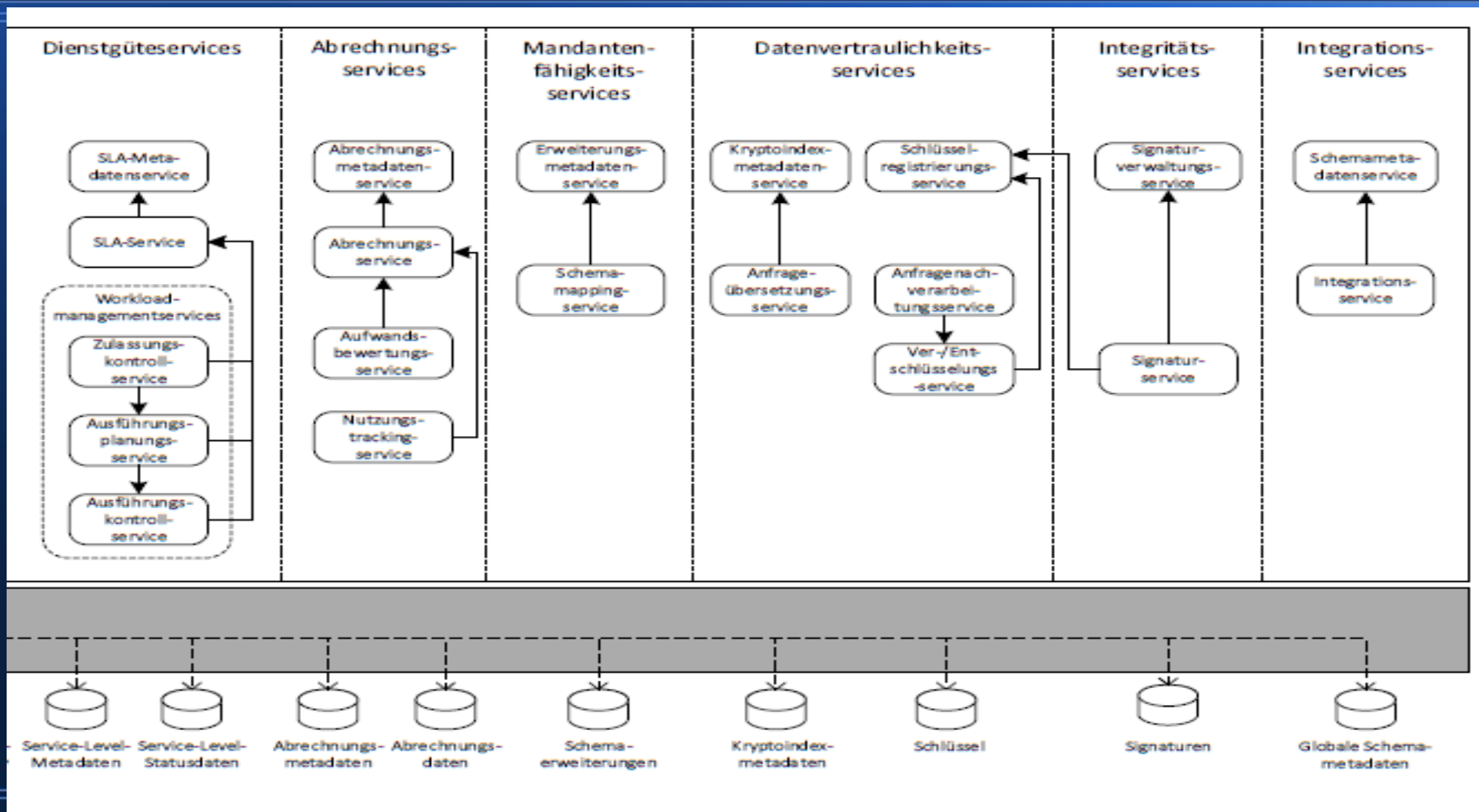
- Das Service-based Datamanagement System Modell gliedert Services in 4 Gruppen
- 1.) Die Storage Services die auf byteebene arbeiten in Zusammenarbeit mit Dateisystemfunktionen
- 2.) Die Access Services sind für physische Repräsentation der Datensätze verantwortlich
- 3.) Die Data Services bieten Zugriff auf Daten in logischen Strukturen wie Tabellen oder Sichten
- 4.) Die Extension Services sind Services zur Erweiterung von z.B. neuen Datentypen bereitstellen
- Greifen auf übrige Services zurück

Database-as-a-Service Model



- Model in zwei Teile gegliedert
- Besteht aus Basisservices und erweiterten Services
- Basisservice umfasst Speicher-, Zugriffs-, und Datenservice
- Hinzu kommen Transaktionsservices
- Metadaten-services für Metadaten und Zugriffsrechte

Database-as-a-Service Model



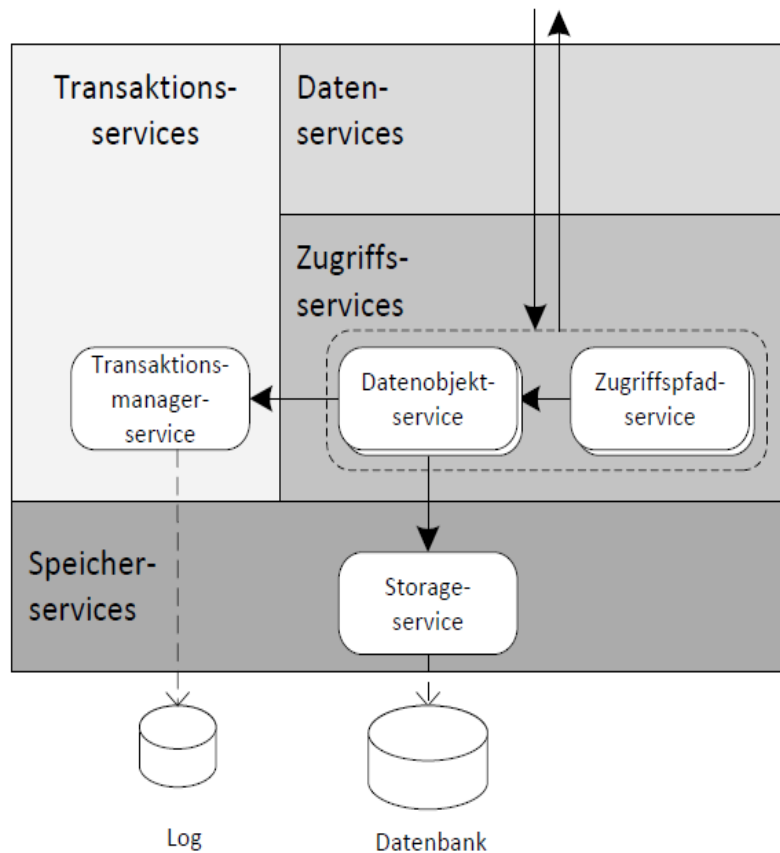
Adaption des Modells an Forschungsprototypen

- Implementierung verteiltem Datenbanksystems auf Basis von Amazon S3 und SQS
- Datenbanksystem soll Speicherung und Abruf von Daten ermöglichen
- S3 ermöglicht Speicherung schlüsselindexierter BLOB's
- Dienen als Seitengeber
- SQS dient zur Transaktionsverwaltung durch Message-Queues
- Übrige Funktionalität durch clientseitige Softwareschicht bereitgestellt
- Datensatz besteht aus beliebiger Bytefolge und einem Schlüssel
- Datensatz wird als Kollektion abgelgt
- Interne Datrstellung als B^+ Baum \rightarrow Baum wird in S3 gespeichert

Adaption des Modells an Forschungsprototypen

- Es können sekundäre B⁺ Bäume angelegt werden die Attribute indexieren
- Datensätze können durch Angabe von Schlüssel – oder Attributwerten unter Angabe von Kollektion abgerufen werden
- Schreiboperationen werden über SQS koordiniert
- Es werden 2 SQS-Queues zugeordnet
- Erste dient Speicherung von Sperrtoken
- Zweite dient zur Protokollierung der Änderungen
- Erst wird Sperrtoken und dann Protokollierung durch Client abgerufen
- Bei Erfolg wird Protokolleintrag gelöscht und Sperrtoken freigegeben

Adaption des Modells an Forschungsprototypen

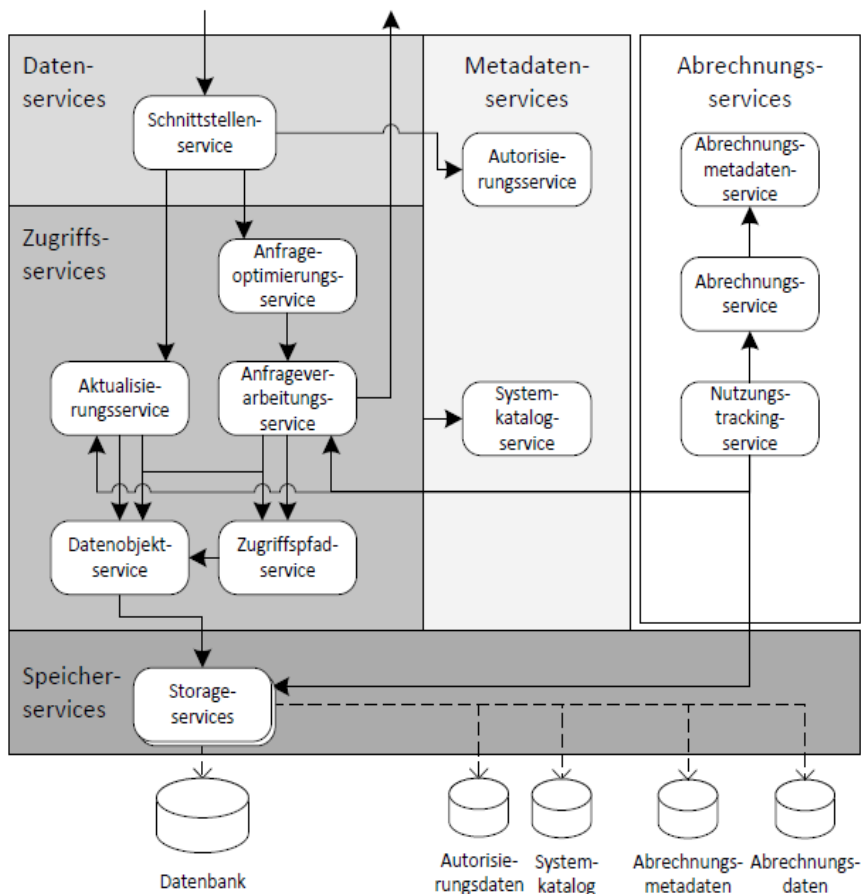


- Speicherservice durch S3 realisiert
- Transaktionsservice dient Protokollierung durchzuführender Operationen
- Verwaltung von Sperren
- Durch SQS realisiert
- Datensatzverwaltung u. Verwaltung sek. Zugriffsstrukturen sind clientseitige Funktionalität
- Datenobjekt-service u. Zugriffspfad-service durch gemeinsame Schnittstelle sichtbar nach außen

Adaption des Modells an Amazon Simple DB

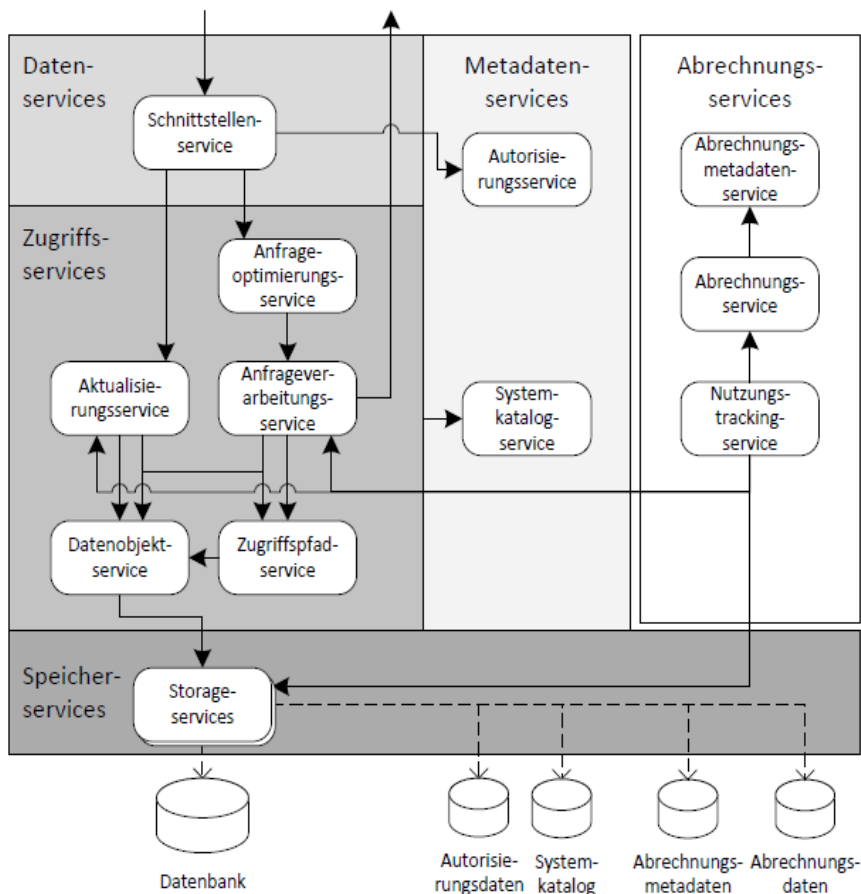
- Amazon Simple DB ist Webservice zur Speicherung untypisierter Multi-Maps (Items)
- Service stellt Einfüge-, Modifizier-, lösch-, und Amfrage Operationen zur Verfügung
- Gespeicherte Werte werden automatisch indexiert
- Anfragen mittels Simple-DB spezifischer Anfragesprache
- Zugriff nur für authentifizierte und autorisierte Nutzer
- Simple DB verfolgt den Eventual Consistency Ansatz, d.h. Konsistenter Zustand gesichert man weis nur nicht wann
- schreibende Operationen auf genau 1 Item

Adaption des Modells an Amazon Simple DB



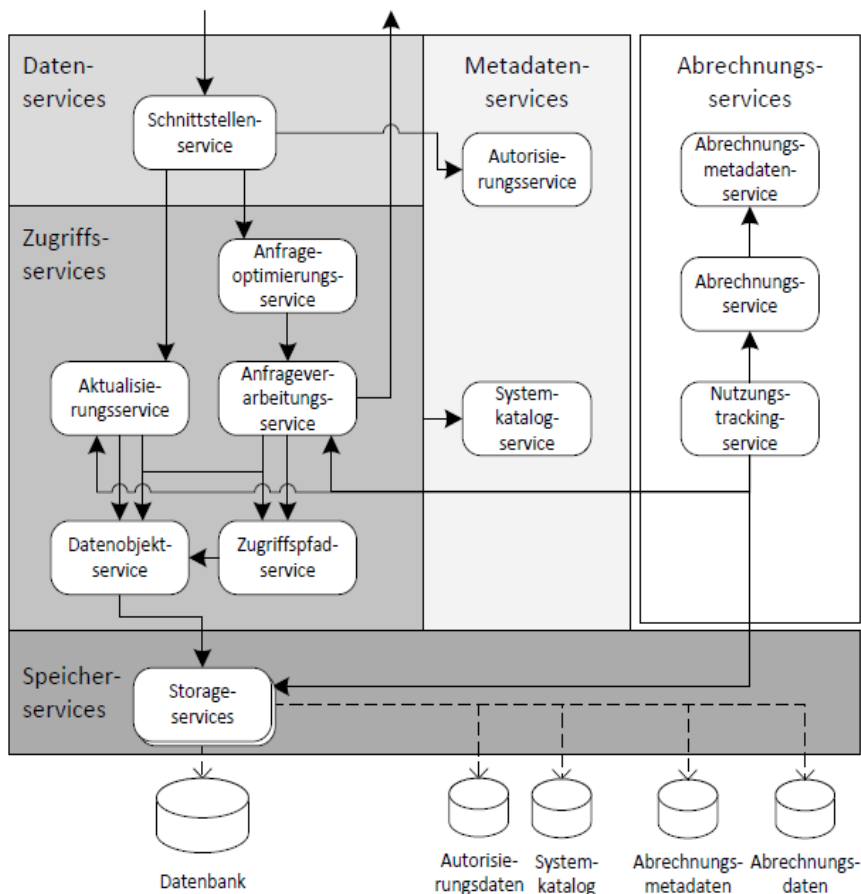
- Es existieren keine Sperrmechanismen → die mit jüngstem Zeitstempel erlangt Gültigkeit
- Abrechnung erfolgt nutzungsbasiert anhand Anzahl ausgeführter Operationen und übertragener Datenmenge

Adaption des Modells an Amazon Simple DB



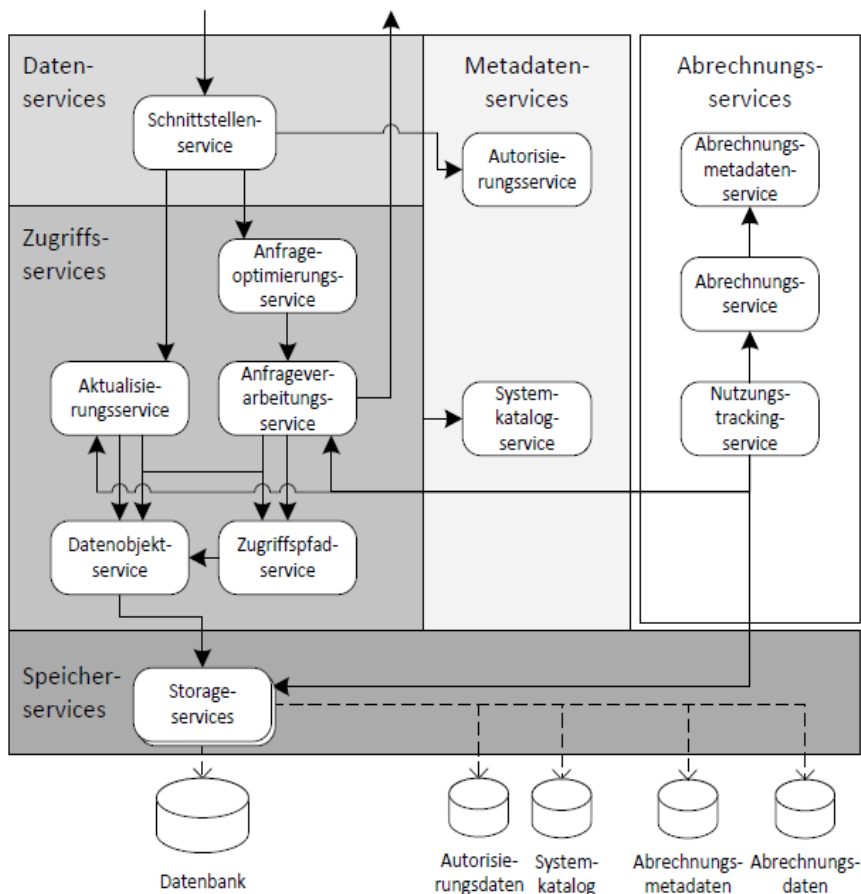
- Schnittstellenservice kapselt interne Services → einziger Service der extern erreichbar
- Operationen werden vor Ausführung durch Autorisierungsservice authentifiziert und autorisiert
- Schreibende Operationen gehen an Aktualisierungsservice
- Erst Anfrageoptimierungsservice dann Anfrageverarbeitungsservice

Adaption des Modells an Amazon Simple DB



- Datenobjektsservice ermöglicht Zugriff auf Items
- Bei Aktualisierung erstellten Indexe auf Zugriffspfadservice verwaltet
- Können in Anfrageverarbeitung mit einbezogen werden
- Daten werden repliziert gespeichert → auf ebene der Speicherservices existieren mehrere Stageservices

Adaption des Modells an Amazon Simple DB



- Aufgrund nutzungsbasierter abrechnung beschränkt sich Abrechnungsservice auf zentralen Abrechnungsservice, Abrechnungsmetadaten-service und Nutzungstracking-service

Zusammenfassung

- Database-as-a-Service bietet viele Vorteile im Vergleich zu lokal installierten Datenbanken
- Es müssen aber noch einige Aspekte wie Mandatenfähigkeit geklärt werden
- Es gibt bereits bestehende Ansätze die verdeutlichen wie Weit die Entwicklung fortgeschritten ist

Vielen Dank für die Aufmerksamkeit

Literatur

[AFG+09] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds: A berkeley view of cloud computing. Technical report, University of California at Berkeley, February 2009.

[ASS+09] Ashraf Aboulnaga, Kenneth Salem, Ahmed A. Soror, Umar Farooq Minhas, Peter Kokosielis, and Sunil Kamath. Deploying database appliances in the cloud. IEEE Data Eng. Bull., 32(1):13–20, 2009.

[ea00] Fred Hoch et al. Software as a service: Strategic backgrounder. 2000.

[KK09] Fabian Panse Norbert Ritter Kathleen Krebs, Marc Holze. Konfiguration und Spezifikation bedarfsgerechter Dienstleistungen zur Datenverwaltung. <http://dbs.unileipzig.de/file/BTW2009.pdf>, 2009.

[Lan09] Jens Lansing. Database-as-a-Service: Charakterisierung und Modellentwurf. <http://dbs.uni-leipzig.de/file/BTW2009.pdf>, 2009.

[MB08] David Graf Donald Krossmann Tim Kraska Matthias Brantner, Daniela Florescu. Building a Database in the Cloud. <http://www.dbis.ethz.ch/research/publications/dbs3.pdf>, 2008.