Transaktionsmodelle

Christian Ziemann

Fachbereich Informatik

Lehrgebiet Datenverwaltungssysteme

Gliederung

- Geschachtelte Transaktionen
  - Geschlossen geschachtelte Transaktionen
  - Offen geschachtelte Transaktionen
- Geschäftstransaktion
  - Geschäftsprozess
  - Business Transaction Framework
  - Atomaritätsebenen
- ACID-Transaktionen in Geschäftstransaktionen

ACID-Eigenschaften

Bisher : ACID in “flachen Transaktionen“

Atomarität
Konsistenz
Isolation
Dauerhaftigkeit

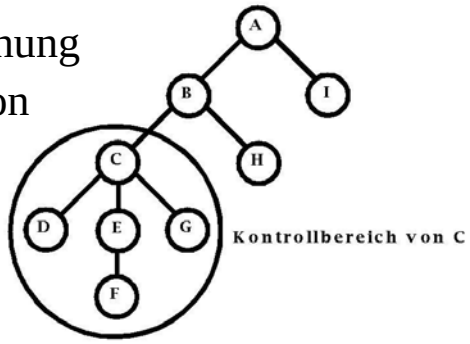
Sichergestellt durch TP-Monitore

Geschlossen geschachtelte Transaktionen

- Langlebige Transaktionen
- Hierarchie von Sub-Transaktionen
- Modularisierung von Anwendungsfunktion
- Fein-granulare Fehlerbehadlung
- Intra-Transaktionsparallelität

Aufbau einer ggT

- Transaktionsbaum
  - Knoten = Transaktion / Subtransaktion
  - Kanten = statische Aufrufbeziehung
  - Wurzel = Top-Level-Transaktion
  - Vorgänger = Vater
  - Nachfolger = Kinder
  - Kontrollbereich von C = Sub-Transaktion mit Wurzel C



Merkmale ggT

- Isolierte Rücksetzbarkeit
  - Fehler in TA T => Rücksetzen von T und Transaktionen im Kontrollbereich von T (Vater unverändert)
  - Vater trägt Kontrolle
- Kontrollstruktur
  - Kontrollstruktur zu parallelen und verteilten Ausführen
- Kontrolliertes Zurücksetzen
  - Fehler in TA T => Rücksetzen von T und Transaktionen im Kontrollbereich von T
- Recovery mittels zustandsorientiertes Undo-Recovery

ACID in ggT

- Rücksetzregel
  - Wird TA zurückgesetzt →Alle TA‘s im Kontrollbereich zurückgesetzt
- Commit-Regel
  - Lokale Commit macht Änderungen der Vater-TA sichtbar
  - Endgültiges Commit erfolgt, wenn alle Vorfahren Commit ausgeführt haben

ACID in ggT (2)

- Sichtbarkeitsregel
  - Nach Commit von Sub-TA→Änderungen werden Vater sichtbar gemacht
  - Objekte, die Vater TA sichtbar gemacht wurde, können Sub-TA‘s zugänglich gemacht werden
  - Änderungen für gleichzeitig aktive Sub-TA‘s nicht sichtbar

Aus Commit- und Rücksetzregel → Atomarität für Sub-TA

Sichtbarkeitsregel → Isolation parallel laufender Sub-TA

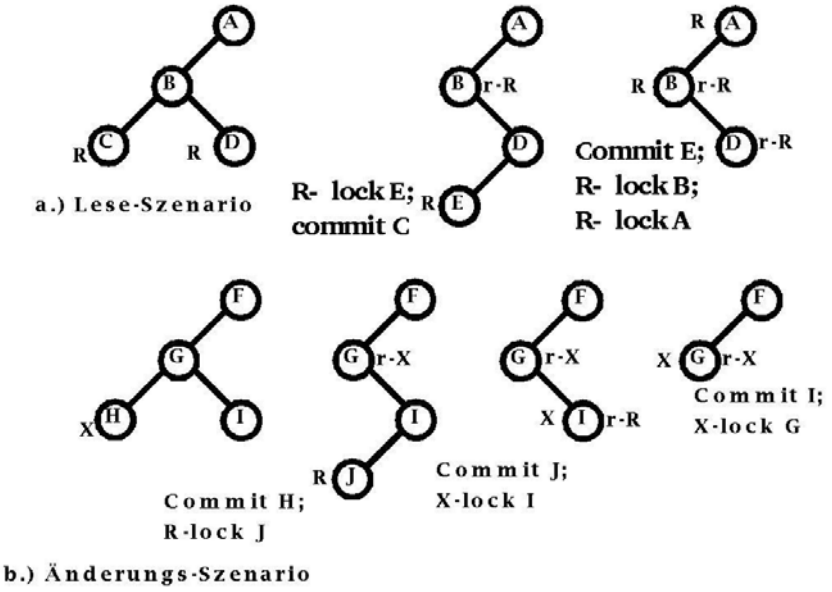
Intra-Transaktionparallelität

- Weder Vater/Kind noch Geschwister-Parallelität
- Nur Geschwister-Parallelität
- Nur Vater/Kind Parallelität
- Vater/Kind-Parallelität und Geschwister-Parallelität

Synchronisation ggT

- Beim Commit von T erbt Vater-TA Sperren
- Wird T abgebrochen, so werden alle Sperren freigegeben
- TA T kann eine X-Sperre anfordern, falls
  - keine andere TA eine X- oder R-Sperre auf dem Objekt besitzt
  - alle TA‘s die eine r-R-Sperre oder r-X-Sperre auf dem Objekt halten, Vorfahren von T sind
- TA T kann eine R-Sperre anfordern, falls
  - keine TA besitzt X-Sperre auf diesem Objekt
  - alle TA‘s, die eine r-X-Sperre auf dem Objekt halten, Vorfahren von T sind

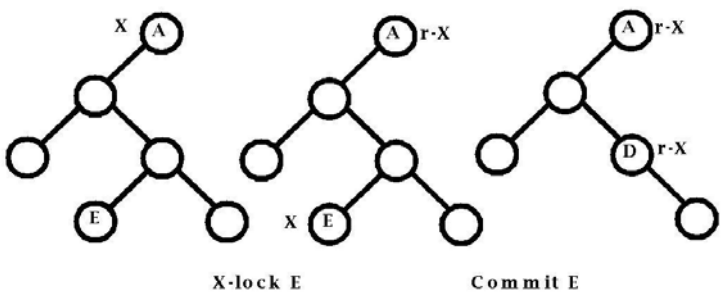
Bsp.



Weiterreichen von Sperren

- Lastet auf eine TA T z.B. eine X-Sperre, so wird diese Sperre an die Sub-TA weitergereicht
- Sperre X auf T wird zu r-X

Bsp.:



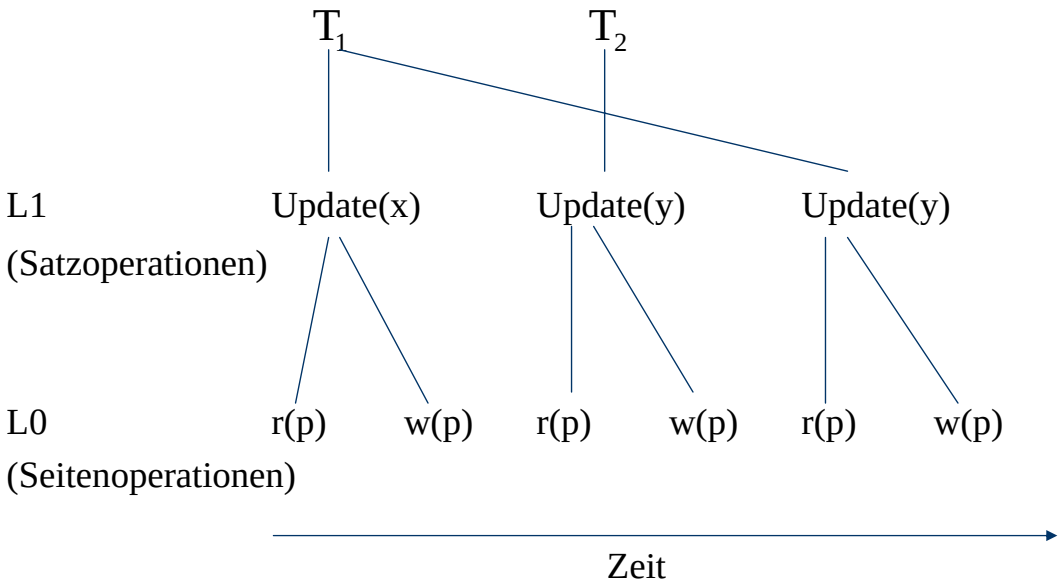
Offen geschachtelte Transaktionen

- Inter/Intra-Transaktionsparallelität
- Kurze Sperrzeiten
- Änderung einer Sub-TA augenblicklich sichtbar
- Synchronisation erforderlich
- Kompensierende Transaktionen

Mehrebenen Transaktionen

- Nutzt Objekthierarchie
- Feste Anzahl der Schichten
- Operationen von Ebene i realisiert durch Ebene i-1
- Schachtelungstiefe gleich
- Geringe Synchronisationskonflikte
- Auf jeder Ebene: Logging- und Recovery-Funktionen

Mehrebenen Transaktionen(2)



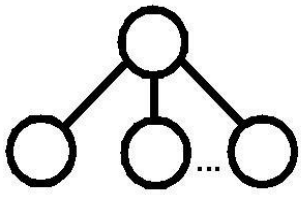
Sagas

Sequenz von TA‘s + kompensierenden TA‘s

$$[(T_1, C_1), (T_2, C_2), ..., (T_{n-1}, C_{n-1}), (T_n, C_n)]$$

Top-Level-TA: CD

Sub-TA: ACID, serialisierbar



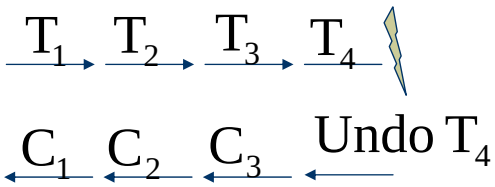
Sagas(2)

Ausführung:

$T_1, T_2, \ldots, T_{n-1}, T_n$

Abbruch von T_4

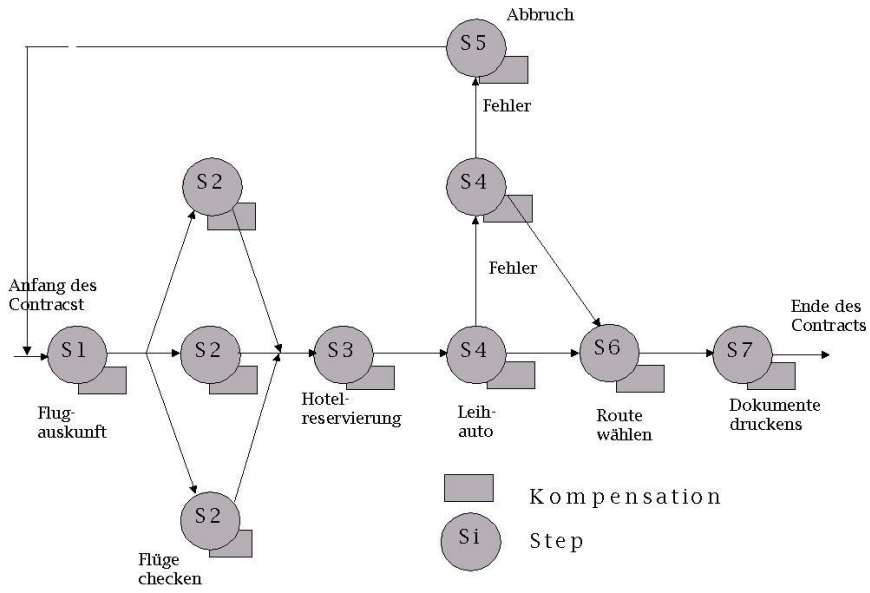
Undo T_4, C_3, C_2, C_1



Contracts

- Erweiterung des Saga-Konzepts
- Ablaufkontrolle (Skript)
- Sub-TA (Step)
- Reiche Kontrollstruktur
  - Sequenz, Parallelität, Verzweigung, etc.
- Kompensierende TA‘s
- Synchronisation über Invarianten
- Persistentes Kontext-Management

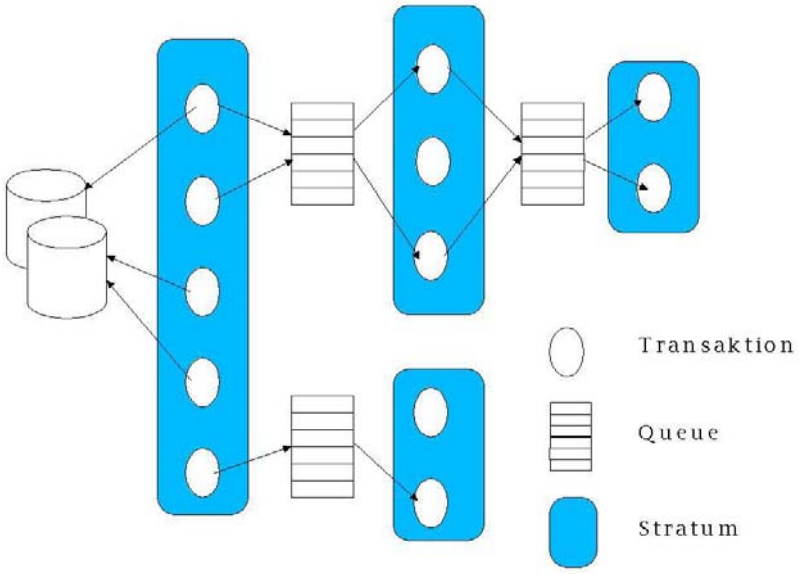
Contracts (Bsp.)



Stratifizierte Transaktionen

- Zerlegung von Transaktion T in $T_1, T_2, ..., T_{n-1}, T_n$
- Jedes T_i mit persistente Warteschlange Q_i verbunden
- Nebenläufige TA‘s in Strata $S_1, ..., S_m$ zusammenfassen
  - $S_i \subseteq_m T$ mit $S_i \neq 0$ und $S_i \cap S_j = \{\}$ für $i \neq j$ und $\bigcup_{j=1} S_j = T$

Stratifizierte Transaktionen(2)



Recovery in ogT

- Anwendungsspezifische kompensierende Sub-TA‘s
- Nicht zwingend genauer Umkehrprozess
- Neue DB-Zustand gleicht nicht altem DB-Zustand

Probleme kompensations-basierter Fehlerbehandlung

- Hoher Aufwand
  - Kompensationsprogramm für jede Sub-TA
    - Hohe Flexibitilät
    - Enormer Aufwand
    - Hohe Mitverantwortung des Programmierers
- Scheitern der Kompensation
  - Kompensationen dürfen nicht scheitern
  - Sicherheitslücke nach mehrfacher Wiederholung von Kompensationen → manuelle Fehlerbehandlung
- Nicht kompensierbare Operationen
  - Nicht alle Operationen kompensierbar
  - Nicht kompensierbare Operarionen zum Schluss

Geschäftsprozesse

- Langlebige Transaktionen
- Geschäftsprozess = Menge von logisch zusammenhängender Aufgaben
  - Physisches Resultat
  - Leistung
- Aktivität realisiert definierte Funktion
  - Einfach
  - Komplex
- Geschäftsregeln = Ansammlung von Statements
- Geschäfts-Prozess-Management unterstützt:
  - Definition von langlebigen TA‘s
  - Ausführung von langlebigen TA‘s
  - Kontrolle von langlebigen TA‘s

Business Transaction Framework

Unterstützung von

- Geschäftstransaktionsmodellen
  - Langlebige TA‘s, herkömmliche TA‘s, Exception-Handling-Mechanismen, kompensierende Aktionen, Atomarität
- Koordinationsprotokollen
  - Koordination von verteilten Anwendungen
  - Bietet einer Aktivität den benötigten Kontext, um mit anderen Services zu interagieren
- Geschäftsprotokollen
  - Definition von Sende/Empfangs-Folge
  - Inhalt der Nachricht

Geschäftstransaktionsmodell

Geschäftstransaktionen haben folgende Eigenschaften:

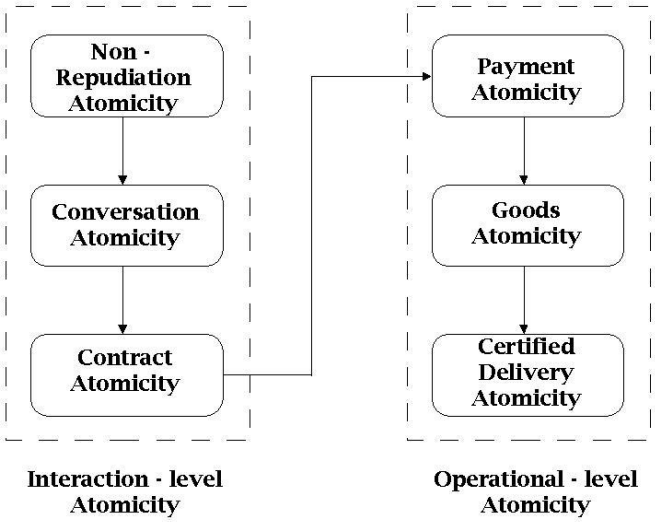
- Repräsentieren Funktion ähnlich dem Geschäft
- Beziehen mehrere Parteien / Unternehmen ein
- Definieren Kommunikationsprotokollverbindungen
- Basieren auf formalen Abkommen

Geschäftstransaktionsmodell(2)

Geschäftstransaktion enthält:

- Verhandlungen
- Verpflichtungen
- Verträge
- Logistik
- Verschiedene Bezahlungsinstrumente
- Exception-Handling

Atomaritätsebenen



Atomaritätsebenen(2)

- Business Interaction-level Atomicity
  - Non-repudiation Atomicity

    Verbindlichkeit der TA, Überprüfbar durch Historie
  - Conversation Atomicity

    Korrelation von Folgen von Anforderungen der Parteien. Möglichkeit zur Rücksetzung auf konsistenten Zustand
  - Contract Atomicity

    Bindet bis zu n Partner, Verträge definieren gesetzliche Bedingungen und technische Spezifikationen

Atomaritätsebenen(3)

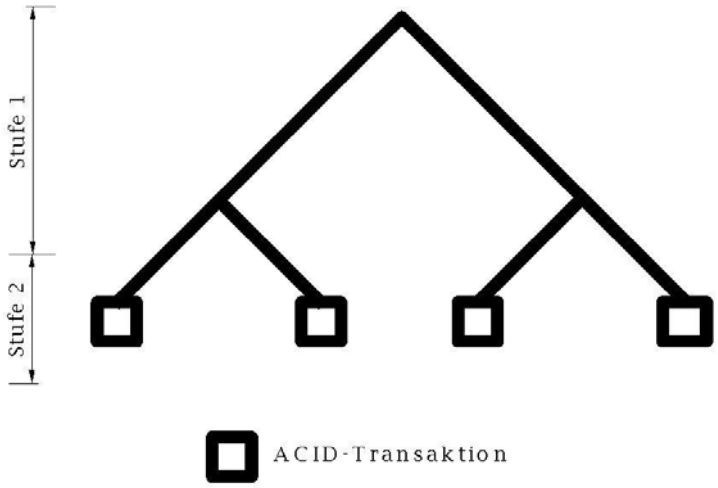
- Operational-level Atomicity
  - Payment Atomicity

    Transfer von Geld, kein erzeugen oder vernichten von Geld
  - Goods Atomicity

    Vollständiger Austausch von Geld und Ware, grundlegende Idee: Lieferung und Geldtransfer in Geschäftstransaktion
  - Certified Delivery Atomicity

    Garantiert das Liefern der richtigen Ware

ACID-TA‘s in Geschäftstransaktionen



Zusammenfassung

- Geschachtelte Transaktionen
  - Geschlossen geschachtelte Transaktionen
  - Offen geschachtelte Transaktionen
- Geschäftstransaktion
  - Geschäftsprozess
  - Business Transaction Framework
  - Atomaritätsebenen
- ACID-Transaktionen in Geschäftstransaktionen