

Verteilte Transaktionen mit Java

Worauf Ihr vorbereitet sein solltet

frank.werres@msg-life.com

Transaktionen

Eigenschaften 1

Atomar - Alles oder Nichts

Konsistenz - Keine Konsistenzregeln werden verletzt

Transaktionen

Eigenschaften 2

Isolation - Erst nach positivem Abschluss sichtbar

Dauerhaft - Ergebnis abfragbar

Atomic, Consistent, Isolated, Durable: ACID-Properties

Datenbank-Transaktionen



Klammerung logisch zusammenhängender Änderungen (aka SQL-Befehle)

ACID-Properties

Atomar

(Konsistenz)

(Isoliertheit)

(Dauerhaft)

Transaktionen im ApplikationServer

Steuerung durch Annotationen

„Atomare“ Methoden

EJB: @TransactionAttribute(MANDATORY | NEVER | REQUIRED ...) an Methoden

Spring: @Transactional

Transaktionen im ApplikationServer



Erweiterungen des Datenzugriffs

`javax.persistence.PersistenceException:`

`org.hibernate.exception.GenericJDBCException: Cannot open connection ...`

...

Caused by: `javax.transaction.SystemException: java.lang.Throwable: Unable to enlist resource, see the previous warnings. tx=TransactionImple...`

“This can happen when you have more than one datasource ... and no XA datasources configured ...”

(Alternative Ideen mit Hilfe des DBMS führen zum gleichen Ergebnis)

Transaktionen mit mehreren Beteiligten



XA-Ressourcen

XA: Transactional Architecture

Transaktionierende Ressourcen-Manager (Datenbanken, Dateisysteme, ...)

Koordinationsinstanz: Transaktions-Manager

JEE-Standard: JTA (Java Transaction API)

Transaktionen mit mehreren Beteiligten

Synchronisationsprotokoll

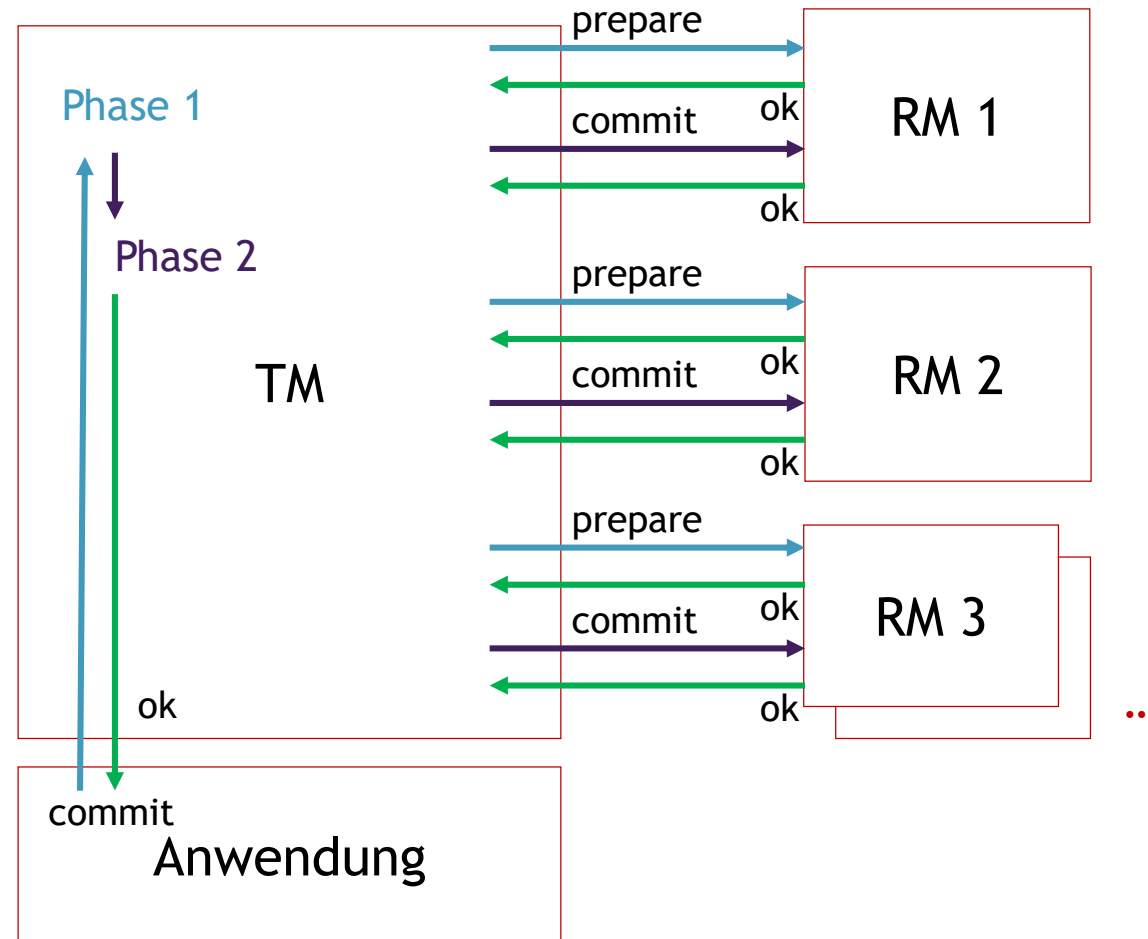
2PC - X/Open

Aufteilung der TX-Bearbeitung in 2 Phasen (sic!)

Pre-Commit-Phase + Commit-Phase

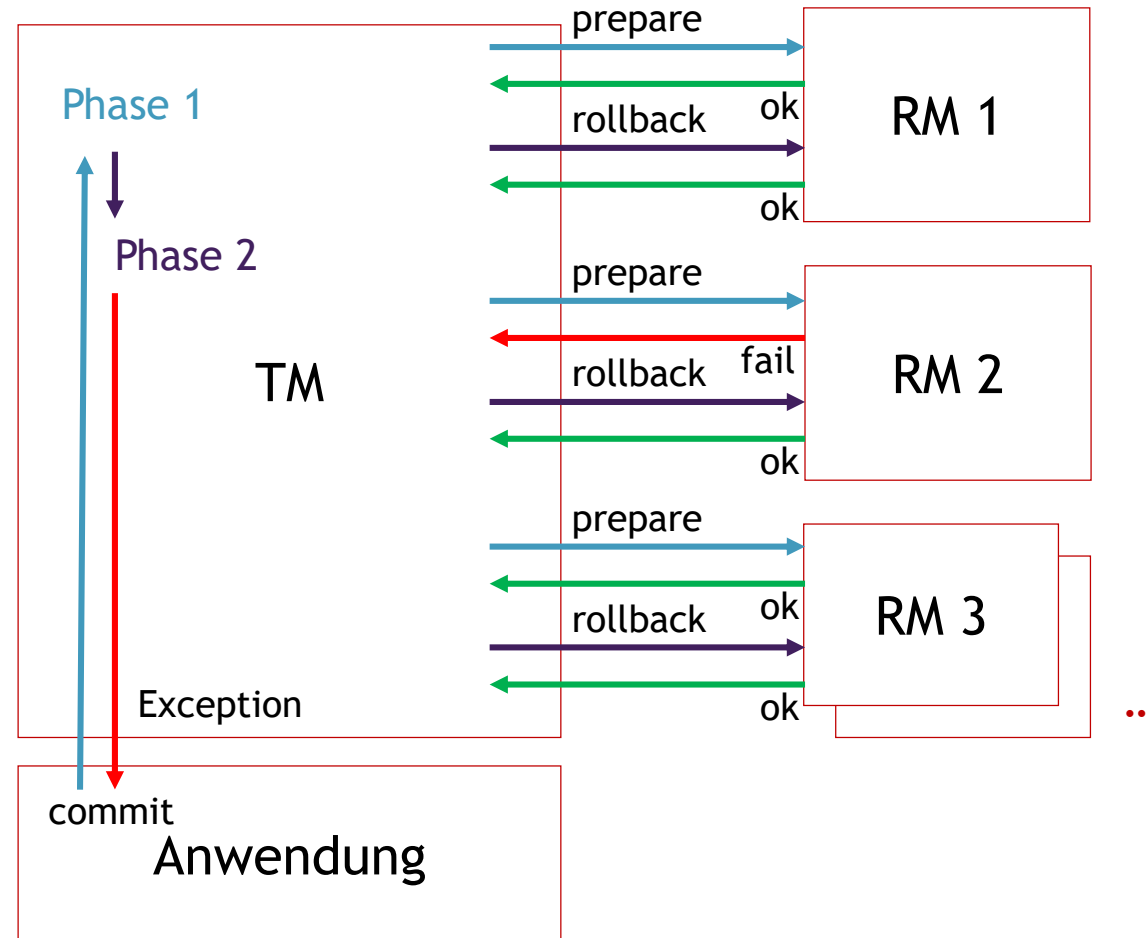
2PC

Ablauf bei Schönwetter



2PC

Ablauf bei Regen



2PC

Spezifizierte ‚Unzuverlässigkeit‘

2.3.3 Heuristic Branch Completion

„Some RMs may employ heuristic decision-making: an RM that has prepared to commit a transaction branch may decide to commit or roll back its work independently of the RM. It could then unilaterally release resources. This may leave them in an inconsistent state. When the RM ultimately directs an RM to complete the branch, the RM may respond that it has already done so. The RM reports whether it committed the branch, rolled it back, or completed it with mixed results (committed some work and rolled back other work).”

2PC

Unzuverlässigkeit nicht zu vermeiden

Nachrichten-Verlust

Notwendigkeiten aus Ressourcen-Situation, Deadlock-Erkennung, Betrieb, ...



Heuristic-Exceptions



Unzuverlässigkeit nicht zu vermeiden

JSE 6 & 7

```
javax.transaction.HeuristicCommitException
```

“This exception is thrown by the rollback operation on a resource to report that a heuristic decision was made and that all relevant updates have been committed.”

```
javax.transaction.HeuristicRollbackException
```

“This exception is thrown by the commit operation to report that a heuristic decision was made and that all relevant updates have been rolled back.”

Heuristic-Exceptions

Unzuverlässigkeit nicht zu vermeiden

```
javax.transaction.HeuristicMixedException
```

“This exception is thrown to report that a heuristic decision was made and that some relevant updates have been committed and others have been rolled back.”

Heuristic-Exceptions

Unzuverlässigkeit nicht zu vermeiden

Seit Java 7

`javax.transaction.xa.XAException`

`XA_HEURCOM, XA_HEURRB, XA_HEURMIX`

`XA_RBCOMMFAIL, XA_RBDEADLOCK, XA_RBTIMEOUT`

`XA_HEURHAZ` – The transaction branch may have been heuristically completed

`XA_RBROLLBACK`

– Indicates that the rollback was caused by an unspecified reason

`XA_RBTRANSIENT` – May retry the transaction branch

`XA_RETRY` – Routine returned with no effect and may be reissued

Ergebnis

Kaputte Transaktionen







InDoubt-Transaktionen: Manchmal auflösbar, manchmal nicht!

[Application servers](#) > [server1](#) > [Transaction service](#) > **Transactions with heuristic outcome**

Use this page to review transactions that completed with a heuristic outcome.

⊕ Preferences

Clear

Select	Local ID	Heuristic outcome	Global ID
You can administer the following resources:			
<input type="checkbox"/>	3	11	00000138B462CCB9000000003000000003F808FE26
<input type="checkbox"/>	4	11	00000138B4666725000000003000000004F808FE26
Total 2			

Ergebnis

Auflösbare InDoubt-Situationen

Abhängig vom Ressourcen-Manager

Produkt-spezifische Verfahren

Stichwort: Datenbank-Warmstart

Zusammenspiel aller beteiligten Komponenten notwendig !! Komplexe Sache !!



Zusammenfassung

Unzuverlässigkeit nicht zu vermeiden

2PC nicht absolut sicher

Inkonsistente Datenzustände möglich, erfordern fachl. Kompensation/Korrekturen

Ergebnis der Logik - prinzipbedingt - unvermeidbar

Nicht Ergebnis eines bewussten oder fahrlässigen Spezifikationsfehlers

Schlussfolgerungen

Unzuverlässigkeit nicht zu vermeiden

Wiederanlaufverfahren für Ressourcen-Manager ausprobieren

FIT

Fachliche Kompensationsszenarien entwickeln

2PC ist kein „Silver Bullett“!

Lasst es sein!

Sucht eine Alternative!

LASST ES SEIN!

**Vielen Dank für
Eure Aufmerksamkeit.**

Frank Werres

frank.werres@msg-life.com