

# Replikation in Datenbanken

Vortrag: Datenbanken II

Yves Adler

Hochschule für Technik, Wirtschaft und Kultur Leipzig  
Fachbereich Informatik, Mathematik und Naturwissenschaften

- 1 Einführung  
Anwendungsszenarien
- 2 Grundlegende Konzepte  
asymmetrische vs. symmetrische Replikation  
synchrone vs. asynchrone Replikation  
Konfliktauflösung
- 3 Replikation in der Praxis  
Replikation mit Oracle 10g  
Spezielle Replikationsverfahren
- 4 Quellen

- 1 Einführung  
Anwendungsszenarien
- 2 Grundlegende Konzepte  
asymmetrische vs. symmetrische Replikation  
synchrone vs. asynchrone Replikation  
Konfliktauflösung
- 3 Replikation in der Praxis  
Replikation mit Oracle 10g  
Spezielle Replikationsverfahren
- 4 Quellen

- 1 Einführung  
Anwendungsszenarien
- 2 Grundlegende Konzepte  
asymmetrische vs. symmetrische Replikation  
synchrone vs. asynchrone Replikation  
Konfliktauflösung
- 3 Replikation in der Praxis  
Replikation mit Oracle 10g  
Spezielle Replikationsverfahren
- 4 Quellen

- 1 Einführung  
Anwendungsszenarien
- 2 Grundlegende Konzepte  
asymmetrische vs. symmetrische Replikation  
synchrone vs. asynchrone Replikation  
Konfliktauflösung
- 3 Replikation in der Praxis  
Replikation mit Oracle 10g  
Spezielle Replikationsverfahren
- 4 Quellen

# Einführung

**Oracle:** *„Replication is the process of copying and maintaining database objects, such as tables, in multiple databases that make up a distributed database system.“* (Ora07a)

**Replikation:** Als Replikation (im Zusammenhang mit Datenbanken) wird der Prozess zur mehrfachen Datenspeicherung und Datenpflege in (verteilten) Datenbank- bzw. Informationssystemen bezeichnet.

Allerdings entsteht auch ein nicht unerheblicher Mehraufwand beim Design, der Implementierung und der Wartung des Datenbanksystems.

# Gründe für den Einsatz von Replikation

## Verfügbarkeit/Recovery

Der Ausfall einer Kopie der Daten stellt keinen „Totalausfall“ des Gesamtsystems dar, sollten diese Daten repliziert auf einem anderen Server zur Verfügung stehen.

## Performance

Replikation *kann* schnellen Zugriff auf Daten gewährleisten, so ist es möglich die Last auf mehrere Server zu verteilen und dadurch die Last des Gesamtsystems zu reduzieren. Ein Nutzer greift dann typischerweise auf eine lokale oder Netztopologisch „nahe“ Kopie der Daten zu, dadurch werden die lokalen Antwortzeiten auf eine Anfrage reduziert. Dies betrifft jedoch zumeist nur Lesetransaktionen.

# Gründe für den Einsatz von Replikation

## Disconnected Computing

Ein Nutzer kann bestimmte Daten lokal replizieren, mit Ihnen *Offline* arbeiten und später wieder mit dem Gesamtsystem synchronisieren.

## Reduzierung der Netzwerklast

Sind die Daten über verschiedene Geografische Lokationen (Bsp: Niederlassungen einer Bank) verteilt, so kann Replikation die Netzwerklast im Vergleich zu einem zentralisierten System drastisch reduzieren.

# symmetrische Replikation

Bei der Symmetrischen Replikation wird ein Update auf allen Rechnern, bzw. auf allen Replikaten des Replikationsverbundes durchgeführt, man spricht auch von „Update anywhere“/„Update everywhere“.

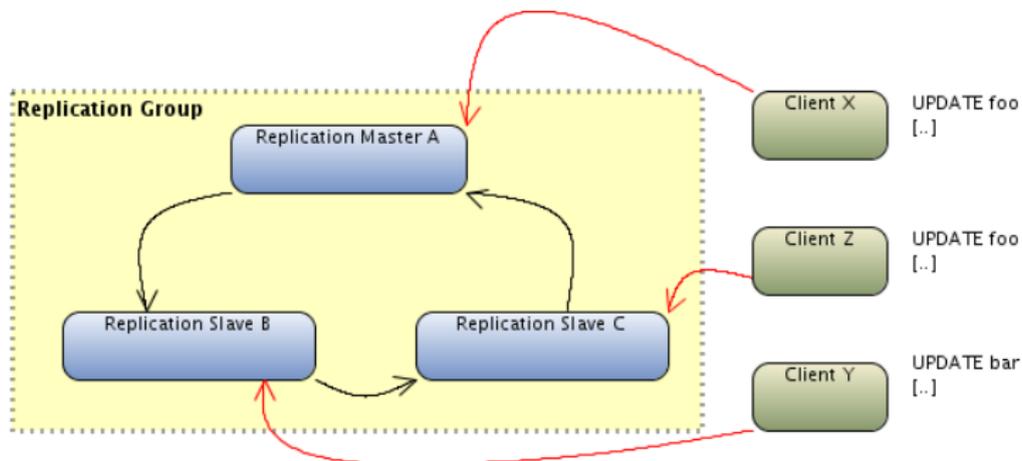


Abbildung: Symmetrische Replikation

# asymmetrische Replikation

Bei einer Asymmetrischen Replikation wird ein Update der Daten nur auf einem Rechner(verbund)/Replikat, der sogenannten „Primary Copy“ bzw. dem „Update Master“, zugelassen und durchgeführt. Dieser Rechner(verbund) propagiert nach erfolgtem Update die Änderungen an den Rest des Replikationsverbundes.

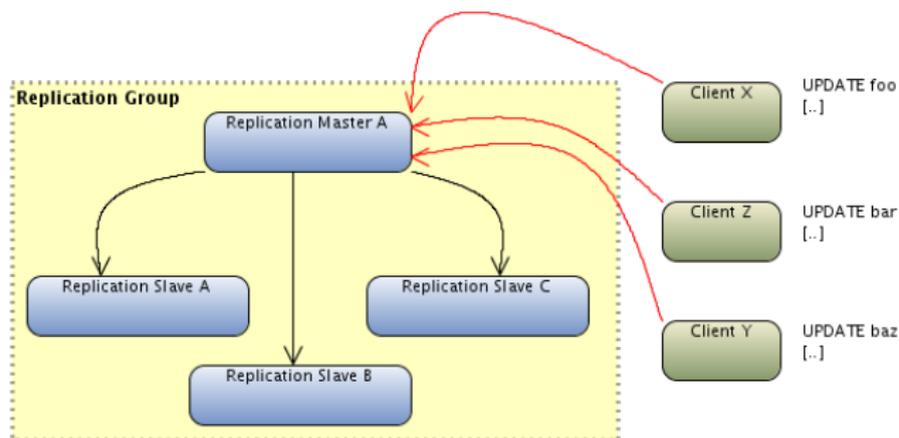


Abbildung: Asymmetrische Replikation

# synchrone/asynchrone Replikation

## synchrone Replikation („eager“/„pessimistic“ Replication)

Bei der synchronen Replikation wird eine Transaktion zeitgleich auf allen Replikaten durchgeführt und ist erst dann erfolgreich beendet, wenn die Transaktion auf allen Replikaten erfolgreich durchgeführt wurde. Sollte auf einem Replikat die lokale Transaktion scheitern, so ist auch die globale Transaktion gescheitert.

## asynchrone Replikation („lazy“/„optimistic“ Replication)

Bei der asynchronen Replikation ist es erlaubt, dass sich die Replikate unterscheiden, d.h. es wird lediglich sichergestellt dass das replizierte Datenbankobjekt konvergiert.

# Allgemeine Konfliktauflösung

- Timestamp basierte Verfahren:
  - „*Latest Timestamp*“:  
Tritt ein Konflikt zwischen zwei Replikaten bei einem Timestamp basierten Replikationsverfahren auf, so wird das Replikat mit dem „jüngeren“ Timestamp gewählt. Diese Strategie ist zwar Problembehaftet, erfüllt aber trotzdem das Konvergenzkriterium.
  - „*Earliest Timestamp*“:  
Stellt das Gegenteil der „Latest Timestamp“ Methode dar, wählt also das „älteste“ Replikat.
- Master/Slave - „*Mastercopy wins*“:  
Bei Konflikten gewinnt immer das Replikat auf dem Master-Server, dies kommt häufig beim „Disconnected Computing“ zum Einsatz.

# Allgemeine Konfliktauflösung

- *„Site Priority“*:  
Bei Prioritätsbasierten Konfliktauflösungsverfahren werden nicht alle Replikationsstandardorte als gleichberechtigt erachtet. Dies ist meist der Fall, wenn ein Standard den Anspruch auf „präzisere“ erhebt.
- *„Notice User“*:  
Wird oft als letzte Möglichkeit angesehen, wenn andere Verfahren scheitern wird der Nutzer aufgefordert den entstandenen Konflikt manuell aufzulösen.

# Attributbasierte/Spaltenbasierte Konfliktauflösung

- „*Additive and Average*“: Bei zwei konkurrierenden Werten wird einfach der Durchschnitt der beiden gebildet. Je nach dem, ob man „Additive“ oder „Average“ verwendet wird dabei auf eine andere Formel zurückgegriffen.
- „*Minimum and Maximum*“: Bei zwei konkurrierenden Werten wird einfach das Minimum, bzw. Maximum der beiden gebildet.
- „*Priority Group*“: Kommt bei konkurrierenden Zugriffen auf eine Relation zum Einsatz. Dabei werden die Spalten der Relation mit Prioritäten versehen, die Spalte mit der höchsten Priorität bekommt als erstes den Zugriff.

# Replikation mit Oracle 10g

Oracle 10g kann folgende Datenbankobjekte replizieren:

- Tabellen (oder Partitionen von Tabellen), Views / Object Views
- Indexe und Indextypes
- Trigger
- Packages, Package Bodies sowie Stored Procedures / Functions
- User-Defined Types / Type Bodies und User-Defined Operators
- Synonyms

## Replication Groups

Grundlage für jede Replikation mit Oracle 10g sind sogenannte „Replication Groups“, in diesen werden die zu replizierenden Objekte zugeordnet.

# Multimaster Replikation

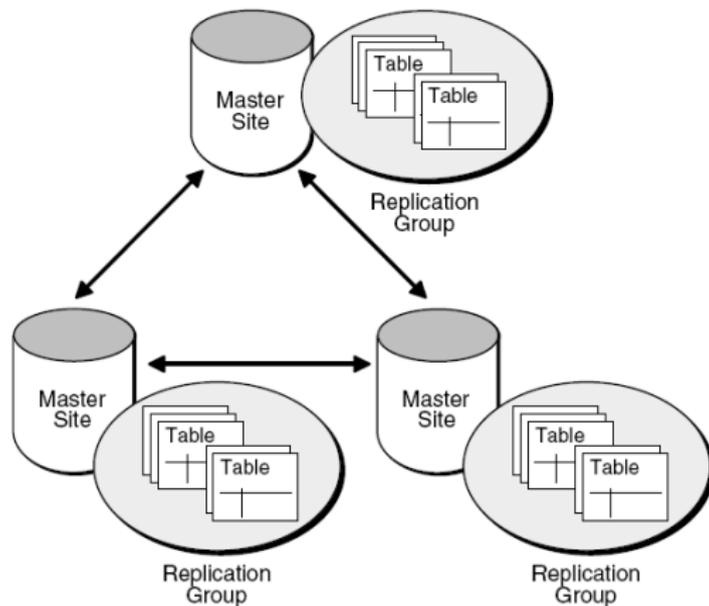


Abbildung: Multimaster Replikation - (Ora07a, Seite 1-5)

# Multimaster Replikation

Definition: (Ora07a, Seite 1-4)

*Multimaster replication (also called peer-to-peer or n-way replication) enables multiple sites, acting as equal peers, to manage groups of replicated database objects. Each site in a multimaster replication environment is a master site, and each site communicates with the other master sites.*

# Multimaster Replikation

## Eigenschaften

- symmetrische Replikation
- standardmässig asynchrone Replikation
- kann optional auch als synchrone „Multimaster“ - Replikation konfiguriert werden.
- prozedurale Replikation kann ebenfalls optional genutzt werden

# Materialized View Replikation

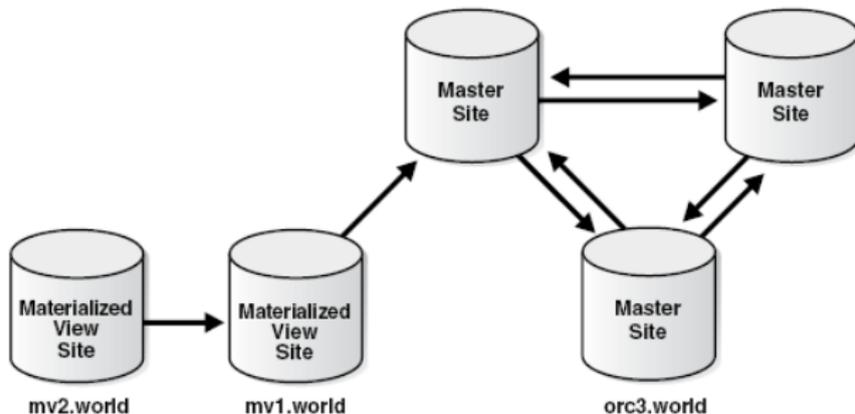


Abbildung: „Multitier Materialized Views“ Replikation - (Ora07a, Seite 3-23)

# Materialized View Replikation

## „Single Master Replikation“

- Die (Read-Only) materialized View Replikation ist technisch sicherlich die einfachste in Oracle 10g zur Verfügung stehende Replikation. Sie wird auch als „Single-Master“-Replikation bezeichnet und kann sowohl synchron, als auch asynchron ablaufen.

## Die materialized View Replikation...

- ...ermöglicht schnellen lokalen Zugriff
- ...reduziert die Last auf dem Master-Server
- ...kann die Sicherheit erhöhen, da nur bestimmte Daten (ein beliebiges Subset) vom Master repliziert werden können.
- ...bietet die Möglichkeit zum erstellen von „Disconnected Materialized View Enviroments“

# Materialized View Replikation

## Refresh Arten

Die materialized-View-Replikation unterstützt folgende Refresh-Arten:

- **Fast refresh** - Nutzt das materialized View Log um nur die Spalten, welche sich seit dem letzten Update verändert haben zu aktualisieren.
- **Complete refresh** - Aktualisiert die komplette materialized View
- **Force refresh** - führt (wenn möglich) ein „Fast refresh“ aus, sollte dies nicht möglich sein wird eine komplette Aktualisierung („Complete Refresh“) durchgeführt.

## „updateable“ materialized Views

Eine interessante Variante stellt die „updateable“ materialized Views, welche dem Nutzer lokal Änderungen an der materialized View erlauben. Diese lokalen Änderungen werden dann an den Master übermittelt und von dort weiter zu allen Knoten der Replication Group.

# „Multimaster und Materialized View Hybrid“ - Replikation

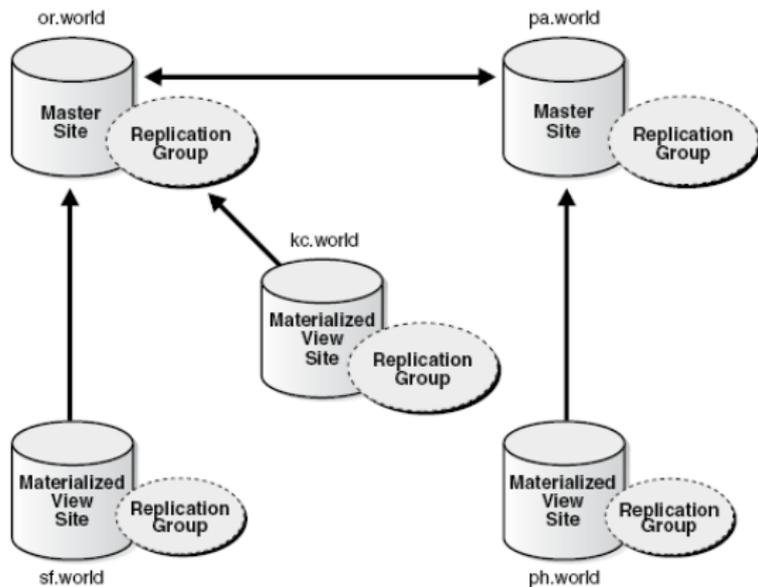


Abbildung: „Multitier Materialized Views“ Replikation - (Ora07a, Seite 3-23)

# „Multimaster und Materialized View Hybrid“ - Replikation

## Hybrid Replikation

- In Oracle 10g gibt es die Möglichkeit Multimaster und Materialized View Replikation zu kombinieren. Dies ist sicherlich ein in der Praxis häufig anzutreffendes Vorgehen, da es die Vorteile beider Replikationsstrategien kombiniert ohne nennenswerte Nachteile zu generieren.

# DBS unabhängige Replikation?

- **Dateisystem Replikation** - Dabei wird die Replikation von dem Datenbanksystem ins Dateisystem verlagert.
- **„Snapshot-Replikation“: Replikation im Datawarehouse Umfeld** - Stellt eine „One-Way“-Replikation dar, meist liefert ein Quellsystem (regelmässig) Daten an ein Datawarehouse, welche dort geladen und somit repliziert werden. Dies kann einerseits durch einen Export/Import-Mechanismus oder durch einen Database-Link und Materialized Views geschehen.

## Literatur

- [KE04] Kemper, Alfons und André Eickler: *Datenbanksysteme - Eine Einführung*, Kapitel Verteilte Datenbanken, Seiten 443–478. Oldenbourg, München, 5., aktualisierte und erweiterte Auflage, 2004.
- [Kud07] Kudraß, Thomas (Herausgeber): *Taschenbuch Datenbanken*, Kapitel Verteilte und förderierte Datenbanksysteme, Seiten 394–426. Hanser, München, 2007.
- [MyS08] MySQL AB: *MySQL 5.1 Referenzhandbuch*, 2008.
- [Ora07a] Oracle: *Oracle® Database Advanced Replication 10g Release 2 (10.2) - Part Number B14226-02*, 2007.
- [Ora07b] Oracle: *Oracle® Database Advanced Replication Management API Reference 10g Release 2 (10.2) - Part Number B14227-02*, 2007.
- [Pet06] Petković, Dušan: *SQL Server 2005 - Eine umfassende Einführung*, Kapitel Datenreplikation, Seiten 485–499. dpunkt.verlag, Heidelberg, 1. Auflage, 2006.
- [Pos08] The PostgreSQL Global Development Group: *PostgreSQL 8.3.1 Documentation*, 2008.
- [SKS02] Silberschatz, Abraham, Henry F. Korth und S. Sudarshan: *Database System Concepts*. McGraw Hill, New York, 4. Auflage, 2002.