

Replikation

Allgemein

Inhalt

- ▶ Definition
- ▶ Pro & Contra
- ▶ Anwendung
- ▶ Synchrone & asynchrone Replikation
- ▶ Unidirektionale & bidirektionale Replikation
- ▶ Klassifikation
- ▶ Resümee

Definition

„Verfahren der Datensicherung bei dem dieselben Daten von einem primären Speichermedium auf ein oder mehrere sekundäre Speichermedien kopiert werden.“ – itwissen.info

► Backup + Synchronisation

Unterschied zu Caching

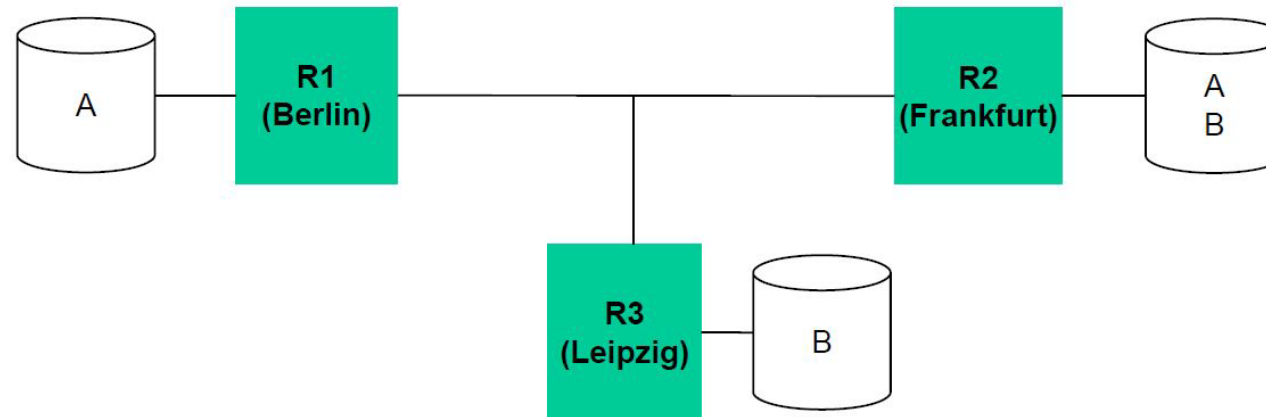
- ▶ meist dauerhaft
- ▶ statisch ausgewählt
- ▶ administrativer Aufwand

Vorteile

- ▶ Skalierbarkeit
- ▶ Verfügbarkeit
- ▶ Performance
- ▶ Disconnected Computing

Nachteile

- ▶ Aufwand
- ▶ Speicherbedarf
- ▶ Komplexität



Anwendung

Erhaltung der Datenkonsistenz

- Kopien wechselseitig konsistent zu halten: *1-Kopien-Äquivalenz*
- kleine Kopienzahl

**Zielkonflikte
der
Replikationskontrolle**



Erhöhung der Verfügbarkeit, effizienter Lesezugriff

- große Kopienzahl
- Zugriff auf beliebige und möglichst wenige Kopien

Minimierung des Änderungsaufwands

- kleine Kopienzahl
- möglichst wenige Kopien synchron aktualisieren

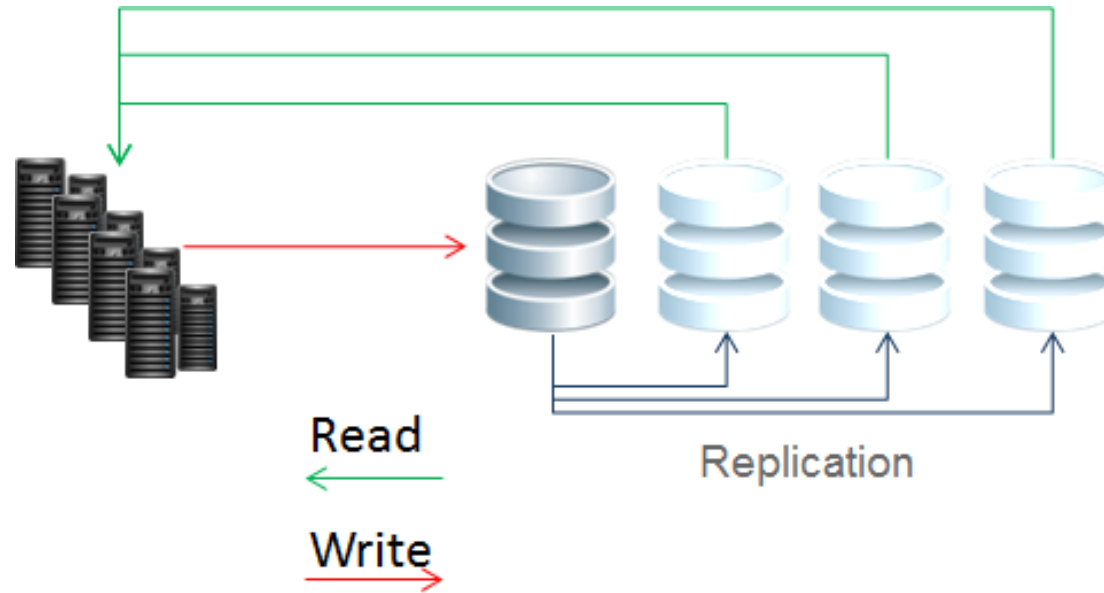
Anwendung: Mobile Computing

- ▶ z.B.: Außendienstmitarbeiter
- ▶ offline arbeiten
- ▶ Teilreplikation
- ▶ tägliche Synchronisierung
- ▶ Ziel: Verfügbarkeit



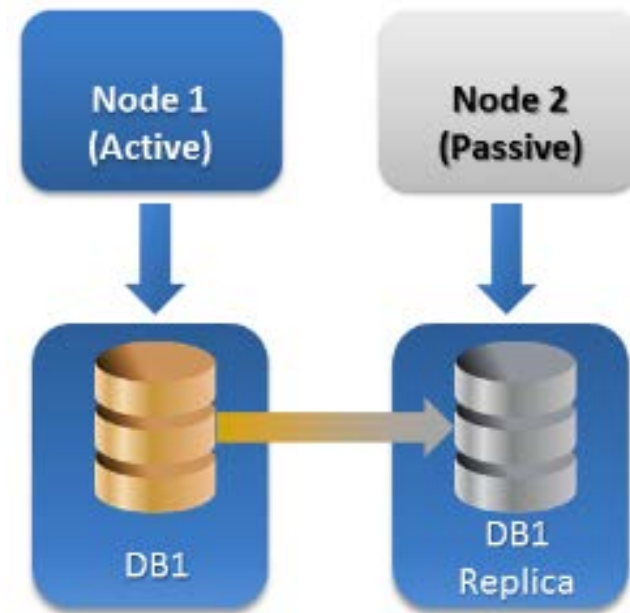
Anwendung: Skalierbarkeit von Leselast

- ca. 97 % Leseanfragen
- Master/Slave
- Ziel: Lastverteilung



Anwendung: Hochverfügbarkeit

- ▶ Master/Slave
- ▶ wenige Kopien
- ▶ Ziel: Verfügbarkeit



Synchrone Replikation

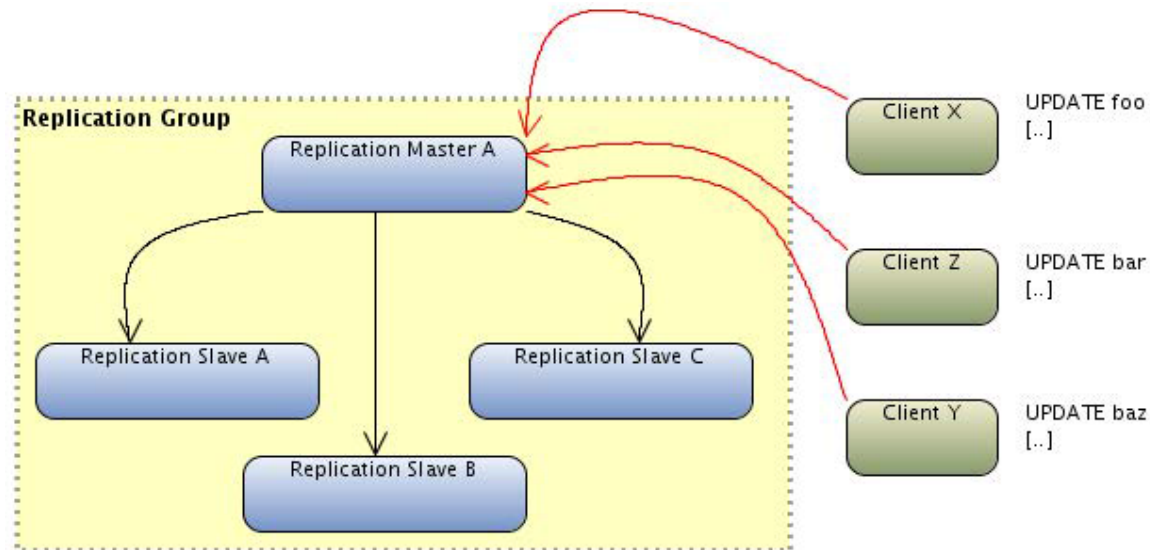
- ▶ Pro:
 - ▶ transaktionale Konsistenz
 - ▶ keine Konflikte
- ▶ Contra:
 - ▶ Verhalten bei Teilausfällen
 - ▶ Sperren
 - ▶ Performance
- ▶ Einsatz: Skalierung von Leselast

Asynchrone Replikation

- ▶ Pro:
 - ▶ Schreibperformance
 - ▶ Verfügbarkeit
- ▶ Contra:
 - ▶ Konvergenz
 - ▶ Konflikte
- ▶ Einsatz: Mobile Computing & Hochverfügbarkeit

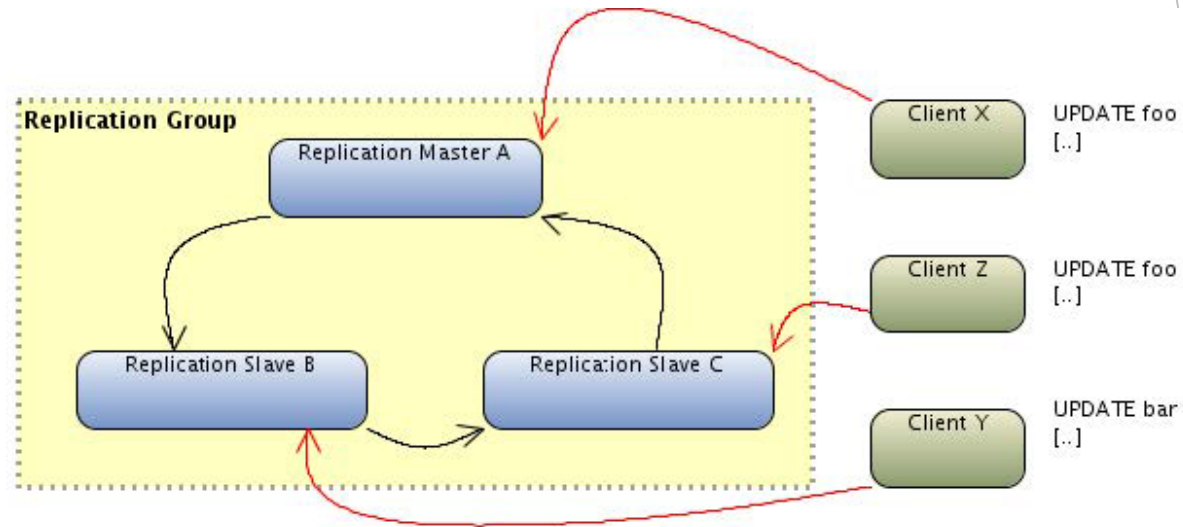
Unidirektionale Replikation

- Master/Multi-Slave
- 1 ändernde Instanz
- konfliktfrei
- keine Skalierung der Schreiblast
- Einsatz: Skalierung von Leselast & Hochverfügbarkeit



Bidirektionale Replikation

- ▶ Multi-Master
- ▶ gleiche Rechte
- ▶ Skalieren von Schreibzugriffen
- ▶ gesteigener Replikationsaufwand
- ▶ Einsatz: Mobile Computing



Klassifikation

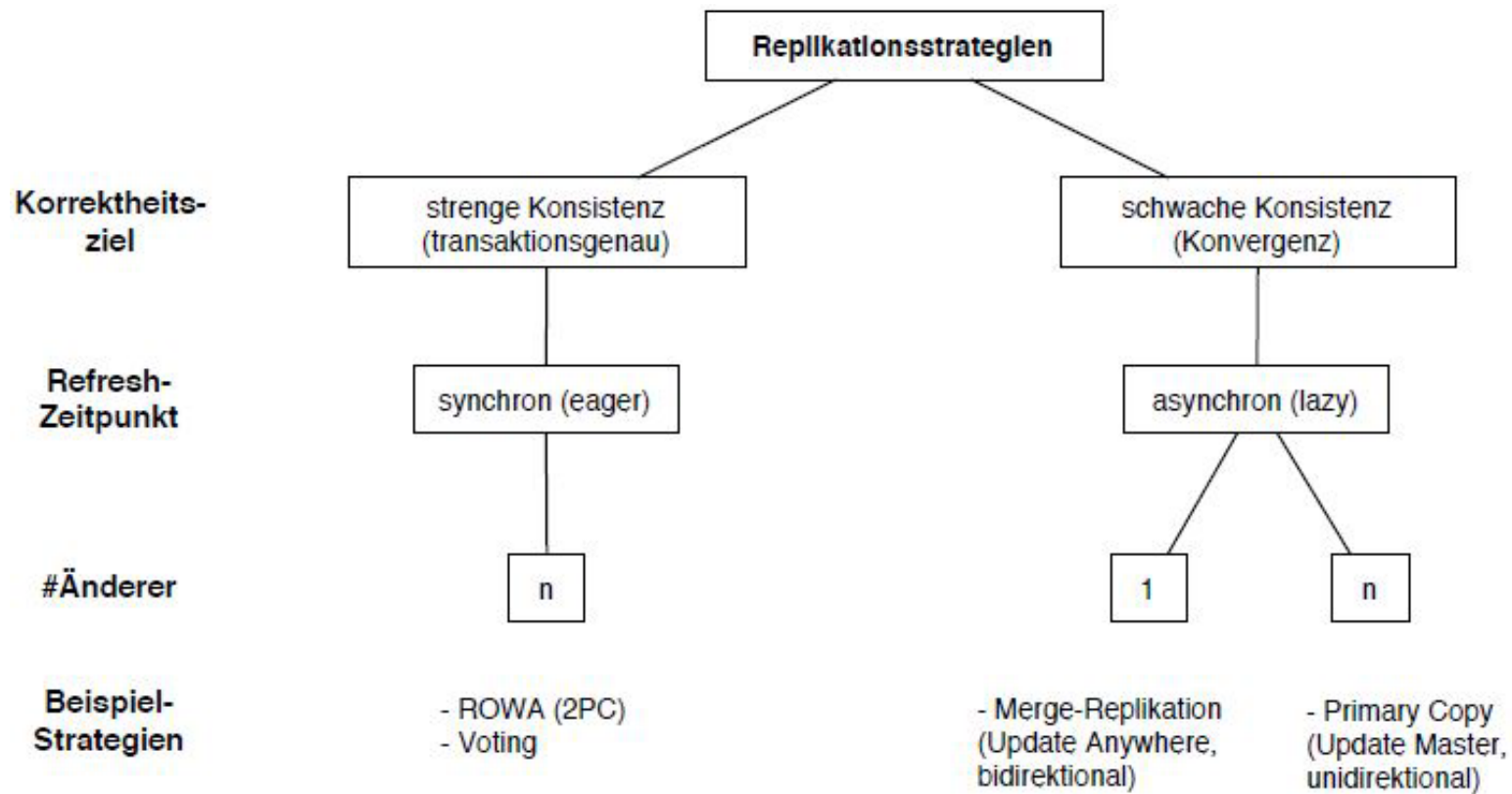
► Synchronisierung

► Wann?

► In welche Richtung?

Propagation vs. Ownership	Lazy	Eager
Group	n transactions n object owners	one transaction n object owners
Master	n transactions one object owner	one transaction one object owner

Klassifikation



Resümee

- ▶ absichtliche Redundanzen + Synchronisierung
- ▶ Performance, Verfügbarkeit, Skalierbarkeit
- ▶ Verwaltungsaufwand, Konflikte
- ▶ Synchronisierungszeitpunkt & -richtung