

Oracle-Datenbankmigration mit minimalen Ausfallzeiten



Agenda

- ▶ **Allgemeines**
- ▶ **Abgrenzung**
- ▶ **Herkömmliche Migrationswege**
- ▶ **Zeiteffiziente Migrationswege**

Agenda

- ▶ **Allgemeines**
- ▶ Abgrenzung
- ▶ Herkömmliche Migrationswege
- ▶ Zeiteffiziente Migrationswege

Migrationsdefinition

„Unter **Migration** (von lateinisch *migratio* für „Wanderung“ oder „Übersiedlung“) werden im Rahmen der Informationstechnik verschiedenartige Umstellungsprozesse in Datenverarbeitungssystemen verstanden. Gelegentlich werden Migration und Portierung (von lateinisch *portatio* für „Herbeischaffen“ oder „Transport“) synonym verwendet.“ (Quelle: Wikipedia)



Migrationsüberlegung

- ▶ Warum überhaupt migrieren?
 - Hardware, Software, Standort
- ▶ Welche SLAs müssen eingehalten werden?
 - Nutzerzugriffe, Schnittstellen
- ▶ Fallback-Szenarien?
 - Fallback-Strategien!

Agenda

- ▶ Allgemeines
- ▶ **Abgrenzung**
- ▶ Herkömmliche Migrationswege
- ▶ Zeiteffiziente Migrationswege

Abgrenzung

- ▶ Eine Datenbankmigration ist **nicht**:
 - Patching innerhalb eines Releases
 - Physikalisches Upgrade
 - Ändern der Anwendung

Abgrenzung

▶ Stattdessen

- „Bewegung der Daten in ein neues Skelett“
- Dies kann auf Ebene der Datenbank, der Software sowie der Hardware/Plattform oder gar zu einem anderen Zeichensatz stattfinden

Agenda

- ▶ Allgemeines
- ▶ Abgrenzung
- ▶ **Herkömmliche Migrationswege**
- ▶ Zeiteffiziente Migrationswege

Datenbankmigration

- ▶ Typischer Ablauf einer „herkömmlichen“ Migration
 1. Herabsetzen der Nutzungsberechtigung der DB
 2. Bereitstellen der Daten mittels exp/Datapump/TS
 3. Transport der Daten zur neuen Umgebung
 4. Aufbau der neuen Umgebung als neue Datenbank
 5. Import der Quelldaten
 6. Aktivierung des Betriebs

- ▶ Zeitaufwand: min. 1 AT

Datenbankmigration

▶ Vorteile

- Sichere Methode, tausendfach erprobt
- Geringe Implementierungsaufwände

▶ Nachteile

- Downtime!
- Verletzung von SLAs

Das geht besser!

Agenda

- ▶ Allgemeines
- ▶ Abgrenzung
- ▶ Herkömmliche Migrationswege
- ▶ **Zeiteffiziente Migrationswege**

Vorgaben

- ▶ Migrationen dauern **immer** zu lang
 - Die Nutzer/Kunden/Fachbereiche wollen möglichst ZERO-Downtime
 - Oftmals ohne „echten“ Mehrwert für die Betroffenen

- ▶ Migrationen mit wenig Verzögerungen
 - kosten Geld
 - oder Zeit

Minimal Downtime Migrationen

▶ Zeiteffiziente Migrationswege

- Dataguard
- Replikation
- Incremental Backups
- Manuelle Migration

Minimal Downtime Migrationen

▶ Zeiteffiziente Migrationswege

- **Dataguard**
- Replikation
- Incremental Backups
- Manuelle Migration

Dataguard

- ▶ Dataguard ist prädestiniert für minimal Downtime bei
 - Hardwarewechsel
 - Standortwechsel
- ▶ Nicht für Release-Wechsel geeignet!
- ▶ Enterprise Edition!

Dataguard

- ▶ Aufwand ist relativ gering
 - Duplizieren der Datenbank
 - Anlegen von Standby-Controlfile + Logs
 - Aktivieren des Redo Transports
 - Switchover mit DGMGRL oder EM

Minimal Downtime Migrationen

▶ Zeiteffiziente Migrationswege

- Dataguard
- **Replikation**
- Incremental Backups
- Manuelle Migration

Replikationsprodukte

- ▶ Oracle: GoldenGate

- sehr teuer

ORACLE®

- ▶ Oracle: Streams

Oracle Streams is deprecated in Oracle Database 12c Release 1 (12.1). Use Oracle GoldenGate to replace all replication features of Oracle Streams.

ORACLE®

- ▶ DBVisit: Replicate

- Preiswert
- Leistungsfähig



Minimal-Downtime Migration?

- ▶ Vorgehensweise bei der Migration
 - Vorbereitende Maßnahmen (Analyse des Datenmodells)
 - Aufbau einer **konsistenten** Datenkopie als Basis
 - Implementierung der Replikation
 - Deaktivierung von Services in Quelldatenbank
 - Umkehrung der Replikation
 - Aktivierung von Services in Zieldatenbank

Vor- und Nachteile

▶ Vorteile

- Alle möglichen Szenarien können abgedeckt werden
Releasewechsel, Plattformwechsel, Hardwarewechsel etc.
- Services können einzeln migriert werden
- Datenreorganisation möglich

▶ Nachteile

- Kosten- und Zeitaufwand
- Mögliche Einschränkungen bei Datentypen

Minimal Downtime Migrationen

▶ Zeiteffiziente Migrationswege

- Dataguard
- Replikation
- **Incremental Backups**
- Manuelle Migration

RMAN Incremental Backups

- ▶ Sehr gut geeignet bei
 - Wechsel von Standort
 - Wechsel von Hardware
 - Physikalischer Umstrukturierung
 - NOLOGGING Operationen in DB
 - **Standard Edition!**

Vorgehensweise

▶ Einfache Herangehensweise

- RMAN Incremental Level 0 Backup auf Quelldatenbank
- RMAN Restore auf Zielsystem
- RMAN Incremental Level 1 Backup auf Quelldatenbank
- RMAN Recover auf Zielsystem
- Shutdown auf Quellplattform
- Startup auf Zielplattform



Minimal Downtime Migrationen

▶ Zeiteffiziente Migrationswege

- Dataguard
- Replikation
- Incremental Backups
- **Manuelle Migration**

Manuelle Migration

- ▶ Wird notwendig bei
 - Kostendruck
 - Plattform/Releasewechsel

- ▶ Erfordert
 - Einfaches Datenmodell
 - Implementierungsaufwand
 - möglicherweise Zugriff auf interne Dokumentation

Implementierungsbeispiel

- ▶ Datapump
- ▶ PL/SQL
- ▶ Differentielle Übertragung von Änderungen
(*manuelle Replikation*)
- ▶ Intensive Erforschung des Datenmodells
- ▶ Gut geeignet für Datenmodelle mit vielen Stammdaten und wenigen Bewegungsdaten



Referent

Sven Salzwedel

Senior Systemberater Support

T: +49 351 25859-2819

F: +49 351 25859-3699

sven.salzwedel@robotron.de