



DOAG Regionaltreffen München

Oracle Datenbank-Migration

Andrew Lacy
Senior Consultant
OPITZ CONSULTING München GmbH



Specialized
Oracle Database

DOAG, München, 16. Mai 2011

Agenda

- 1. Migrationsvarianten und Begriffe**
- 2. Migration durch Import in neue Datenbank**
- 3. Hardware-Migration mit Übernahme der Datenbank**
- 4. Migration/Upgrade mit Transportable Tablespace**
- 5. Hardware-Migration mit Data Guard**
- 6. Datenbank-Upgrade mit "Rolling Migration"**
- 7. Fazit**

1

Migrationsvarianten und Begriffe

Migrationsvarianten, Begriffe

■ Datenbank-Upgrade

- Austausch der Software (ORACLE_HOME) durch neuere Version
- Anpassen DataDictionary (catupgrd.sql)
- Dauer abhängig von Hardware und Anzahl Datenbank-Objekten

■ Hardware-Migration / Plattform-Migration

- Transfer der Datenbank auf einen anderen Server
- Gleiche Architektur, gleiche Betriebssystem-Familie (z.B. SuSE → RedHat)
- Gleiche Architektur, anderes Betriebssystem (z.B. Windows → Linux)
- Andere Architektur, anderes Betriebssystem (z.B. AIX → Linux)
- 32 / 64 - bit

■ Kombination von DB-Upgrade mit Hardware-Migration

■ Little-/Big-Endian

- Angabe, wo im Speicher das höchstwertige Byte steht
- Bsp: „Motorola-Format“ → Big-Endian, „Intel-Format“ → Little-Endian

■ Downtime

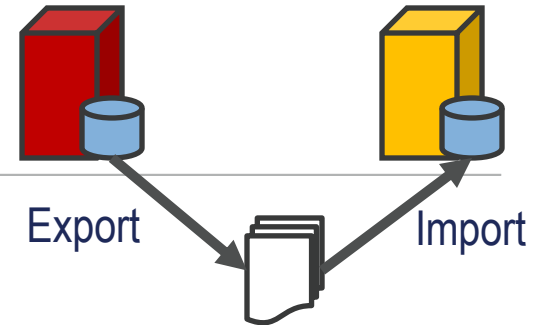
- Zeit, während dem kein Client-Zugriff möglich ist. Wie minimieren?

2

Migration durch Import in neue Datenbank

Import in neue Datenbank

■ Neue Datenbank aufbauen und Daten importieren mit Oracle-Mitteln



■ Oracle **exp / imp** Utility

- 😊 In allen Oracle Versionen verfügbar
- 😊 Plattform-unabhängig
- 😞 Langsam (kein direct-path-load, keine Parallelisierung)
- 😞 Keine Erzeugung einzelner User (nur bei Full-Import)

■ Oracle **Datapump expdp / impdp** Utility

- 😊 Gleiche Vorteile wie exp/imp aber Schneller als exp/imp (direct-path-load, Parallelisierung)
- 😞 Erst ab Oracle 10g verfügbar
- 😞 Nicht alle Datentypen korrekt unterstützt (insb. bei Datapump über NETWORK_LINK)

■ Aufgrund Datentyp-Problemen ggf. Kombination beider Tools

- 😞 Problematisch: LONGs, Object-/XMLType.Columns, PARTITIONS, ... → prüfen!
- 😞 Generell: keine (eigenen) Objekte unter SYS! (Public synonyms)

Import in neue Datenbank

- **Neue Datenbank aufbauen und Daten importieren mit Oracle Mitteln und eigenen Scripts**
 - sqlplus und sqlloader
 - Exportieren als Flat-file mit sqlplus, importieren mit sqlloader
 - ☺ Direct-Path Import
 - ☺ parallelisierbar
 - ☹ Nur Datenübernahme, Scripts für Erstellung und Transfer selbst zu implementieren
 - Database-Link
 - create table as select ...
 - Etwa gleiche Vor- und Nachteile wie bei sqlplus/sqlldr

Import in neue Datenbank

■ Neue Datenbank aufbauen und Daten importieren mit Zusatzsoftware

■ Eventuell applikatorischer Datentransfer

■ Oracle GoldenGate (Von Oracle gekaufte Replikationslösung)

😊 Nicht auf Oracle-Datenbanken beschränkt

☹️ Signifikante Kosten

■ Quest Shareplex

😊 Simples Handling für Migrationszwecke

☹️ Kosten & Beschränkung auf Oracle

■ Generelle Vor- und Nachteile

■ Vorteile:

😊 Gleichzeitige Migration auf neuere Datenbank möglich

😊 Neu aufgebaute, reorganisierte Datenbank, Nutzung aller neuen Features möglich

😊 Plattform-unabhängig

■ Nachteile:

☹️ Langsam. Insbesondere der Aufbau von Indexen kann sehr lange dauern

☹️ User, Rollen, Objekte müssen neu erstellt werden

3

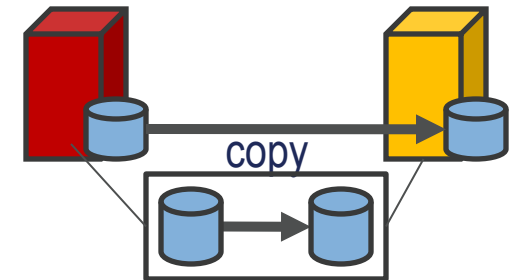
Hardware-Migration mit Übernahme der Datenbank

Hardware-Migration mit Übernahme der DB

■ Kopieren der bestehenden Datenbank

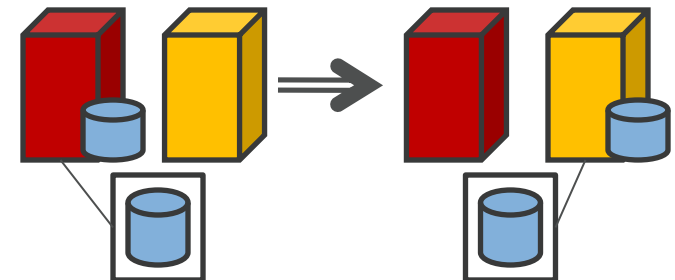
- Via Netzwerk (NFS, SMB, scp, ftp, ...) oder SAN

😊 Vorteil: bei Misserfolg steht alte Umgebung noch zur Verfügung



■ Verschieben des bestehenden Storage auf neuen Server

- Physischer Transfer der Disks, oder
- umhängen der LUN's im SAN



■ Vor- und Nachteile

- 😊 Technisch einfach
- 😊 Risikoarm, da keine Änderung an der Datenbank
- ☹️ Downtime für die Zeit des Kopierens
- ☹️ Nur innerhalb der gleichen Plattform
- ☹️ Kein Upgrade der Oracle-Version
- ☹️ Keine Reorganisation der Daten/Indizes

Hardware-Migration mit Übernahme der DB

■ Migration mit Transportable Database

■ Erfolgt über RMAN

- 😊 Gesamter Inhalt, inkl. User, Prozeduren usw. wird übernommen
- 😊 Gewisse Plattform-Migrationen sind möglich
- 😞 Nur für Plattformen desselben Endian-Formats
- 😞 Erst ab Oracle 10g
- 😞 Downtime für den Transport ist größer als mittels DataGuard

4

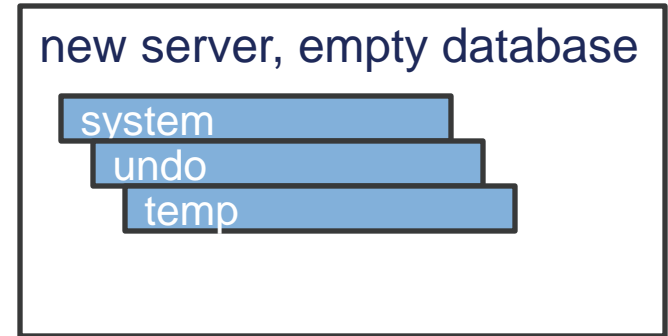
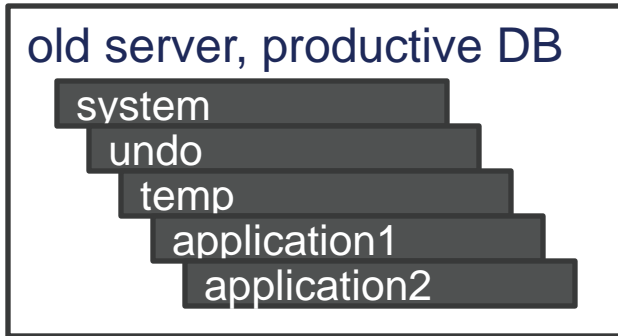
Hardware- / Plattform-Migration mit Transportable Tablespace

Hardware- / Plattform-Migration mit Transportable Tablespaces

- **Neue Datenbank aufbauen und Tablespaces übernehmen**
 - Applikations-Tablespaces können an neue Datenbank "angehängt" werden
 - Tabellen mit Daten werden übernommen
 - aber keine User, Prozeduren usw. → z.B. mit impdp übernehmen
 - Verwendung der Funktionalität "**Transportable Tablespaces**"
 - Vorteile
 - ☺ funktioniert auch über Plattformgrenzen little-/big-Endian hinweg (RMAN – erst ab 10g)
`convert datafile '/oradata/aix.dbf' from platform 'AIX-Based Systems (64-bit)'`
`format '/oradata/linux.dbf';`
 - ☺ Datenbank-Upgrade möglich
 - ☺ Fallback auf alten Server möglich
 - Nachteile
 - ☹ Nicht möglich für SYSTEM Tablespace und Objekte die SYS gehören
 - ☹ User, Prozeduren (deren Information in SYS-Tabellen liegen) müssen zuvor erstellt werden

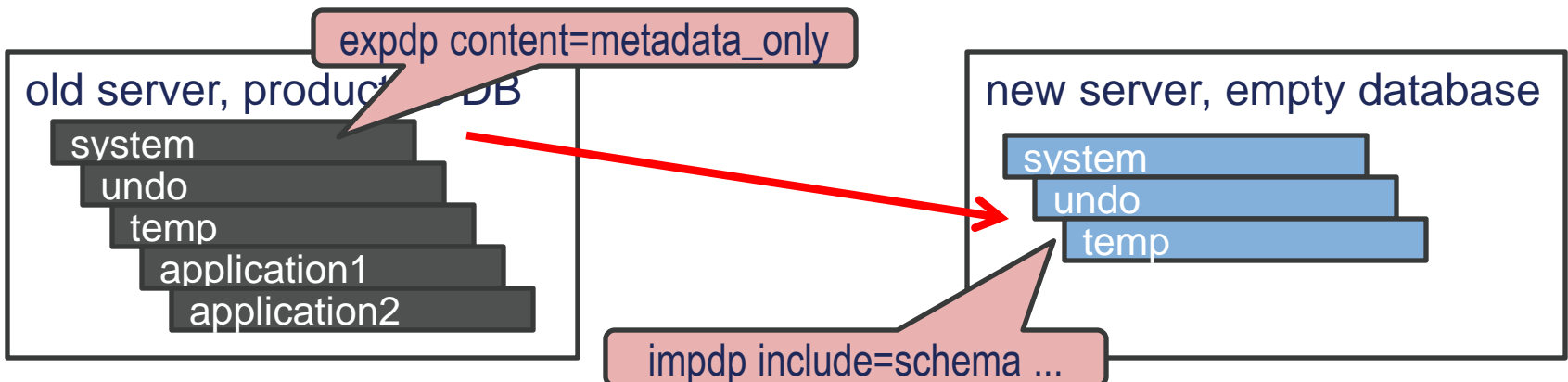
Migration/Upgrade mit Transportable Tablespace

■ Vorbereiten eines neuen Servers mit neuer Datenbank



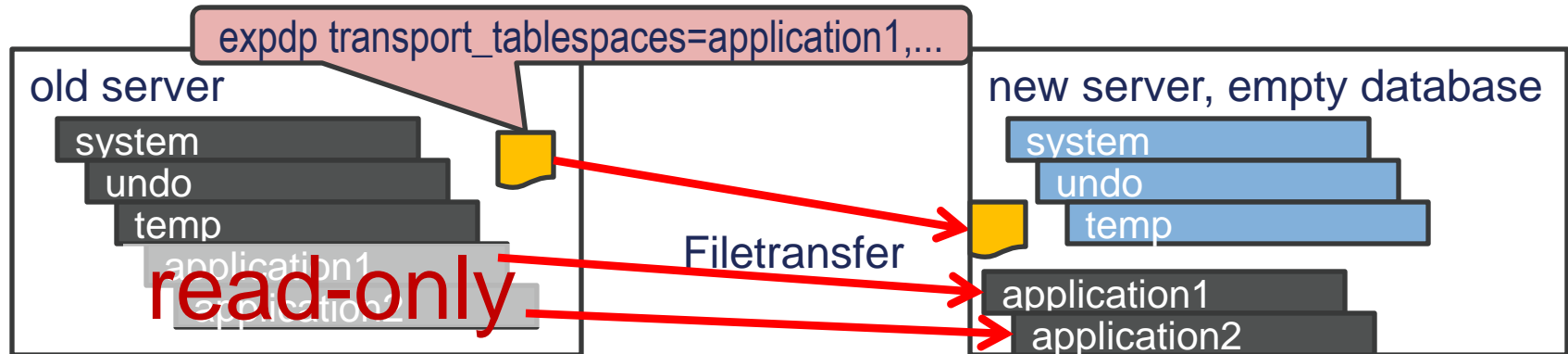
■ Erstellen User, Packages, Grants usw. auf neuem Server

- ohne Tabelleninhalt



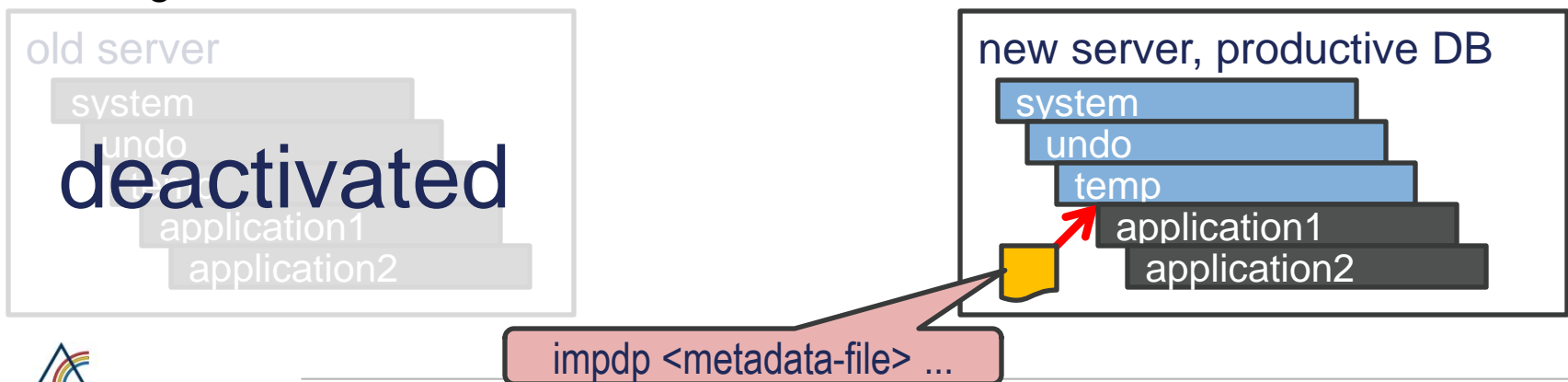
Migration/Upgrade mit Transportable Tablespace

■ Transferieren der Tablespaces (mit Metadaten)



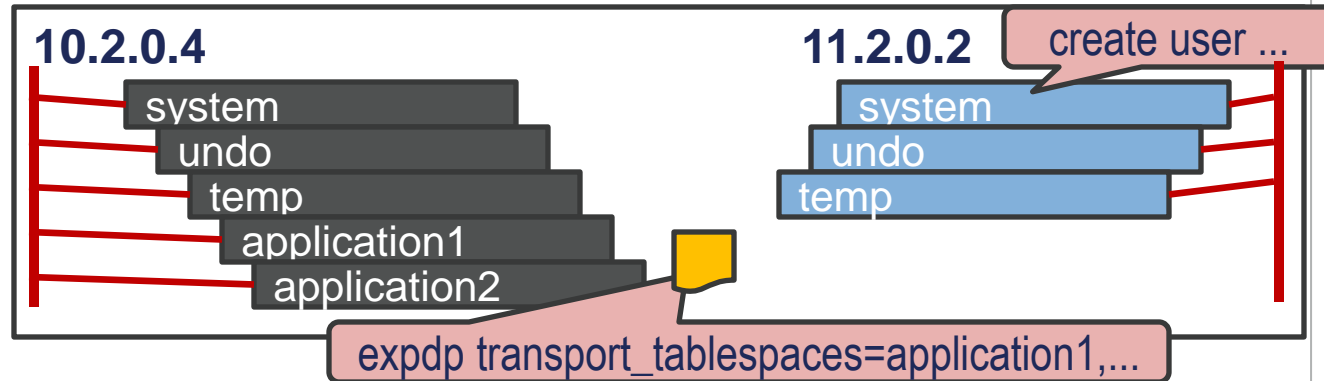
■ Importieren der transportierten Tablespaces

- Gegebenenfalls vorher rman convert datafile



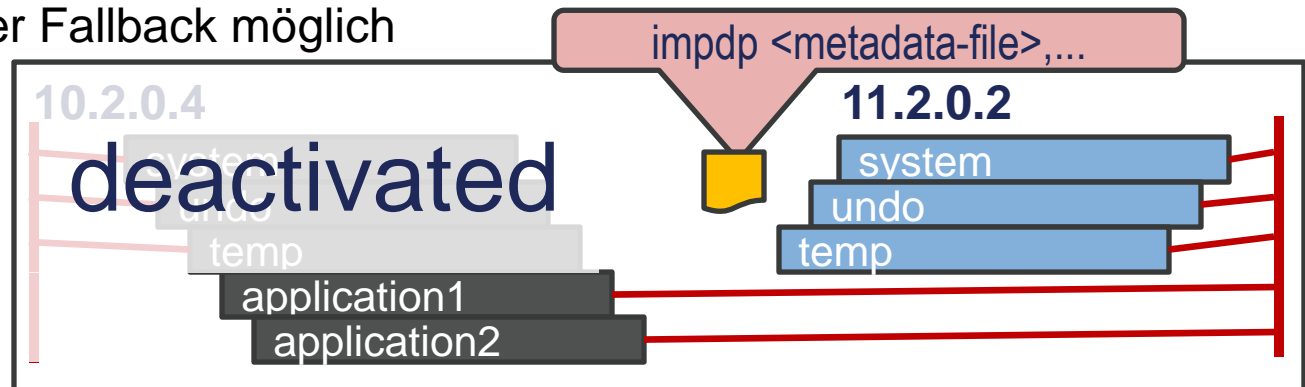
Upgrade auf demselben Server

■ Vorbereiten neue Datenbank und Metadaten



■ Importieren der Transportable Tablespaces

- Kein Umkopieren notwendig
- Aber: kein direkter Fallback möglich

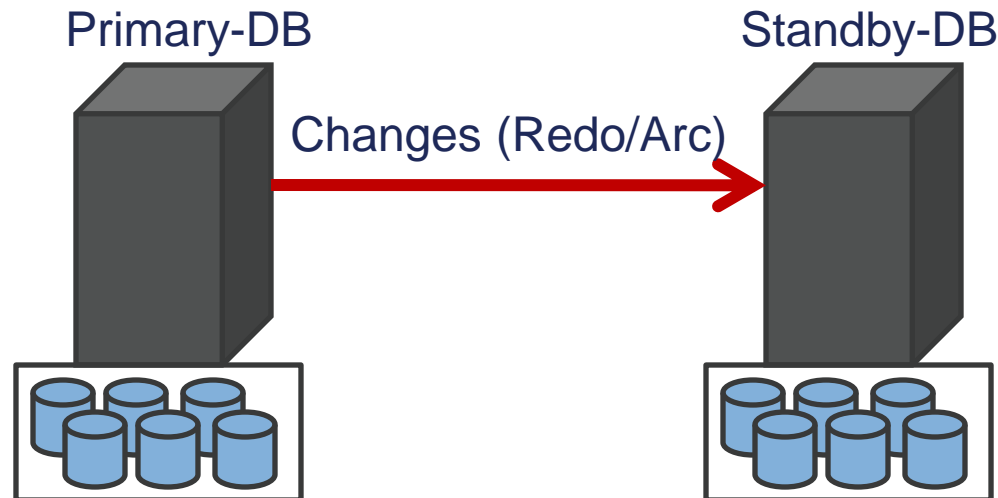


5

Hardware-Migration mit Data Guard

Data Guard

■ Was ist eine Standby Database (Data Guard) ?



- Eine Standby-Datenbank (Standby) ist eine Kopie einer Primärdatenbank, (Primary) die fortlaufend mit den Änderungen der Primary nachgeführt wird (synchron oder asynchron, mit oder ohne toleriertem Datenverlust)
- Wenn die Primary ausfällt, kann die Standby "aktiviert" werden, d.h. sie übernimmt bei einem Failover die Rolle der Primary
- Zu Wartungszwecken kann die Rolle auch getauscht werden ("switchover")

Unterstützte Data Guard (DG) Konfigurationen

■ Primary / Standby auf Servern derselben **Oracle Plattform**

```
SQL> select platform_id, platform_name from v$database;
```

```
PLATFORM_ID PLATFORM_NAME
```

```
-----
```

```
10 Linux IA (32-bit)
```

(11.2.0.1.0)

- ↓ Microsoft Windows (32-bit)
- ↓ Microsoft Windows (x64)
- ↓ Linux x86
- ↓ Linux x86-64
- ↓ Solaris (SPARC) (64-bit)
- ↓ Solaris (x86-64)
- ↓ HP-UX Itanium
- ↓ HP-UX PA-RISC (64-bit)
- ↓ AIX (PPC64)

■ Erlaubt sind:

(MOS Note 413484.1)

- Unterschiedlicher Hersteller (z.B. HP / IBM)
- Unterschiedliche CPU, unterschiedliche Speichergröße
- Unterschiedliche OS-Distribution oder Version (RedHat/SuSE, OEL4/OEL5)

■ Innerhalb gewisser **Betriebssystem-Familien**

- z.B. Linux **x86** / Linux **x86_64**, Windows **32bit** / **64bit** (414043.1)

■ Zwischen verschiedenen Plattformen **gleicher Endianness**

- Nur einige Kombinationen zertifiziert. z.B. (413484.1)
 - Windows 32bit ↔ Linux 32bit
 - HP-UX PA-RISC ↔ HP-UX Itanium
 - Solaris Sparc ↔ AIX Power

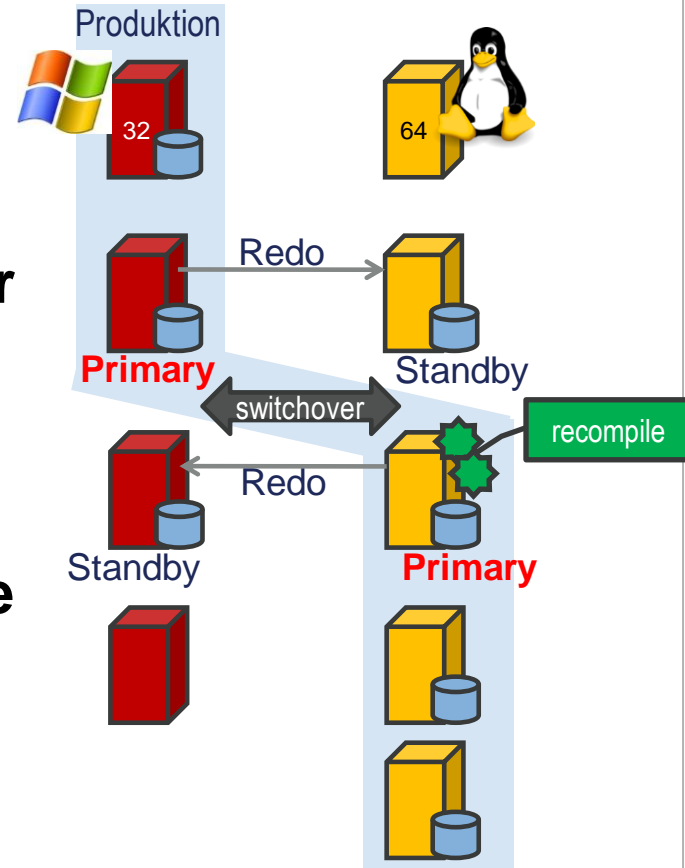
Einige Data Guard Plattform Kombinationen (Physical Standby)

Standby Primary	Mswin 32bit (id=7)	Mswin 64bit (id=12)	Linux 32bit (id=10)	Linux 64bit (id=13)	HPUX PA-RISC (ID=3)	HPUX Itanium (ID=4)	Solaris x86-64 (ID=20)	Solaris Sparc (ID=2)	AIX Power (ID=6)
Mswin32		10g	11g	11g					
Mswin64	10g			11g					
Linux32	11g			10g					
Linux64	11g	11g	10g				11g		
HPUX pa-risc						10g			
HPUX Itanium					10g				
Solaris x86-64				11g					
Solaris Sparc									11.2.0.2
IBM AIX								11.2.0.2	

 Kombination 32/64Bit, Neu-Kompilierung von Packages erforderlich

Migration von 32bit Windows auf 64bit Linux

- Aufbau eines **neuen Linux Servers**
 - Aufbau einer Data Guard Umgebung zwischen dem **alten** und **neuen Server**
 - Dies erfolgt ohne Downtime der Primary
 - **Switchover auf neuen Server**
 - Kurzer Unterbruch im Minutenbereich
 - **Rekompilieren der Datenbank-Objekte**
 - Abhängig von Anzahl/Performance CPU und Anzahl Packages
 - Entfernen des **alten Windows Servers**
-
- **Fallback-Szenario: Switchover auf alte Umgebung zurück**



6

Datenbank-Upgrade mit "Rolling Migration"

Rolling Upgrade

■ Prinzip

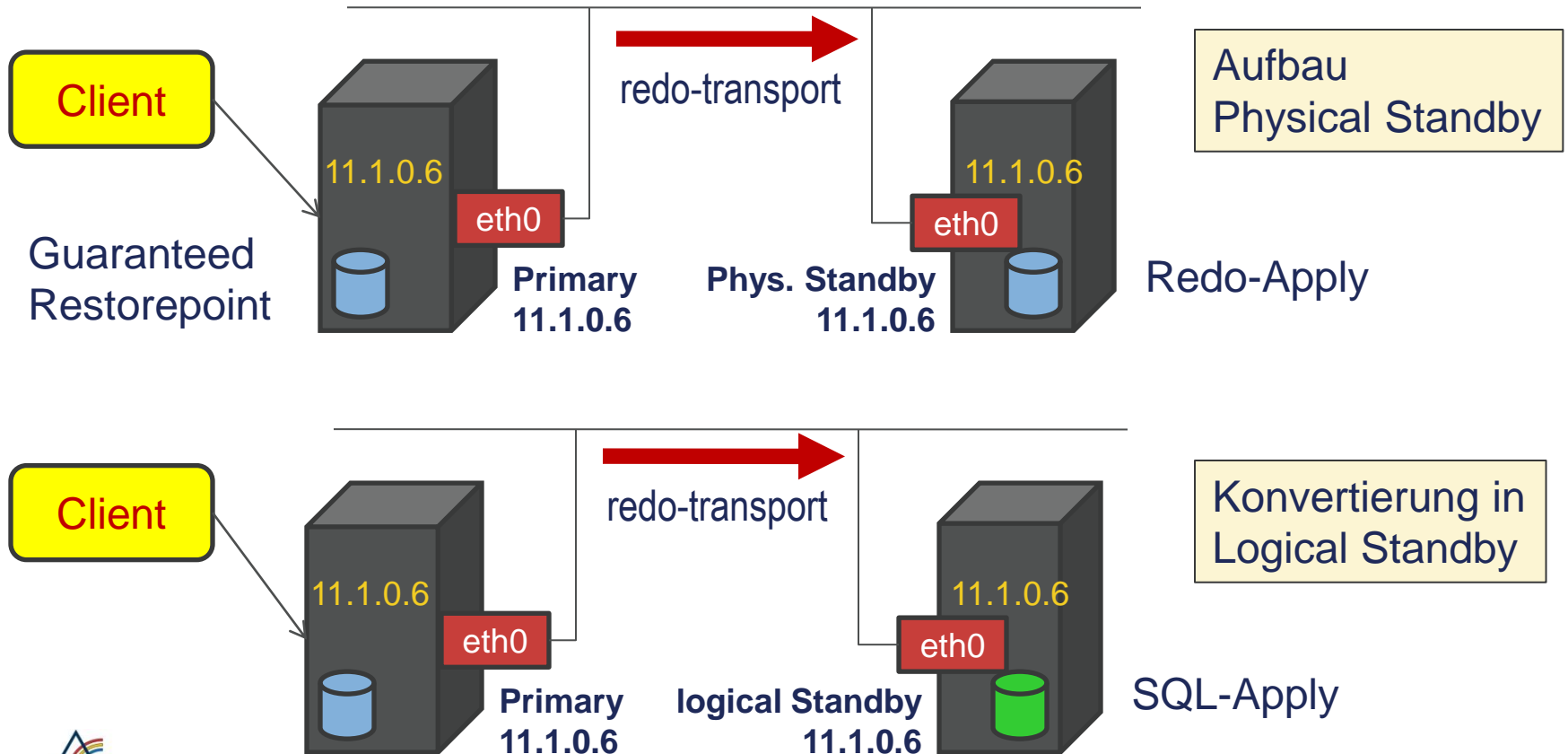
- Primary-Datenbank läuft wie gewohnt weiter
- Standby-Datenbank wird in eine Logische Standby-DB (LSB) konvertiert
 - LSB ist schreibbar; nicht Block-Änderungen, sondern SQL-Statements werden appliziert
 - ☹️ LSB hat viele Einschränkungen: nicht alle Datentypen sind unterstützt
→ verhindert möglicherweise den Einsatz
- Log-Applizierung wird gestoppt
- LSB wird wie eine "normale" Datenbank migriert (Primary läuft weiter)
- Nach der Migration werden Änderungen der Primary auf LSB appliziert (sql)
- Switchover auf die LSB → Migration auf neue Version

- Hinweis 11g: Rolling Migration mit Physical Standby („Transient LSB“)
 - Erzeugt im Hintergrund trotzdem eine LSB, nur die DBID ändert sich dabei nicht
`alter database recover to logical standby keep identity;`
 - 😊 Nach Flashback und CONVERT TO PHYSICAL STANDBY: REDO Apply auf alte Produktion
 - ☹️ Sämtliche Limitierungen der LSB existieren weiterhin (nicht supportete Datentypen usw.)

Rolling Migration

■ Erstellung Logical Standby

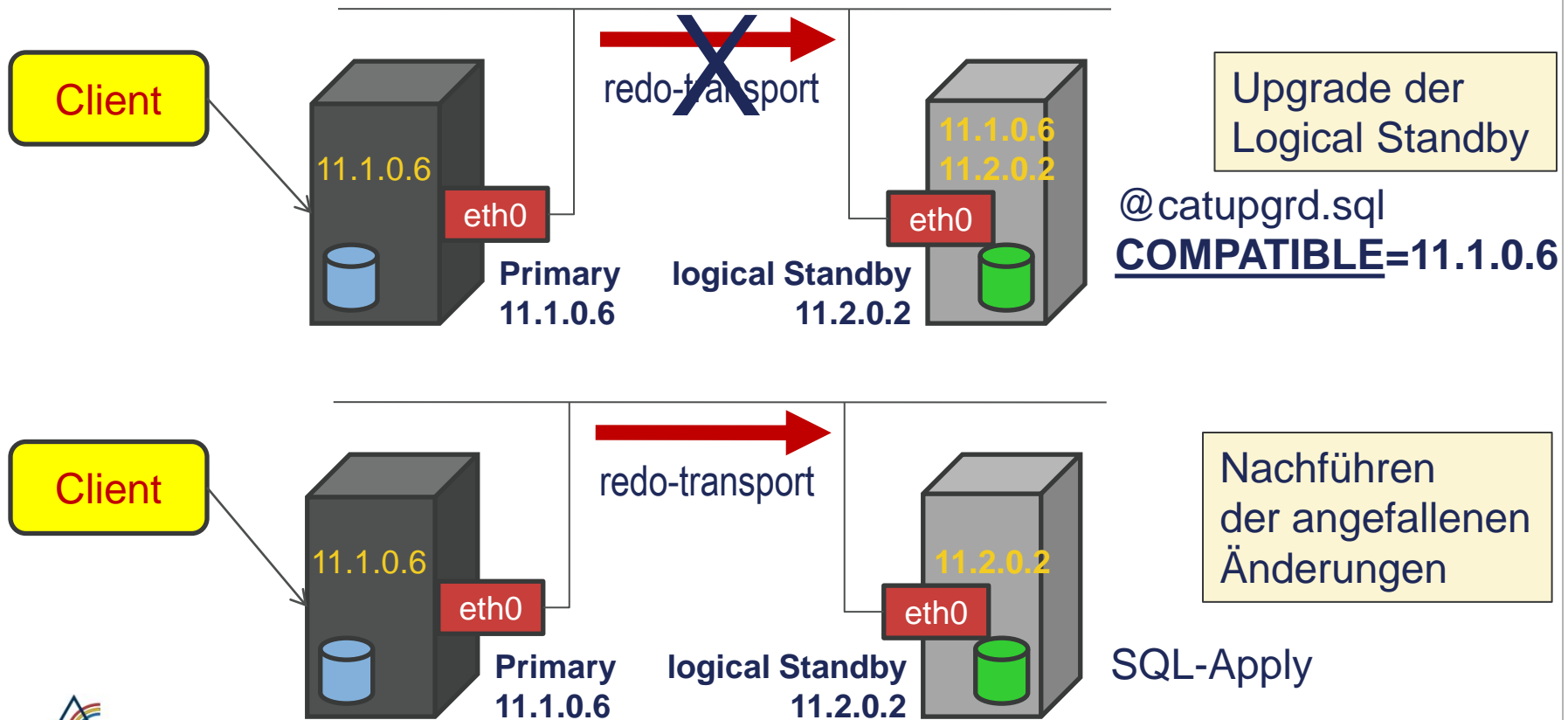
- Physical- zu Logical-Standby konvertieren



Rolling Migration

■ Migration

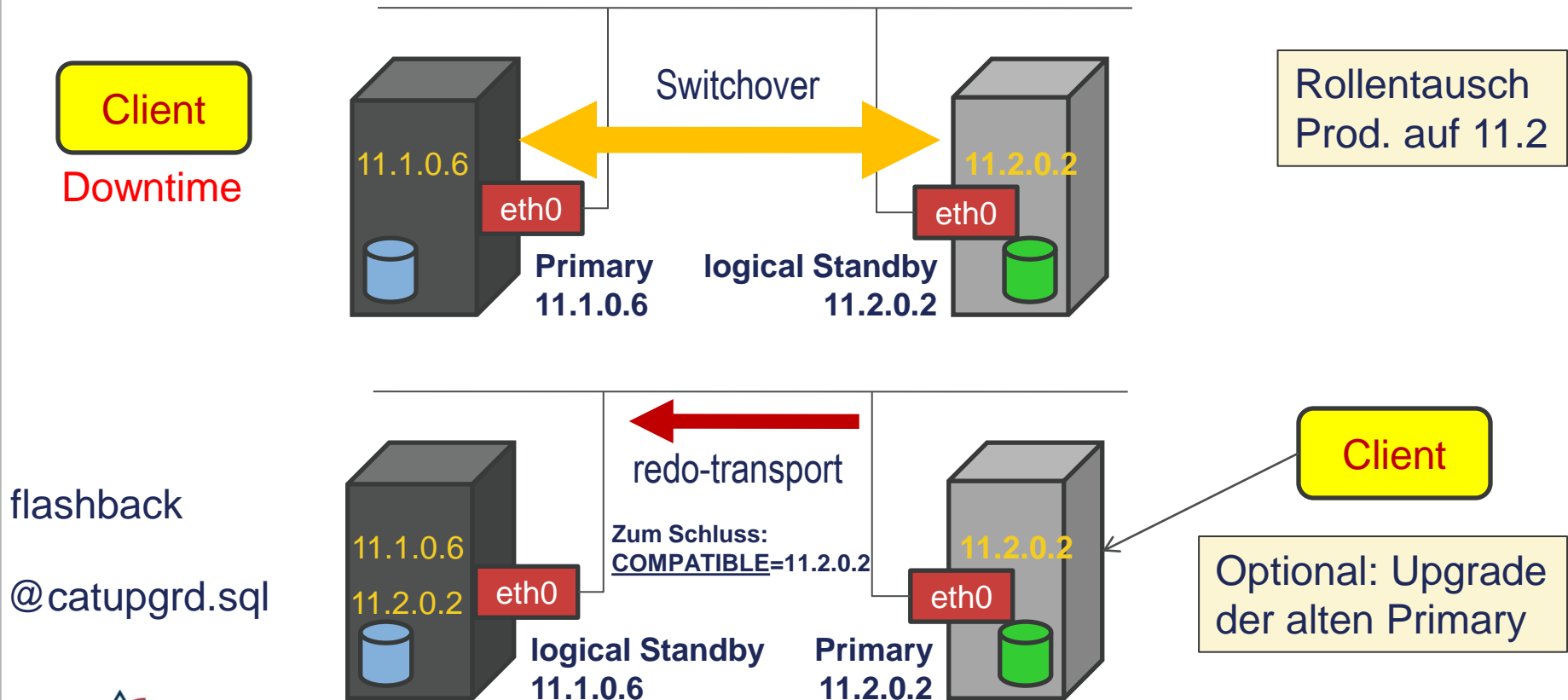
- Upgrade der Logical Standby
- Nachführen der Änderungen, die auf der (alten) Primary angefallen sind



Rolling Migration

■ Migration abschliessen

- Switchover auf die migrierte Logical Standby (kurze Downtime für Clients)
- Die alte Primary kann nun auf die neue Version portiert werden



Mixed Version (Redo Transport) Support

- **MOS Note 785347.1: Enterprise Edition 10.1.0.3 to 11.1.0.7**
- **Streams**
 - COMPATIBLE: Target \geq Source
- **Logical Standby for Rolling Upgrade**
 - Seit 10.1.0.3
 - COMPATIBLE: Target = Source
- **Physical DataGuard**
 - Target \neq Source **NOT SUPPORTED**
 - NEU: 11.1 \rightarrow 11.2 Rolling Upgrade via Transient Logical Standby
(Script: 949322.1 und MAA=*Maximum Availability Architecture* Literatur)
 - Upgrade der physical Standby (Logical: CONVERT TO PHYSICAL, aus neuem Home gestartet) via Apply der Redos der neuen Primary DB

Rolling Migration mit Plattform-Wechsel

Standby Primary	Mswin 32bit (id=7)	Mswin 64bit (id=12)	Linux 32bit (id=10)	Linux 64bit (id=13)	HPUX PA-RISC (ID=3)	HPUX Itanium (ID=4)	Solaris x86-64 (ID=20)	Solaris Sparc (ID=2)	AIX Power (ID=6)
Mswin32		32→64							
Mswin64	32→64			(*)					
Linux32				32→64					
Linux64		(*)	32→64						
HPUX pa-risc									
HPUX Itanium									
Solaris x86-64									
Solaris Sparc									
IBM AIX									

Metalink 1085687.1

(*) nur auf Itanium, aber nicht x86-64

7 **Fazit**

Hardware-Migration

■ Keine generelle Best Practice

- Abhängig von den Unterschieden der Plattform

■ Reduktion der Downtime

- Kopieren der Datenbankfiles ist schneller als Export/Import der Daten
 - Keine Neu-Berechnung von Indizes
- Ab Beginn des Kopierens muss die Datenbank gestoppt sein
 - Sonst gehen nachfolgende Änderungen verloren
- Diese Einschränkung fällt bei Migration mit Data Guard weg
 - Es werden auch Datafiles kopiert für die Standby, aber dies geschieht bei laufender Produktion, also ohne Downtime
 - Durch Applizierung der Änderungen haben wir immer eine aktuelle Kopie, die schon bereit steht

Datenbank-Upgrade

■ **Prinzipielle Varianten**

- Upgrade der bestehenden Datenbank
- Statt Upgrade: Datenübernahme in neuere Datenbank

■ **Problem der Downtime**

- Während Upgrade-Script (catupgrd.sql) läuft, kein Client-Zugriff
- Lange Downtime bei vielen Objekten

■ **Möglichkeiten zur Downtime-Reduktion**

- Vorherige De-Installation nicht benötigter Komponenten
 - Wird Java, XML, Oracle Text usw. wirklich verwendet?
- Rolling Upgrade
 - Verwendung einer Logical Standby Database
- Transportable Tablespace
 - Analog zum Beispiel der Plattform-Migration
 - Tablespaces werden an eine neuere Datenbank angehängt
- Minimal Downtime mit zusätzlicher Replikation (GoldenGate, Shareplex,...)

Datenbank-Upgrade

■ Tipps

■ Upgrade Advisors

- **Oracle Support Upgrade Advisors [ID 250.1]** mit vielen Verweisen auf weitere Dokumente
 - Database Upgrade from 9.2 to 11.2 [264.1]
 - Database Upgrade from 10.2 to 11.2 [251.1]

■ TTS – Transportable Tablespaces

- **Master Note for Transportable Tablespaces (TTS) -- Common Questions and Issues [ID 1166564.1]**
 - 10g : Transportable Tablespaces Across Different Platforms [ID 243304.1]
 - What Objects Are Exported With Transportable Tablespaces (TTS)? [ID 883153.1]

■ DataGuard Physical Standby

- **Master Note for Data Guard [ID 1101938.1]**

■ DataGuard Logical Standby

- Mixed Oracle Version support with Data Guard Redo Transport Services [ID 785347.1]
- Data Guard Support for Heterogeneous Primary and Logical Standbys in Same Data Guard Configuration [ID 1085687.1]
- Troubleshooting Logical Standby [ID 215020.1]

■ MAA – Oracle Maximum Availability Architecture

- <http://www.oracle.com/technetwork/database/features/availability/index.html>

Datenbank-Upgrade

■ Tipps

- Ultimate Tipp =
- http://blogs.oracle.com/UPGRADE/entry/find_me_at_the_beaches_and_her
- Bei Mike Dietrich, von Oracle

Fragen und Antworten



Kontakt Daten

- **Andrew Lacy, Senior Consultant**



OPITZ CONSULTING München GmbH

andrew.lacy@opitz-consulting.com

Telefon +49 89 680098 0

Mobil +49 173 173 279