

Herausforderungen bei der Unicode Migration von Oracle Datenbanken







Agenda



- Vorstellung CarajanDB GmbH
- > Ziel
- > Zeichenkonvertierung
- Umstellung der Datenbank
- Migration mit Export / Import
- > Planung einer Migration
- Minimal Downtime Migration

CarajanDB



- > Experten mit über 30 Jahren Oracle Erfahrung
- > Spezialisten für
 - Backup & Recovery
 - Hochverfügbarkeit
 - Healthchecks
 - Performance Optimierung
 - Einsatz von Oracle Standard Edition
 - Oracle in virtuellen Umgebungen und in der Cloud
 - Oracle Migrationen (HW, Unicode, Konsolidierung, Standard Edition)
 - Monitoring (Grid / Cloud Control, HLMM, Foglight, Spotlight)
- Schulung und Workshops (Oracle, Toad)





Ziel einer Migration

Primäre Ziele der Umstellung



> Für den Kunden:

 "Es soll möglich sein, weitere Sprachen in der Datenbank zu nutzen (z.B. Kanji, Arabisiche Schriftzeichen, etc.)"

> Für die Anwendungsentwicklung

 "Es soll egal sein, in welcher Sprache die Daten eingegeben werden, d.h. keine Unterschiede bei der Programmierung geben"

> Für den Datenbank Administrator

- "Es dürfen keine Zeichen verloren gehen"
- "Die Umstellung soll möglichst einfach sein"



Zeichenkonvertierung

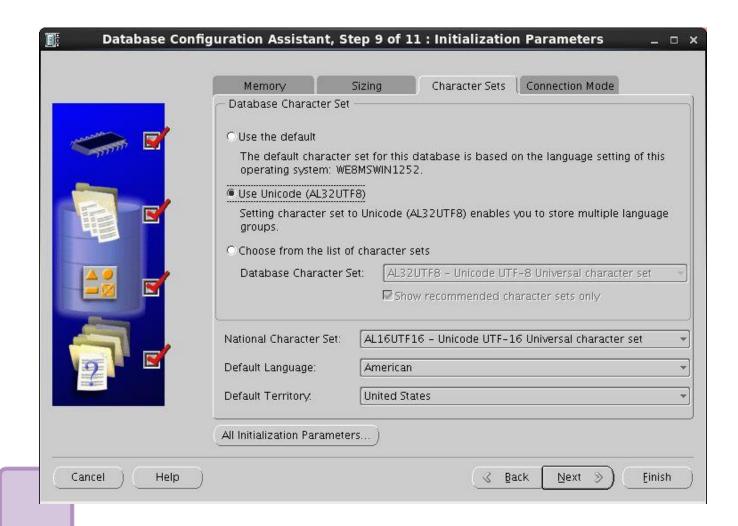
Zeichenkodierungen



Zeichensatz	A	Ö	ä	ß	€
US7ASCII	0x41	N/A	N/A	N/A	N/A
WE8PC850	0x41	0x99	0x84	0xE1	N/A
WE8ISO8859P1	0x41	0xD6	0xE4	0xDF	N/A
WE8ISO8859P15	0x41	0xD6	0xE4	0xDF	0xA4
WE8MSWIN1252	0x41	0xD6	0xE4	0xDF	0x80
AL32UTF8	0x41	0xC3 96	0xC3 A4	0xC3 9F	0xE2 82 AC

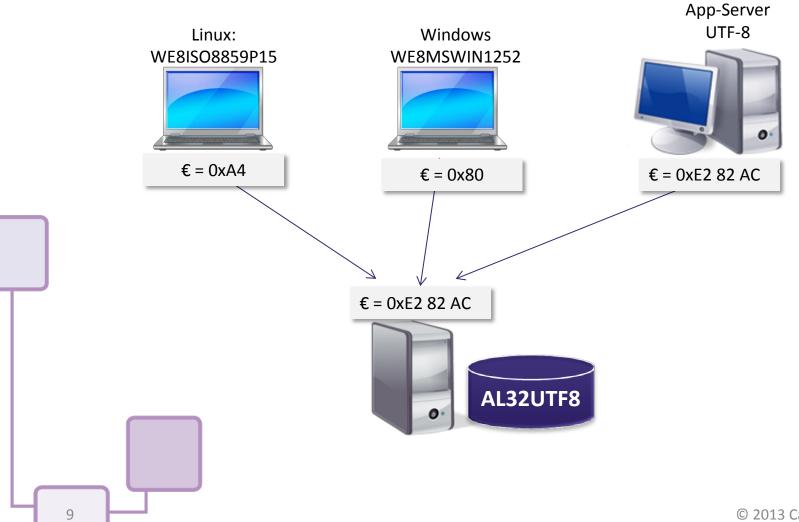
Auswahl des Charactersets





Zeichenkonvertierung





... und die Ziele?



> Für den Kunden → OKAY

 "Es soll möglich sein, weitere Sprachen in der Datenbank zu nutzen (z.B. Kanji, Arabisiche Schriftzeichen, etc.)"

➤ Für die Anwendungsentwicklung → ???

 "Es soll egal sein, in welcher Sprache die Daten eingegeben werden, d.h. keine Unterschiede bei der Programmierung geben"

➤ Für den Datenbank Administrator → ???

- "Es dürfen keine Zeichen verloren gehen"
- "Die Umstellung soll möglichst einfach sein"

Ein Beispiel (1)



Client: DOS-Box:

```
SQL> insert into umlaute values (1,'Test ü ß');

1 Zeile wurde erstellt.

SQL> commit;

Transaktion mit COMMIT abgeschlossen.

SQL> select * from umlaute;

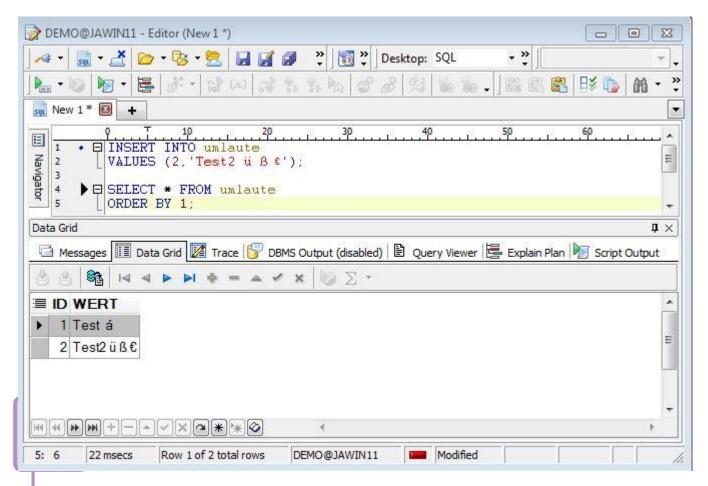
ID WERT

1 Test ü ß
```

Ein Beispiel (2)



Windows Tool (Toad)



Wer hat recht?



```
SELECT s.SID, s.username, s.PROGRAM,
           i.client charset, i.client version
 FROM v$session s, v$session connect info i
WHERE i.SID=s.SID
  AND s.PROGRAM IN ('Toad.exe', 'sqlplus.exe')
  AND s.username='DEMO'
  AND i.network service banner LIKE 'Windows NT%';
   SID USERNAME
                 PROGRAM
                                  CLIENT CHARSET CLIENT VERSION
  331 DEMO
              Toad.exe
                                UTF16
                                                  11.2.0.3.0
                 Toad.exe
                                                  11.2.0.3.0
  482 DEMO
                                  UTF16
                                WE8MSWIN1252
                                                  11.2.0.3.0
   484 DEMO
                  sqlplus.exe
```

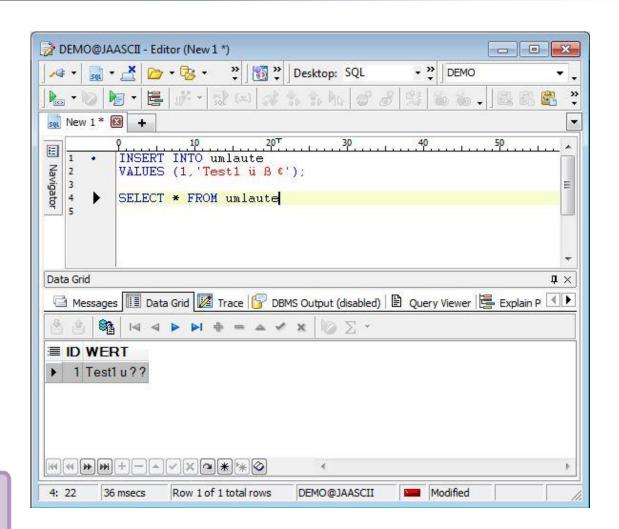
Umlaute in US7ASCII



- > Das geht doch gar nicht!
- > Aber was, wenn ich der Datenbank einfach mitteile, dass keine Konvertierung notwendig ist?

Toad





15

... und Putty?



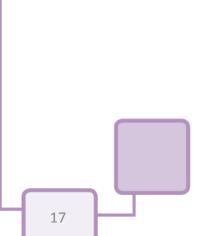
```
insert into umlaute
values (2, 'Test2 ü ß Ö');
1 row created.
SQL> commit;
Commit complete.
SQL> select * from umlaute
  2 ;
        ID WERT
         1 Test1 u ? ?
         2 Test2 üßÖ
```

Warum?



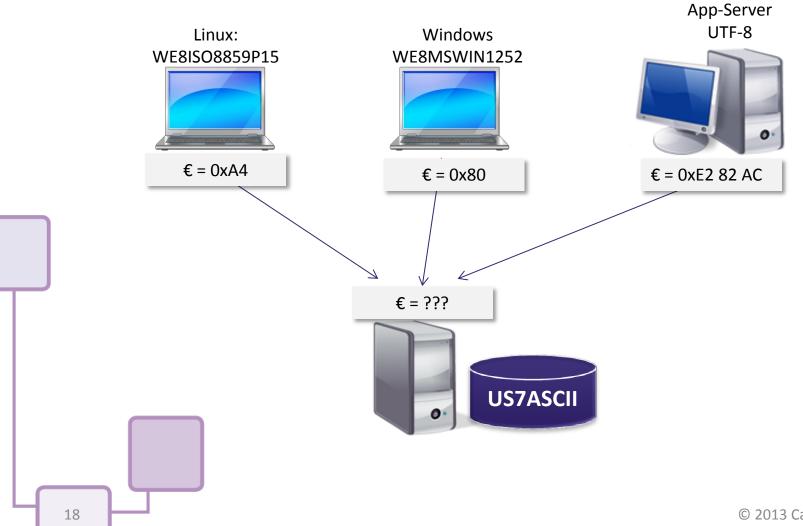
export NLS LANG=GERMAN GERMANY.US7ASCII

> Damit geht die Datenbank (US7ASCII) davon aus, dass keine Konvertierung notwendig ist!



Ein Extrembeispiel





Lösung



- > Herausfinden, mit welchen Zeichensatz die Daten eingegeben wurden
- > Datenbank auf diesen Zeichensatz einstellen
- Daten auslesen
- > Neue Datenbank erstellen (mit Unicode Zeichensatz)
- Daten wieder einlesen
- Auszeit für die Anwendung bis zu Tagen!!!

Technische Lösung



> Starten der Datenbank im Restricted Mode:

SQL> STARTUP RESTRICT

> Switch des Datenbank Zeichensatzes:

SQL> ALTER DATABASE

CHARACTERSET INTERNAL USE WE8ISO8859P15;

- > Export der Schemata, die diesen Zeichensatz benutzen
- > Switch des Datenbank Zeichensatzes:

SQL> ALTER DATABASE

CHARACTERSET INTERNAL USE AL32UTF8;

> Export der Schemata, die diesen Zeichensatz benutzen

... und die Ziele?



> Für den Kunden -> EGAL

 "Es soll möglich sein, weitere Sprachen in der Datenbank zu nutzen (z.B. Kanji, Arabisiche Schriftzeichen, etc.)"

➤ Für die Anwendungsentwicklung → ???

 "Es soll egal sein, in welcher Sprache die Daten eingegeben werden, d.h. keine Unterschiede bei der Programmierung geben"

➤ Für den Datenbank Administrator → PROBLEM!!!

- "Es dürfen keine Zeichen verloren gehen"
- "Die Umstellung soll möglichst einfach sein"



3. Migration mit Export / Import

Export der alten Datenbank



- Muss es unbedingt ein Full Export sein?
- ➤ → Besser Schemaexport!

```
DUMPFILE="export.dmp"
LOGFILE="exp_export.log"
DIRECTORY=DATA_PUMP_DIR
FLASHBACK_TIME="TO_TIMESTAMP('2012-10-13 09:00:00','YYYYY-MM-DD HH24:MI:SS')"
COMPRESSION=NONE
CONTENT=ALL
SCHEMAS=('BASIS', 'DEMO')
```

> Dadurch Bereinigung (z.B. HR Schema)

Daten importieren



- > Kopieren der Dump Datei auf den Zielrechner
- > Alternative: Datenbank Link verwenden

```
DUMPFILE="export.dmp"

LOGFILE="imp_export.log"

DIRECTORY=DATA_PUMP_DIR

STREAMS_CONFIGURATION=n

TABLE_EXISTS_ACTION=SKIP

SKIP_UNUSABLE_INDEXES=y

CONTENT=ALL

PARTITION_OPTIONS=none
```

Erfolgreich?



```
Processing object type SCHEMA EXPORT/SEQUENCE/SEQUENCE
Processing object type SCHEMA EXPORT/TABLE/TABLE
ORA-39083: Object type TABLE: "DEMO". "AUFTRAEGE" failed to create with error:
ORA-00439: feature not enabled: Partitioning
Failing sql is:
CREATE TABLE "DEMO". "AUFTRAEGE" ("AUFID" NUMBER(10,0) NOT NULL ENABLE, "PERSID" NUMBER(10,0) NOT
NULL ENABLE, "AUFDATUM" DATE, "LIEFERDATUM" DATE, "AUFSTATUS" CHAR(1 BYTE) NOT NULL ENABLE) PCTFREE
10 PCTUSED 0 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING STORAGE (INITIAL 65536 NEXT 1048576
MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 BUFFER POOL DEFAULT FLASH CACHE DEF
Processing object type SCHEMA EXPORT/TABLE/TABLE DATA
. . imported "DEMO"."PRODUKTE"
                                                        172.5 MB 374 rows
. . imported "DEMO". "POSITIONEN"
                                                        714.9 KB
                                                                   29780 rows
. . imported "BASIS". "NACHNAMEN"
                                                        177.2 KB 11552 rows
. . imported "BASIS"."ORTE"
                                                        190.7 KB 6104 rows
. . imported "BASIS". "VORNAMEN"
                                                        178.2 KB 8856 rows
. . imported "DEMO"."ADRESSEN"
                                                        106.7 KB
                                                                   2000 rows
. . imported "DEMO". "PERSONEN"
                                                        39.68 KB
                                                                   1000 rows
. . imported "DEMO"."TELEFONE"
                                                        108.1 KB 4000 rows
. . imported "BASIS". "STRASSEN"
                                                        20.24 KB
                                                                   640 rows
                                                                    16 rows
. . imported "DEMO"."BUNDESLAENDER"
                                                        5.531 KB
. . imported "DEMO"."PRODUKTGRUPPEN"
                                                        10.61 KB 156 rows
ORA-02374: conversion error loading table "DEMO". "STATUS"
ORA-12899: value too large for column STATUSID (actual: 2, maximum: 1)
ORA-02372: data for row: STATUSID: 0X'C4'
. . imported "DEMO". "STATUS"
                                                        5.835 KB 5 out of 6 rows
```



Planung einer Migration

Längensemantik



> Tabellendefinition

- > Was bedeutet die Längenangabe?
 - CHAR(1) = 1 BYTE oder 1 Zeichen?
 - VARCHAR2(50) = 50 BYTE 0 der 50 Zeichen?
- Besser:
- > Alternativ:
 - ALTER SESSION SET NLS_LENGTH_SEMANTICS='CHAR';

Längensemantik ausschalten



> Längensemantik eingeschaltet

```
CREATE TABLE STATUS (
STATUSID CHAR(1 BYTE) NOT NULL,
KURZBESCHREIBUNG VARCHAR2(10 BYTE),
BESCHREIBUNG VARCHAR2(255 BYTE)
) TABLESPACE USERS;
```

> Längensemantik ausgeschaltet

```
CREATE TABLE STATUS (
STATUSID CHAR(1) NOT NULL,
KURZBESCHREIBUNG VARCHAR2(10),
BESCHREIBUNG VARCHAR2(255)
) TABLESPACE USERS;
```

Einschränkungen



- Maximale Länge der Datentypen:
 - CHAR = 2000 BYTE
 - VARCHAR2 = 4000 BYTE
- Was ist, wenn der Datentyp nicht ausreicht?
 - VARCHAR2(4000) → CLOB
- Maximale Länge von Objektnamen = 30 BYTE
- > Indexlänge maximal ein Oracle Block

Was ist mit der Größe der Datenbank PARAJANDB

```
SELECT SUM (bytes) / 1024 / 1024 MByte
FROM dba_segments
WHERE owner = 'DEMO'
AND segment_name NOT LIKE 'BIN$%';
```

```
MBYTE
-----
187,625
```

```
MBYTE
-----
356,1875
```

Warum?



```
SELECT segment_name, tablespace_name, bytes
  FROM dba_segments
WHERE owner = 'DEMO' AND segment_name NOT LIKE 'BIN$%'
ORDER BY bytes DESC;
```

Alte Datenbank:			
SEGMENT_NAME	TABLESPACE_NAME	BYTES	
SYS_LOB0000051856C00006\$\$	USERS	191889408	
POSITIONEN	USERS	983040	
PK_POSITIONEN	USERS	983040	
SYS_IL0000051856C00006\$\$	USERS	327680	

Neue Datenbank:					
SEGMENT_NAME	TABLESPACE_NAME	BYTES			
SYS_LOB0000014229C00006\$\$	USERS	369098752			
POSITIONEN	USERS	983040			
PK_POSITIONEN	USERS	655360			
AUFTRAEGE	USERS	458752			
SYS_IL0000014229C00006\$\$	USERS	393216			

Oracle Dokumentation



"Oracle recommends using Unicode for all new system deployments."

"Data in CLOB columns is stored in a format that is compatible with UCS-2 when the database character set is multibyte, such as UTF8 or AL32UTF8. This means that the storage space required for an English document doubles when the data is converted."

Vorgehen für eine Migration



- > Objektdefinitionen ohne Längensemantik exportieren
 - Z.B. Toad → Generate Schema Script
- > Anpassen der Definitionen
 - Ändern von VARCHAR2 → CLOB, wenn 4000 Zeichen
 - Ändern von NVARCHAR2 → VARCHAR2
 - Ändern von Objektnamen (hoffentlich nicht!)
- Erstellen der Schemata auf dem Zielsystem mit Längensemantik 'CHAR'
- > Ausschalten von Foreign Key Constraints
- Exportieren der Daten (CONTENT=DATA_ONLY)
- Importieren der Daten (TABLE_EXISTS_ACTION=APPEND)
- Foreign Key Constraints einschalten (NOVALIDATE)

... und die Ziele?



> Für den Kunden → OKAY

 "Es soll möglich sein, weitere Sprachen in der Datenbank zu nutzen (z.B. Kanji, Arabisiche Schriftzeichen, etc.)"

➤ Für die Anwendungsentwicklung → OKAY

 "Es soll egal sein, in welcher Sprache die Daten eingegeben werden, d.h. keine Unterschiede bei der Programmierung geben"

> Für den Datenbank Administrator

- "Es dürfen keine Zeichen verloren gehen" → OKAY
- "Die Umstellung soll möglichst einfach sein" →????



Minimal Downtime Migration

Kundenbeispiel BMW



> Herausforderungen

- Datenbankgröße ca. 12 TB (Fahrzeugdatenbank)
- Verwendung des so genannten National Language Character sets (NCHAR, NVARCHAR2, NCLOB)
 - Sollten nach CHAR, VARCHAR2, CLOB migriert werden
- Downtime < 4 Stunden

Lösung



- > Aufbau einer zweiten Datenbank
- ➤ Laden der Daten mit Oracle "Bordmitteln (export / import) und eigenen Skripten (für CLOBs) im laufenden Betrieb
 - D.h. während des Entladens kommen neue Datensätze bzw.
 Sätze ändern sich
 - Protokollierung der Änderungen
 - Nachfahren der Änderungen auf dem Zielsystem
- > Quest SharePlex for Oracle für die Replikation
 - Wichtigstes Kriterium: Bekomme ich einen exakten Punkt, um zu sagen: "ab hier beginnt die Replikation?" Derzeit nur mit SharePlex möglich!

Starten der Replikation

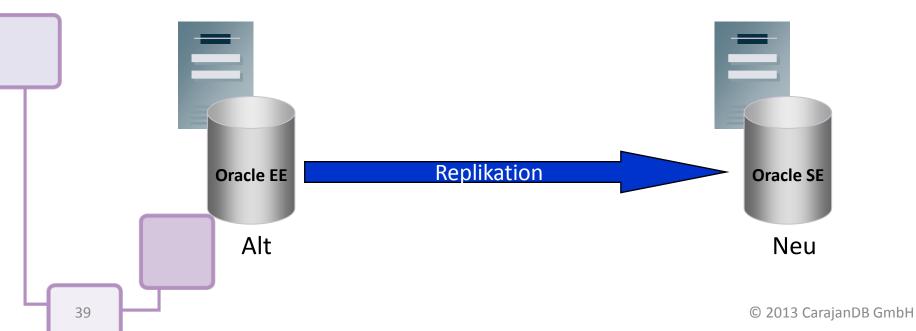


- Capture / Mine aller Änderungen mit der Replikationssoftware
- > Kein Apply / Post auf dem Zielsystem!!!
- Data Pump Export / Import mit Flashback SCN
- Reconcile des Apply / Post Prozesses auf die SCN
- > Starten des Apply / Post Prozesses
- Warten ...
- ➤ Datenbanken in Sync → Umdrehen der Replication (Fallback Lösung)

Minimal Downtime Migration



- > Migration von Zeichensätzen
- > Plattformwechsel
- Umstellung auf Oracle Standard Edition
- Migration nach Oracle xxx



Vorteile



- Geringe bis gar keine Downtime (in der Regel nur durch die Anwendung bzw. Reconnect der Application Server bestimmt)
- ➤ Migration während des laufenden Betriebs → nicht so fehleranfällig wie Nachtarbeit und Stress durch Zeitvorgaben
- ➤ Replikation in beide Richtungen möglich → Fallback, falls die Migration erst einmal nicht klappt

Voraussetzungen



- Möglichst keine Änderungen (DDL) während der Migrationsphase
- > Langer Migrationszeitraum (ca. 6 Monate oder mehr)
 - Allerdings sollte jede Migration ausgiebig getestet werden!!!
- > Klar definiertes Projekt
 - Kein "wir machen auch noch mal eben …"
- > Support durch Spezialisten

Primäre Ziele der Umstellung



➤ Für den Kunden → Okay

 "Es soll möglich sein, weitere Sprachen in der Datenbank zu nutzen (z.B. Kanji, Arabisiche Schriftzeichen, etc.)"

➤ Für die Anwendungsentwicklung → Okay

 "Es soll egal sein, in welcher Sprache die Daten eingegeben werden, d.h. keine Unterschiede bei der Programmierung geben"

➤ Für den Datenbank Administrator → Okay

- "Es dürfen keine Zeichen verloren gehen" → Okay
- "Die Umstellung soll möglichst einfach sein" → Okay

Zusammenfassung



- > Kritisch für die Umstellung auf Unicode sind:
 - Ausreizen der maximalen Datentyp Längen (4000 Byte)
 - "Schmutzzeichen" (keine Zeichenkonvertierung)
 - CLOB Länge
- Migration am Besten mit Data Pump Export / Import
 - Schemata vorher anlegen mit Längensemantic = "CHAR"
 - Einzelne Schemata exportieren und Importieren
- > Replikationslösungen für Minimal Downtime Migration
 - Migration im laufenden Betrieb
 - Fallback Möglichkeit



Fragen?

Johannes Ahrends www.carajandb.com



Johannes.ahrends@carajandb.com

