Datenbankmigration nach Unicode

Jan Schreiber, Loopback.ORG
Wolfgang Schick, Logica Deutschland
Hamburg / Frankfurt / Sulzbach

Schlüsselworte
Datenbank, Unicode, Migration

Einleitung
Ein international tätiges Logistikunternehmen plant seine Anwendung zur Transportabrechnung künftig nicht nur in Deutschland einzusetzen, sondern auch in seinen europäischen Ländergesellschaften. In diesem Vortrag wird die Umstellung von Datenbank, Anwendung und Schnittstellen auf den Unicode-Zeichensatz im Detail dargestellt und es wird deutlich, dass die Migration trotz Tool-Unterstützung auch fachlich kein triviales Unterfangen darstellt.

Ausgangssituation
Die Anwendung basiert auf Oracle Forms und der Oracle Datenbank 11gR2 und weist sehr viele Schnittstellen unterschiedlichster Natur und zum Teil älteren Datums auf.
Im Rahmen der europäischen Integration werden auch osteuropäische Ländergesellschaften angeschlossen. Durch den kyrillischen Zeichensatz, der im osteuropäischen Raum verwendet wird, entsteht die Notwendigkeit, die Anwendung so zu verändern, dass künftig dieser und andere Zeichensätze durch die Anwendung unterstützt werden. So müssen für die aktuellen Vorhaben zur Rechnungsstellung z.B. Firmen-, Orts-, Straßennamen und Leistungsbeschreibungen kyrillische Zeichen enthalten können. Da die Datenbank im WE8ISO8859P15-Zeichensatz angelegt ist, ist dies bisher nicht möglich.

Unicode als Standard-Zeichensatz
Für neue Datenbankinstallationen gilt seit längerem die Empfehlung, Unicode oder UTF-8 zu verwenden. Auch in Java- und XML-Umgebungen ist die Verwendung von Unicode seit langem üblich. Neue Oracle-Datenbank-Installationen werden vom Oracle Universal Installer als Default im Oracle-Zeichensatz AL32UTF8 (NCHAR AL32UTF16) angelegt. Dieser Zeichensatz wird auch als UTF-8 bezeichnet und sollte nicht mit dem Oracle-Zeichensatz UTF8 verwechselt werden. Bei Oracles UTF8, auch als CESU-8 bezeichnet, handelt es sich um einen Legacy-Zeichensatz, dessen Verwendung nicht mehr empfohlen wird. Er stammt noch aus den Anfangszeiten von Oracles Engagement im Unicode-Bereich.
AL32UTF8 ist ein dynamischer Multibyte-Zeichensatz, der 1-4 Bytes pro abzubildendem Zeichen verwendet. Für ASCII-Zeichen wird 1 Byte, für europäische Zeichen und solche aus dem mittleren Osten werden üblicherweise 2 Bytes und für asiatische Zeichen in der Regel 3 Bytes verwendet. AL32UTF8 entspricht der Unicode 4.0-Konvention.

Migrationspfade nach AL32UTF8
Traditionell gibt es zwei Arten der Migration WE8ISO8859P15-Zeichensätzen nach AL32UTF8:

- Die Export-Import Methode:
  Bei dieser Methode muss eine neue Datenbank mit dem Unicode-Zeichensatz aufgebaut werden. Per Export werden die Daten aus der „alten“ Datenbank exportiert und anschließend mittels Import in die neue Datenbank aufgenommen.
- CSSCAN/CSALTER-Tools:

Bei dieser Methode wird mit dem Oracle-Tool CSSCAN zunächst überprüft, welche Konflikte in den vorhandenen Daten gegenüber dem Ziel-Zeichensatz vorhanden sind. Bestimmte Probleme mit den Ausgangsdaten müssen dann manuell oder durch Reimport gelöst werden.

Darüber hinaus gibt es zwei neuere Ansätze:

- DMU Live-Migration:
  Das Thema Unicode-Migration scheint in der letzten Zeit an Aktualität gewonnen zu haben, so dass sich Oracle dazu entschlossen hat, mit der Datenbankversion 11 ein Java GUI Tool zu entwickeln, welches versucht, den gesamten Prozess in einem geführten Workflow abzubilden. Der "Database Assistant for Unicode" scannt die Datenbank, bereitet die Ergebnisse in einer anschaulichen Übersicht auf und bietet zur Behebung von Qualitätsproblemen einen eigenen Dateneditor, der die Konvertierung und eine abschließende Kontrolle durchführt.
- Abgehängte Migration mittels Streams:
  Bei der abgehängten Migration mit Streams wird die Datenbank zunächst geklont. Zwischen der Originaldatenbank und der Kopie wird eine Streams-Replikation eingerichtet. Ist die Replikation abgeschlossen, wird die replizierte Datenbank angehalten. Jetzt kann diese Kopie ohne Zeitdruck mit dem DMU-Tool konvertiert werden. Abschließend wird die Replikation aktiviert, und die Änderungen, die sich in der Zwischenzeit auf der Originalseite angesammelt haben, werden übertragen. Diese Methode bietet eine ganze Reihe von Vorteilen: Sie ist neben der kurzen Umschaltzeit beliebig oft wiederholbar, die konvertierte Datenbank kann beliebig intensiv getestet werden, bevor endgültig auf sie geschaltet wird.

Auswahl der Migrationsmethode

In der nächsten Phase unserer Untersuchungen wurde mit CSSCAN und DMU eine Bestandsaufnahme auf den in der Ausgangsdatenbank vorhandenen Daten erhoben. Dabei wurden verschiedene Kategorien von Problemen mit den Ausgangszeichen identifiziert:

- Daten, die nicht konvertiert werden müssen. Hierbei handelt es sich um Zahlen oder Zeichen, die im Ausgangs- und Ziel-Zeichensatz identisch abgebildet werden.
- Daten, die konvertiert werden können und müssen. In diese Kategorie fielen ca. 1% der Daten.
- Daten, für die vor der Konvertierung eine Anpassung der Feldlängen im Datenmodell durchgeführt werden muss ("over column limit"). Die Problematik mit solchen Datenbankfeldern resultiert aus den Feldlängenangaben in Byte, die der Anzahl der maximal möglichen Zeichen entspricht.
- Zeichenketten, die im Multibyte-Zeichensatz länger werden, als es der VARCHAR2-Datentyp erlaubt ("over type limit").
- Daten, die offensichtlich nicht im aktuellen Datenbankzeichensatz vorliegen ("invalid binary representation").

In Zusammenhang mit einem gesetzten engen Zeitfenster und den gefundenen Anomalien im Datenbestand sowie im Hinblick auf die verfügbaren Ressourcen erschien für diesen Fall die Live-Migration mit DMU als die effizienteste Methode.

Sonderzeichen im Data Dictionary

Sonderzeichen im Data Dictionary kann das DMU-Tool nur in wenigen Fällen konvertieren. In unserem Vorhaben sorgten insbesondere Umlaute in Trigger-Definitionen und PL/SQL Packages für solche Probleme.

Eine Möglichkeit, damit umzugehen, ist, diese Objekte einzeln zu überarbeiten und entsprechende Sonderzeichen zu entfernen. Bei umfangreicher installierter Software ist das meist wegen der entsprechenden Menge nicht möglich. Eine Alternative ist, diese Objekte zu ex- und nach der Migration wieder zu importieren, oder die Software komplett zu löschen und neu zu installieren.

Bestandsaufnahme der Datenbasis

Ein paar Erläuterungen zu den vorgefundenen problematischen Sonderzeichen:

Decimal-Wert	Hex-Wert	Win	8859-15	Erläuterung, wenn ersichtlich
142	8E	Ä	Ž	A accent grave, oder Z mit Umlauten
145,146	91,92	æ,Æ	`, ´	Einfache Hochkommata oder skandinavisches AE
159	9F	ƒ	Y	Florin oder gespiegeltes Y mit Umlauten, oder „-„
133	85	à	…	Ellipsis (…) als Zeichen, oder a accent grave
149	95	ò	•	o accent grave oder Bullit Point °
151	97	ù	—	Doppeler Unterstrich (mdash) „__“, oder u accent grave
138	8A	è	Š	S mit Umlauten, oder e accent grave
130	82	é	,	Komma, oder e accent aigu

Abb. 1: Einige der vorgefundenen ungültigen Sonderzeichen

In der Bestandsaufnahme erfolgte zunächst ein Scan mit dem DMU-Tool. Dazu musste zunächst die DMU-Software installiert werden. Für die Installation des DMU-Tools kommt jeder Computer mit Verbindung zur Datenbank in Frage, da die Java-Software des DMU-Tools prinzipiell unter jedem System mit Java JDK1.6 läuft. Zusätzlich muss in der Datenbank das Package `SYS.DBMS_DUMA_INTERNAL` installiert werden. Als Datenbankversion für die Nutzung des DMU-Tools ist mindestens Version 11.2.0.3 empfohlen, frühere Datenbankversionen erfordern das Einspielen von Datenbank-Patches.
Als ideale Laufzeit-Umgebung für DMU sehen wir einen Terminalserver oder einen Client in einer Virtual-Desktop-Umgebung, weil die Abläufe mit dem DMU-Tool in der Regel relativ lange dauern, und ein Netzwerkausfall während der Migration zwischen Datenbank und DMU-Client, ebenso wie ein Absturz des Client-Computers selbst verheerend wäre.
Der Scan mit dem DMU-Tool erfordert SYSDBA-Berechtigungen auf der Datenbank und kann während der produktiven Nutzung der Datenbank erfolgen. Die Parallelität, mit der das DMU-Tool die Datenbank scannt, kann so konfiguriert werden, dass nur ein Prozess läuft, der die Datenbank lediglich minimal belastet. Damit dauert der Scan entsprechend länger als bei einer Parallelisierung mehrerer Prozesse.
In der Praxis konnten wir in unserem Vorhaben mit acht Scan-Prozessen ohne Probleme während des produktiven Betriebes arbeiten. In den Nachtstunden konnten wir die Anzahl der Prozesse sogar auf 16 erhöhen. Ein Scan dauerte in der Produktionsumgebung mit 16 Prozessen etwa drei Stunden.

Bereinigung

Ist die Bestandsaufnahme abgeschlossen, kann in die Ergebnisanalyse eingestiegen werden.
In unserem Vorhaben hatten wir zunächst versucht, die ermittelten Probleme in möglichst viele gleichartige Klassen zu unterteilen, deren Behebung dann mit Skripten angegangen werden konnte. Zunächst wurde ein Skript erstellt, das alle Tabellen mit BYTE-Spaltendefinitionen in CHAR-Definition umwandelte. Dies konnte schlicht mittels “`alter table .. modify`” erfolgen.
Dann wurden individuelle Lösungen für die “over type limit”-Fälle entwickelt, meist konnten hier VARCHAR2-Felder in CLOB-Felder umgewandelt werden. Dazu hatten wir eine temporäre CLOB-Spalte angelegt, den Inhalt der VARCHAR2-Spalte mit einem UPDATE-Statement in die temporäre CLOB-Spalte kopiert, die VARCHAR2-Spalte gelöscht und zum Abschluss die temporäre CLOB-Spalte in den ursprünglichen Namen der VARCHAR-Spalte umbenannt.
Bei die “Invalid Binary Representation”-Fälle war ein wenig detektivischer Spürsinn gefragt. Mit etwas Phantasie gelang es allerdings in den meisten Fällen, den ursprünglichen Zeichensatz zu ermitteln. Dann konnten mittels speziell entwickelter Update-Statements die ungültigen Zeichen zusammengefasst bereinigt werden.

Die verbliebenen Fälle mit falscher Schreibweise machten weniger als zehn Prozent der Gesamtmenge aus und wurden durch Einzelfallbearbeitung bereinigt. Das DMU-Tool bietet hierzu ein intuitiv zu bedienendes Interface an, das die fragwürdigen Zeichen farbig markiert.

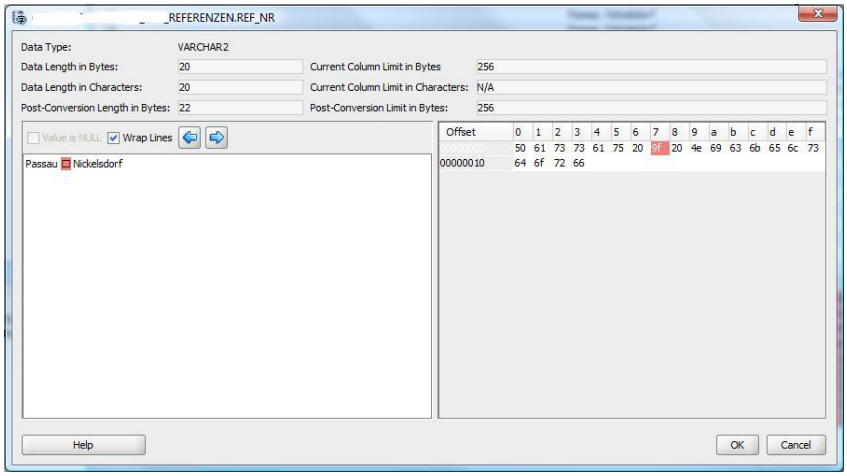

Abb. 2: Einzelsatzbearbeitung im DMU-Tool

Wenn die in der Datenbank vorhandene Software in einem eigenen Schema installiert und von dem Datenschema getrennt ist, kann man das Software-Schema exportieren und anschließend löschen. Anschließend befinden sich keine problematischen Objekte mehr im Data Dictionary.
In unserem Fall mussten die betroffenen Trigger und Packages per Skript gelöscht werden. Zum anschließenden Installieren der Software nach der Migration muss die Software neu installiert werden können.
Zum Abschluss empfiehlt es sich, einen Blick auf den Volumenzuwachs zu werfen. Durch die Multibyte-Repräsentation wächst während der Konvertierung das Datenvolumen an. Aus diesem Grund ist es erforderlich, entsprechend Platz in ASM oder den File-Systemen der Datenbank vorzusehen und darauf zu achten, dass die Datafiles für die Migration hinreichend Reserven aufweisen oder auf AUTOEXTEND stehen.
In unserem Vorhaben hatten wir -übereinstimmend mit der Vorabschätzung des DMU-Tools- etwa 10% Volumenzuwachs ermittelt.

Konvertierung
Vor der Konvertierung muss sichergestellt werden, dass für die umfangreichen Updates, die das DMU-Tool durchführt, genügend Platz für die Archive Logs bereitsteht, bzw. entschieden werden, ob es sinnvoll ist, ARCHIVELOG während der Migration komplett abzuschalten. Die Abschaltung des ARCHIVELOG wirkt sich naturgemäß sehr positiv auf die Laufzeit aus.
In der Version 1.1 des DMU-Tools gibt es einen Bug. Dieser zeigt sich während der Konvertierung in einem Abbruch mit dem Fehler "`ORA-22839: Direct updates on SYS_NC columns are disallowed`". Um das Auftreten des Bugs zu vermeiden, sollte vor der Konvertierung der Event 22838 mit folgenden Kommando: "`alter system set events '22838 TRACENAME CONTEXT LEVEL 1, FOREVER'`" gesetzt werden. Vor der Konvertierung sollten zusätzlich noch alle Jobs abgeschaltet werden ("`alter system set job_queue_processes=0`").
Die Konvertierung selbst konnte ohne Probleme durchgeführt werden. Die Konvertierungszeit lag mit 17 Stunden im errechneten Zeitfenster. Sollten während der Konvertierung einfache Probleme wie eine volle FRA oder ein fehlendes AUTOEXTEND auftreten, hält das DMU-Tool an und bietet die Möglichkeit, den Vorgang fortzusetzen, wenn das Problem behoben wurde.

Die Konvertierung der Abnahme- und Produktionssysteme benötigt eine enge Koordination der Projekt- und Operations-Teams, da sie mit dem Tool selbst und nicht über SQL-Skripte erfolgt. Schon für den interaktiven Teil der Bereinigungsphase werden SYSDBA-Rechte benötigt. Sind diese Privilegien dem Operating vorbehalten, können Massenbereinigungen vorab per Skript und die interaktive Einzelfallbehandlung kurz vor der Konvertierung mit dem DMU-Tool vorgenommen werden. Soll das Tool vom Operating verwendet werden, muss eine Einarbeitung des Personals erfolgen. Der Aufwand dafür ist nicht zu vernachlässigen. In unserem Fall wurde ein separates Team, in dem Projekt- und DBA-Skills gemeinsam vorhanden waren, für die Migration gebildet.

Nacharbeiten

Nach der Konvertierung ist der Datenbank-Zeichensatz umgestellt, aus diesem Grund ist es meist noch erforderlich, entsprechende Client-Settings, beispielsweise die Konfiguration der NLS_LANG-Parameter, anzupassen

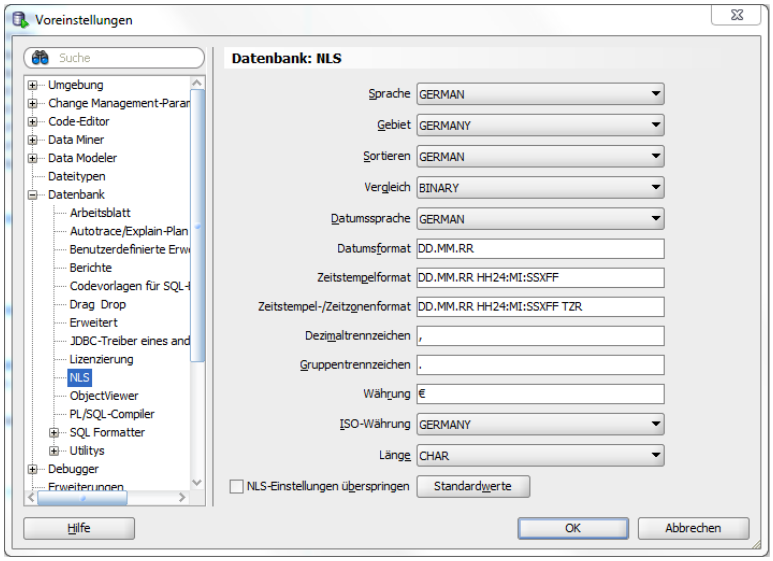

Abb. 3: SQL-Developer-Einstellungen

Des Weiteren ist es empfehlenswert, den Parameter `NLS_LENGTH_SEMANTICS=CHAR` in der Datenbank zu setzen und die NLS_LENGTH-SEMANTICS-Konfiguration der darauf zugreifenden Clients anzupassen, damit zukünftige Objekte nicht in der BYTE-Semantik angelegt werden. Beim SQL Developer von Oracle kann diese Einstellung in den Settings vorgenommen werden.
Auch die Software-Lieferprozesse, die oft in der Unix-Shell mittels SQLPlus erfolgen, sollten so geändert werden, dass diese mit der richtigen Konfiguration gestartet werden, ein besonderes Augenmerk ist hier auf die NLS_LANG-Einstellungen zu legen. Wenn Software-Pakete weiterhin im alten Zeichensatz geliefert und eingespielt werden sollen, muss die Variable NLS_LANG beim Aufruf von SQLPlus auf dem Wert für den Zeichensatz stehen, in dem die Datei erzeugt wurde.
Nach dem ggf. erforderlichen Re-Import der Software sollten alle Datenbankobjekte unter besonderer Beachtung der korrekten NLS-Settings neu kompiliert werden – ebenso müssen ggf. vorhandene Views und Materialized Views, die sich vorher auf Tabellen mit BYTE-Semantik bezogen haben, neu kompiliert werden. Es empfiehlt sich, eine abschließende Kontrolle der dba_tab_columns (CHAR_USED=C) und der dba_plsql_object_settings (nls_length_semantics) vorzunehmen und zu ermitteln, ob alle Objekte in Char-Semantik vorliegen.

Wird Forms verwendet, müssen auch auf dem Forms-Server die NLS-Konfiguration angepasst sowie alle Forms-Module neu übersetzt werden, da sich die Signatur in der Datenbank mit Sicherheit geändert hat.

Schnittstellen
Für dateibasierte Schnittstellen, inklusive XML, das direkt aus der Datenbank erzeugt wird, muss geprüft werden, ob diese zukünftig Dateien in Unicode liefern dürfen. Ist dies nicht der Fall, muss die Ausgabe entsprechend in den zuvor verwendeten Zeichensatz konvertiert werden – zum Beispiel mit Hilfe der Funktionen `utl_file.put_raw` und `utl_i18n.string_to_raw`.
Sind Clients, die über OCI auf die Datenbank zugreifen, nicht Unicode-fähig, können sie weiterhin im alten Zeichensatz betrieben werden, wenn deren NLS-Konfiguration entsprechend eingestellt ist (z.B. auf WE8ISO8859P15). Bei diesem Mischbetrieb können sich in der Folge allerdings Probleme beim Liefern von ISO8859-XML-Dateien ergeben.
Bisher im Datenbestand vorhandene Inhalte sind mit diesen Clients 1:1 abbildbar, weil AL32UTF8 eine Obermenge von WE8ISO8859P15 und ISO8859 ist. Nach der Unicode-Umstellung können im Laufe der weiteren Nutzung jedoch auch Unicode-Zeichen eingegeben werden, die in ISO8859 nicht mehr darstellbar sind. Diese Zeichen würden dann im XML durch das Ersetzungszeichen dargestellt werden. Im ungünstigsten Fall würde dann beispielsweise das Ersetzungszeichen „¿“ statt eines kyrillischen Zeichens ausgegeben werden.
Die Anbindung der Host-Zuliefersysteme über Connect:Direct und OTG gestaltete sich problemlos, da der dazugehörige Oracle-Client mit P15-Einstellung betrieben wurde und alle Konvertierungen vornehmen konnte.

Fazit
Für eine Unicode-Konvertierung mit schlankem Budget und engem Downtime-Fenster erwies das Oracle DMU-Tool als handhabbares erfolgreiches Verfahren mit gutem Ergebnis. Die vorangehende Datenbereinigung wird gut unterstützt und die Konvertierung lief bei allen Datenbanken zuverlässig und schnell ab.
Wir empfehlen die mit der Migration betrauten Personen und deren Vertreter von Anfang bis Ende in das Projekt einzubinden, weil die Einarbeitung in das Tool umfangreich und aufwendig ist.
Besonders aufwändig ist die Bereinigung von in der Ausgangsdatenbank vorhandenen Zeichen, die nicht zum Zeichensatz passen. Hier ist in der Regel eine enge Ansprache mit dem Fachbereich bezüglich der Datenqualität erforderlich.
Bei den Schnittstellen lohnt sich insbesondere ein Blick auf selbst erstellte dateibasierte Verfahren, die sich bzgl. des Zeichensatzes als problematisch erweisen könnten.

Kontaktadresse:

Jan Schreiber
Loopback.ORG GmbH
An der Alster 83
D-20099 Hamburg

Telefon: +49 (0) 40-2263236 0
Fax: +49 (0) 40-2263236 99
E-Mail jans@loopback.org
Internet: www.loopback.org

Wolfgang Schick
Logica Deutschland GmbH & Co. KG
Am Limespark 2
D-65843 Sulzbach

+49 (0) 6196 7742 0
+49 (0) 6196 7742 555
wolfgang.schick@cgi.com
www.cgi.com