Automatische Migration von Host-Dateien in relationale Datenbanken: Ein Praxis-Beispiel

Andreas Loos
pro et con Innovative Informatikanwendungen GmbH, Annaberger Str. 240, 09125 Chemnitz,
Telefon: 0371 5347 324, mailto:Andreas.Loos@proetcon.de

Abstract:

Migrationsprozesse sind komplexer Natur. Sie umfassen in der Regel Hardware-, Software- und Datenmigrationen. Ausgangs- und Zielsysteme unterscheiden sich in einer Vielzahl von Aspekten. Damit ist eine einfache „Übertragung“ irgendeiner Komponente nur in den seltensten Fällen möglich. In der Regel muß jeder Bestandteil des Ausgangssystems einer umfassenden Analyse und einer Transformation unterworfen werden. Das Ausgangssystem ist im Allgemeinen dadurch gekennzeichnet, daß es über viele Jahre gewachsen ist. Dokumentationen existieren daher nicht in jedem Fall. Analyse des Ausgangssystems und Migration sind aus diesen Gründen nur mit weitgehend automatisierter Werkzeug-Unterstützung in vertretbarer Zeit möglich.

Die in diesem Beitrag vorgestellte Daten-Migration ist Teil eines derzeit bearbeiteten, komplexen Projektes, welches die Migration einer vollständigen Applikation von einem Fujitsu-Siemens BS2000-System auf eine UNIX-Plattform beinhaltet.
Die Migration der Programme besteht in der Konvertierung der in der proprietären Programmiersprache SPL entwickelten Sourcen. Für diesen Zweck wurde von pro et con ein Translator entwickelt, welcher den SPL-Quellcode nahezu vollständig automatisch nach C++ konvertiert.

Der BS2000-Datenbestand der Applikation liegt als eine Menge von ca. 600 sequentiellen und indexsequentiellen Dateien vor. Jede Datei besteht aus einer Folge von Datensätzen, nahezu jeder Datensatz beinhaltet einen Satzschlüssel oder eine Satznummer.
Ziel ist eine Datenmigration derart, daß die Daten auf der Zielplattform in einer relationalen Datenbank mit dem Datenbanksystem Oracle 10g verwaltet werden können. Die gesamte Schicht der Datenverwaltung wird dabei gegenüber der Applikation transparent gehalten.
Im weiteren wird hauptsächlich der Prozeß der Datenmigration betrachtet. Dabei charakterisiert der Begriff „Migration“ den gesamten Prozeß, „Konvertierung“ dagegen den konkreten Schritt der Umwandlung einer BS2000-Datei in eine Oracle-Tabelle

Der Prozeß der Datenmigration wird im Projekt in folgende Abschnitte gegliedert:

Daten-Bereitstellung. Aus dem laufenden Betrieb muß ein Abzug aller BS2000-Dateien bereitgestellt werden. Dieser stellt nur einen Schnappschuß dar, welcher den Datenbestand zu einem gegebenen Zeitpunkt widerspiegelt. Die so bereitgestellten Daten sind keine originalen Migrationsdaten, sondern dienen nur Analyseprozessen.

Daten-Analyse. Die erhaltenen Dateien werden nach verschiedenen Gesichtspunkten klassifiziert. Für die Migration ist die Unterscheidung wesentlich, ob eine Datei zum Zeitpunkt des Plattformwechsels konvertiert werden muß (dynamische Dateiinhalte), oder ob bereits im Vorfeld eine Migration erfolgen kann (statische Dateiinhalte).
Der Plattformwechsel von BS2000 nach UNIX bedingt eine EBCDIC ->ASCII Konvertierung der Daten. Da Dateibeschreibungen nur in begrenztem Maße vorhanden sind, muß eine toolgestützte Dateianalyse Aussagen liefern, an welchen Stellen eines Datensatzes ein Zeichen als „textuell“ oder als „binär“ zu interpretieren ist. „Textuelle“ Bytes müssen während der Migration vom EBCDIC in die korrespondierende ASCII-Notation überführt werden. Zeichen hingegen, die „binär“ in den Dateien abgelegt sind, dürfen nicht modifiziert werden. Dieser Analyseprozeß wird in einem ersten Schritt automatisch realisiert. Dazu interpretiert ein Perl-Programm eine BS2000-Datei als Matrix, in der Datensätze als Zeilen, Bytepositionen als Spalten interpretiert werden. Jedes einzelne Zeichen wird sowohl autonom als auch im Kontext mit anderen Zeichen und bereits gewonnenen Erfahrungen als textuell oder binär klassifiziert. Auf diese Art und Weise kann ein signifikanter Anteil Dateien (ca. 80%) sicher bestimmt werden. Sind gesicherte Aussagen nicht möglich, so wird in einem zweiten Schritt eine Analyse mit dem von pro et con entwickelten Werkzeug Flow Graph Manipulator (FGM) vorgenommen.
Basis hierbei ist der Bestand an SPL- Sourcen. Es werden mittels komplexer Datenfluß- und Abhängigkeitsanalysen Datenstrukturen in den Programmen lokalisiert, in welchen die Sätze der BS2000-Dateien manipuliert werden. Auf diese Art und Weise wird die Strukturierung des Datensatzes ermittelt.

Daten-Konvertierung. Die Konvertierung der Daten erfolgt im vorliegenden Projekt so, daß für jede BS2000-Datei eine Oracle-Tabelle erzeugt wird. Jeder Satz einer BS2000-Datei wird zu einem Datensatz der entsprechenden Tabelle. Es erfolgt kein relationales (Re)-Design der Datenbank, sondern es werden lediglich die in den Datensätzen enthaltenen Schlüssel extrahiert und indiziert in einer separaten Tabellenspalte gehalten. Damit ist ein effizienter Zugriff auf die Datensätze möglich. Die Entscheidung, kein vollständig neues Datenbanklayout aufzusetzen, ist aus Projektsicht sinnvoll, da hier die Risiken der Datenmigration minimiert werden.
Die Konvertierung der BS2000-Dateien muß vollautomatisch erfolgen, da für die Migration selbst nur ein sehr enges Zeitfenster vorgesehen ist.
Ein Bestandteil der Konvertierung ist weiterhin die Generierung von Control-Skripten, in denen die für die Administration der Datenbank notwendigen SQL-Statements hinterlegt sind.

Daten-Ladeprozeß: Im Ergebnis der Daten-Konvertierung entstehen Dateien,, die mit dem Oracle SQL-Loader in die Datenbank geladen werden können. Dabei wurden EBCDIC-Daten durch ASCII-Werte ersetzt, wenn es sich um textuelle Teile handelte. Mit dem SQL-Loader wird ein Tool aus dem Oracle-Umfeld benutzt, welches als hinreichend performant und stabil gilt. Die notwendigen Oracle-Tabellen können bereits im Vorfeld angelegt werden, da deren Struktur unabhängig von den konkreten Daten ist. Hier wird mit SQLPLUS ebenfalls ein Oracle-Tool benutzt.

Zugriffsschicht: Bestandteil des Daten-migrationsprojektes ist auch die Migration der Zugriffsschicht, über welcher die Applikation mit der Datenbank kommuniziert. Diese Schicht ist im SPL-Sourcecode des Basissystems als eine Menge von Funktionen definiert. Aufrufe dieser Funktionen werden durch den Translator von SPL nach C++ konvertiert. Die eigentliche Implementierung der Zugriffsfunktionen erfolgt manuell, da im Zielsystem „embedded“-SQL-Statements die Datenmanipulation

realisieren. Bei der Implementierung der Zugriffsfunktionen müssen damit zwei wesentliche Aspekte berücksichtigt werden:

1. Sie müssen bzgl. der SPL-Zugriffsfunktionen semantisch äquivalent arbeiten, da die konkrete Art der Daten-Speicherung für die Applikation transparent bleibt.

2. Sie müssen sich zu den von SPL nach C++ konvertierten Programmen syntaktisch äquivalent verhalten.

Prototyp: Die Frage, inwieweit Oracle 10g den hohen Performance-Anforderungen der Applikation entspricht, wurde bereits im Vorfeld der Migration beantwortet. Dazu wurde ein Prototyp entwickelt, der das grundsätzliche Verhalten der Applikation mit Oracle-Mitteln simuliert. Mit ihm konnten wesentliche Erkenntnisse bezüglich des Datenbankdesigns gewonnen werden, Leistungsmessungen am Prototypen bewiesen eine ausreichende Performance.

Erfahrungen des Projektes lassen sich wie folgt formulieren:

1. Migrationen sind dynamische Prozesse, da während der Projektlaufzeit die Basis-Software und der Datenbestand permanenten Weiterentwicklungen unterworfen ist. Damit verbunden sind kurzfristige Änderungen des Projektinhaltes.

2. Der größte Teil der Migrationsprozesse ist dennoch automatisierbar und formalisierbar, es gibt Bereiche, in denen manuelle Arbeiten unumgänglich sind.

3. Es wird während der Migration mit unterschiedlichen Datenformaten gearbeitet, für die Lese-, Schreib- und Transformations-Funktionalität entwickelt werden muß.

4. Automatisch erzeugte Dokumentationen, die beispielsweise während der Analyse entstehen, müssen in zwei wesentlichen Kodierungsformaten vorliegen: Zum einen als maschinell weiterverarbeitbares Format und zum anderen als lesbares Format („humanes“ Format).

5. Während der gesamten Migration fallen große Mengen an Metadaten an, die nur maschinell auswertbar sind und deshalb in maschinell lesbarer Form aufbereitet werden müssen.

Zusammenfassung: Die im Rahmen dieses Migrationsprojektes entwickelten Tools wurden formalisiert und zu einer Toolbox zusammengestellt, mit der pro et con in der Lage ist, BS2000-Migrationen vollständig und in vielen Bereichen automatisiert zu realisieren. Dabei wird ein Spektrum abgedeckt, welches für BS2000 die Programmkonvertierung von SPL nach C++, die automatische Konvertierung von (JCL) SDF- Prozeduren nach Perl und die Migration der Mainframe-Daten in eine Oracle-Datenbank überstreicht.