

Erfahrungen bei der Migration von MS Access nach APEX

Markus Kühn
Landesbausparkasse Baden-Württemberg
Stuttgart

Schlüsselworte:

APEX, Database, MS Access, SQL*Developer, Migration

Einleitung

MS Access wird in vielen Unternehmen als datenbankgestütztes Entwicklungswerkzeug für kleine Anwendungen eingesetzt. Überlegungen, Microsoft Office nach Oracle OpenOffice zu migrieren, warfen die Frage auf, was nach dieser Migration mit den MS Access-Anwendungen geschieht, die in unserem Hause genutzt werden. Um einen möglichen Migrationsweg aufzuzeigen, wurden daraufhin in unserem Haus die Möglichkeiten, die der SQL*Developer und APEX bieten, evaluiert und getestet. Ebenfalls suchten wir nach weiteren Möglichkeiten und Vorgehensweisen für die Migration von bestehenden MS Access-Anwendungen nach Oracle APEX.

Dieser Vortrag zeigt keinen Standardpfad für eine Migration von MS Access nach Oracle APEX auf, sondern vielmehr die bei einer solchen Migration zu beachtenden Punkte und eine mögliche Vorgehensweise. Ebenfalls werden die Funktionalitäten des SQL*Developers und Oracle APEX gezeigt, die eine Migration von MS Access nach Oracle APEX unterstützen.

Migration – erste Schritte

Analyse

Ein wichtiger Schritt zur Vorbereitung der Migration von MS Access nach APEX ist die Analyse der Anwendung und des ihr zu Grunde liegenden Datenmodells. Das Ziel der Analyse ist die Bestimmung dessen, was benötigt wird, um eine erfolgreiche Migration durchzuführen. Dieses beinhaltet das Aufspüren von Mängeln des Datenmodells, Eigenheiten der Benutzeroberfläche, der Art der Nutzung von Visual Basic (VB) Code sowie anderer Designfragen. Insbesondere ist hier auf das Fehlen von referentieller Integrität zu achten.

In dieser Phase sollte man sich die Zeit nehmen, um Informationen von den Anwendern und Entwicklern zu erhalten. Denn niemand weiß besser wie eine Anwendung funktioniert, als deren Anwender und Entwickler. Ebenfalls sollte in dieser Phase die Erwartungshaltung der Anwender in die richtige Richtung gelenkt werden.

Design

Während dieses Schrittes wird geplant, wie das künftige Datenmodell bzw. die künftige Anwendung aussieht. Hier sollte ebenfalls, wie auch bei der Analyse, das Datenmodell und die Anwendung betrachtet werden. Mit den Informationen aus den Gesprächen mit den Anwendern und Entwicklern, sollte man hier in der Lage sein zu erkennen, welche Optimierungsmaßnahmen für eine erfolgreiche Migration durchgeführt werden müssen. Als weitere Tätigkeit innerhalb dieser Phase wird entschieden, welche Teile der Anwendung überhaupt migriert werden müssen. Schließlich wird innerhalb dieser Phase der VB-Skript Code betrachtet. Hier muss entschieden werden, ob dieser Code mit PL/SQL nachgebildet werden muss, oder ob hierfür APEX Komponenten benutzt werden können.

Entscheidungsgrundlagen für das Migrieren bzw. „Neumachen“

Während der Analyse der MS Access Anwendung wird entschieden, ob diese Anwendung ein Kandidat für eine Migration ist, oder ob sie neu entwickelt werden muss. In unserem Haus wurden für die Entscheidungsfindung folgende Grundlagen verwendet:

- **Einfache MS Access Anwendung**
Besteht eine MS Access Anwendung ausschließlich aus Tabellen, Formularen und Berichten, bietet sich eine Migration dann an, wenn mehr als ein Anwender mit dieser Anwendung arbeitet.
- **Datenkonsolidierung**
Werden mit Hilfe einer MS Access Anwendung Daten aus mehreren Datenbanken miteinander verknüpft und arbeiten mehrere Anwender mit dieser Anwendung, bietet sich eine Migration dann an, wenn sich ein Mehrwert für die Anwender ergibt.
- **Anwendungen mit komplexem VB-Code**
Beinhaltet eine MS Access Anwendung VB-Code, der leicht zu verstehen ist und der eine gewisse Größe nicht überschreitet, bietet sich eine Migration nach APEX an. Hierbei ist zu beachten, dass der VB-Code in PL/SQL überführt werden muss. Diese Arbeiten erfolgen manuell, da weder der SQL*Developer noch APEX eine Unterstützung für die Übernahme von VB-Code nach PL/SQL anbieten.
Ist der in einer MS Access Anwendung vorhandene VB-Code komplex und unverständlich, so verbleibt diese Anwendung in der MS Access Welt.
- **„Anhübschen“ einer MS Access Anwendung**
Soll eine MS Access Anwendung um zusätzliche Funktionalitäten (z.B. PDF-Druck, Mail-Versand, Flash-Grafiken) erweitert oder die Oberfläche einer Anwendung „Browser-tauglich“ gemacht werden, bietet sich ebenfalls eine Migration nach APEX an.

Mögliche Vorgehensweisen bei der Migration

Die folgenden Vorgehensweisen wurden im Hause LBS evaluiert.

Eine MS Access Datenbank – ohne Anwendung – kann mit Hilfe von Bordmitteln aus MS Access in eine Oracle Datenbank migriert werden. Dazu muss zunächst in MS Access über ODBC eine Verbindung zu einer Oracle Datenbank hergestellt werden. Danach können Tabellen aus der MS Access Datenbank nach Oracle exportiert bzw. in Oracle importiert werden. Bei diesem Vorgehen entstehen folgende Nachteile:

- Export bzw. Import nur tabellenweise. Je mehr Tabellen desto höher der Aufwand.
- Memo-Spalten einer MS Access Tabelle werden in LONG-Spalten einer Oracle Tabelle konvertiert. Besitzt eine MS Access Tabelle mehr als eine Memo-Spalte, dann kann diese auf diesem Weg nicht migriert werden.
- AutoWert-Spalten einer MS Access Tabelle werden in einer Oracle Datenbank mittels Sequenzen und Triggern abgebildet. Diese werden bei dieser Vorgehensweise in der Oracle Datenbank nicht erstellt. Daher ist eine manuelle Nacharbeit hier erforderlich.
- Indizes und Constraints werden auf diese Art und Weise ebenfalls nicht migriert. Das bedeutet, dass hier ebenfalls manuelle Nacharbeiten erforderlich sind.

Ab dem SQL*Developer 3.0 steht dem Anwender ein „Copy-To-Oracle“-Feature zur Verfügung mit Hilfe dessen eine Tabelle einer MS Access Datenbank – wiederum ohne Anwendung – in eine Oracle Datenbank migriert werden kann. Hierzu muss im SQL*Developer eine Verbindung zur einer MS Access Datenbank erstellt werden. Die darin vorhandenen Tabellen können dann mittels Mausklicks in eine Oracle Datenbank kopiert (migriert) werden. Dabei werden ebenfalls die Sequenzen und Trigger für das Befüllen der AutoWert-Spalten erstellt. Ansonsten gelten die gleichen Nachteile wie bei der Verwendung von MS Access Bordmitteln.

Eine Migration kann ebenfalls mit der *Oracle SQL*Developer Migration Workbench* und dem *Oracle Application Express Migration Workshop* durchgeführt werden. Folgendes Schaubild zeigt die Vorgehensweise bei einer solchen Migration.

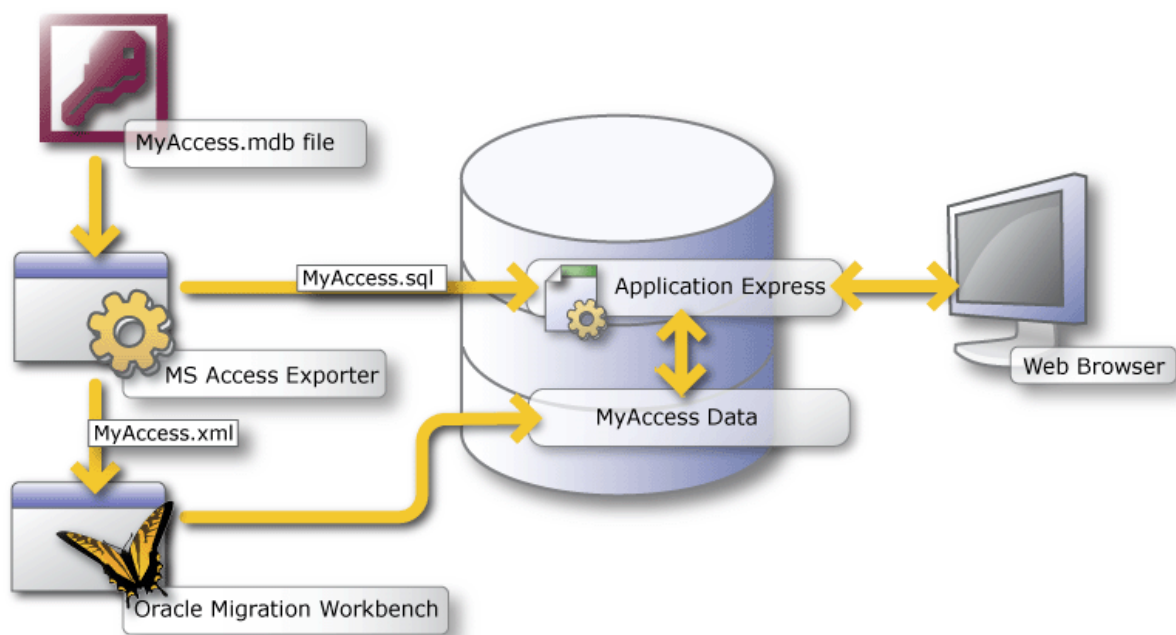


Abb. 1: Funktionsweise einer Migration mittels SQL*Developer und APEX

Dieses Schaubild zeigt, dass eine Migration mittels SQL*Developer und APEX in sieben Schritte eingeteilt werden kann. Diese sind:

1. Exportieren der Metadaten der MS Access Datenbank und Anwendung
2. Migrieren der MS Access Datenbank in eine Oracle Datenbank
3. Erstellen eines APEX-Workspaces
4. Erstellen eines Migrationsprojektes
5. Analysieren der MS Access Anwendung
6. Generieren einer APEX-Anwendung
7. Anpassen der unter Punkt 6 generierten APEX-Anwendung

Im Folgenden werden diese Schritte näher erläutert und die für diese Vorgehensweise notwendigen Vorarbeiten aufgezeigt.

Vorarbeiten

Um eine Migration mittels des SQL*Developers durchführen zu können, muss ein sogenanntes Migrationsrepository vorhanden sein. Dieses beinhaltet Tabellen, Views, Indizes und PL/SQL-Code.

Dieses Migrationsrepository kann in der gleichen Datenbank angelegt werden, in die auch die MS Access Datenbank migriert werden soll. Oracle empfiehlt für dieses Repository ein separates Schema anzulegen.

In der MS Access Datenbank muss sichergestellt werden, dass der Benutzer admin zumindest Leserechte auf das Design und die Daten der Objekte MSysObjects, MSysQueries und MSysRelationships besitzt.



Abb. 2: MS Access Rechtevergabe

Falls in der MS Access Datenbank das Security-Feature eingeschaltet ist, muss dieses ausgeschaltet werden. Dazu ist eine neue MS Access Datenbank - ohne Security - anzulegen, in welche die Tabellen der Quelldatenbank importiert werden müssen. Die zur Zeit vorliegende Version des SQL*Developers unterstützt keine Migration von gesicherten MS Access Datenbanken. Sollten in der MS Access Datenbank verknüpfte Tabellen vorhanden sein, dann müssen vor der Migration die Verknüpfungen aktualisiert werden.

In unserem Haus führte die Verwendung von verknüpften Tabellen dann zu Problemen, wenn folgende Situation eintrat.

- Eine MS Access Datenbank beinhaltet das Frontend (die Anwendung). Darin enthalten sind ebenfalls verknüpfte Tabellen aus einer anderen MS Access Datenbank (Backend). Diese Backend-Tabellen wiederum sind weiter in ein DB2 unter z/OS verknüpft. Während der Migration der MS Access Datenbank wurde für manche dieser DB2-Tabellen keine DDL generiert bzw. es fehlten Tabellenspalten unterschiedlicher Datentypen in der DDL. In diesen Fällen haben wir die DB2-DDL manuell für Oracle angepasst und danach installiert. Die Daten aus DB2 wurden mittels eines PERL-Programms nach Oracle kopiert. Die MS Access Anwendung wurde in APEX komplett neu entwickelt.

Für die Migration wird außerdem der Oracle ODBC-Treiber sowie eine MS Access Volllizenz benötigt. Sind alle diese Vorarbeiten geleistet, kann mit der eigentlichen Migration begonnen werden.

Exportieren der Metadaten

Die Metadaten einer MS Access Datenbank werden mittels des *Exporters für MS Access* erfasst. Dieses Tool wird mit dem SQL*Developer ausgeliefert, und ist eine MS Access Anwendung, die als MDE-Datei - also geschützt und kompiliert - vorliegt. Für jede MS Access Version existiert eine spezifische Exporter-Version. Die MS Access Exporter sind als Zip-Dateien ebenfalls im Standardlieferungsumfang von APEX enthalten (Verzeichnis: \$APEX_HOME/images/migration).

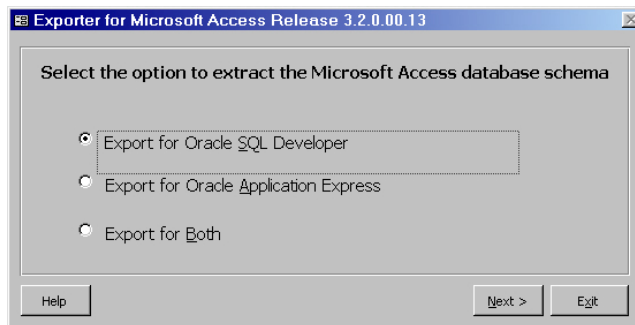


Abb. 3: MS Access Exporter

Hier wird nun angegeben, ob es sich um eine reine Datenbank-Migration (*Export for Oracle SQL Developer*), eine reine APEX-Portierung (*Export for Oracle Application Express*) oder beides handelt. Die Funktionalität *Export for Oracle Application Express* erstellt dabei eine SQL-Datei, die das Einlesen der MS Access Formulare und Berichte in Oracle Application Express ermöglicht.

Die entsprechende Dokumentation des SQL*Developers kann so verstanden werden, dass eine Migration einer MS Access Datenbank sowohl Online, also Wizard gesteuert, als auch Offline, also über das Einlesen einer mit dem Exporter generierten DDL-Datei, erfolgen kann. Dem ist allerdings nicht so. Obwohl bei einer Online-Migration MS Access als verfügbare Quelle angezeigt wird, ist es nicht möglich im Auswahldialog der Quelldatenbank eine MS Access Datenbank anzugeben.

Im nächsten Schritt können nun die zu erfassenden MS Access Datenbanken angegeben werden.

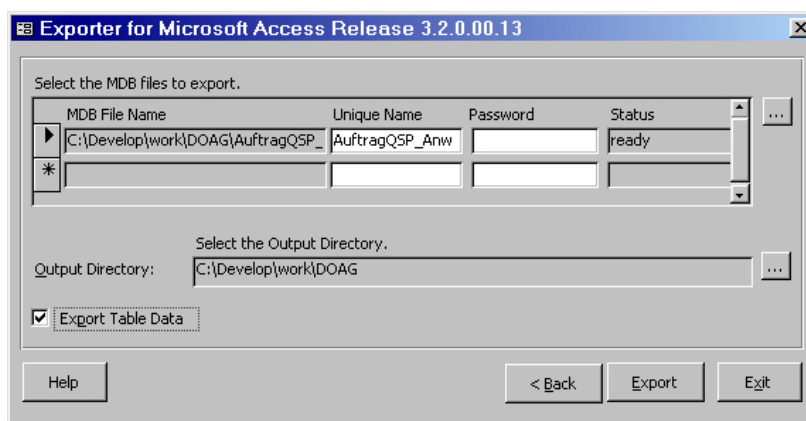


Abb. 4: MS Access Exporter Auswahldialog

Durch einen Klick auf den Export-Button wird dann die Erfassung der MS Access Datenbank gestartet. Nach einer erfolgreichen Erfassung kann der Exporter über den Exit-Button geschlossen werden. Der Exporter erzeugt eine XML-Datei mit den Metadaten der MS Access Datenbank, die im nächsten Schritt für die Migration nach Oracle benötigt wird.

Migrieren einer MS Access Datenbank in eine Oracle Datenbank

Nach der Erfassung kann die erzeugte XML-Datei für eine Migration herangezogen werden. Hierbei ist zu beachten, dass es sich um eine Offline-Migration handeln muss, damit die entsprechende XML-Datei überhaupt angegeben werden kann.

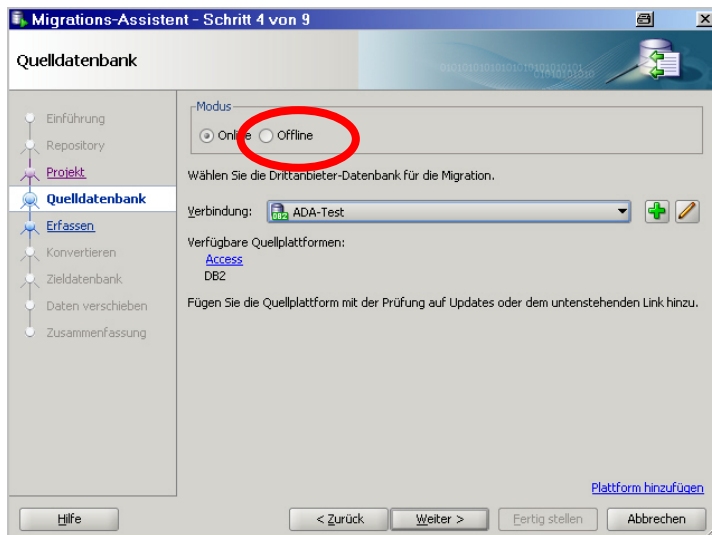


Abb. 5: Migrationswizard des SQL*Developers

Nach Abschluß der Migrationsaktivitäten wird eine SQL-Datei erzeugt, die Befehle für das Anlegen der entsprechenden Datenbankbenutzer sowie deren Objekte beinhaltet.

```
SQL-Arbeitsblatt | Historie
Arbeitsblatt | Query Builder
1  SET DEFINE OFF;
2  PROMPT Creating User Emulation ...
3  CREATE USER Emulation IDENTIFIED BY Emulation DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE TE
4  GRANT CREATE SESSION, RESOURCE, CREATE VIEW, CREATE MATERIALIZED VIEW, CREATE SYNONYM, CREATE
5  SET SCAN OFF;
6  PROMPT Creating User AuftragQSP_Anw ...
7  CREATE USER AuftragQSP_Anw IDENTIFIED BY AuftragQSP_Anw DEFAULT TABLESPACE USERS TEMPORARY TAB
8  GRANT CREATE SESSION, RESOURCE, CREATE VIEW, CREATE MATERIALIZED VIEW, CREATE SYNONYM TO Auftr
9  SET SCAN OFF;
10 PROMPT Creating User AuftragQSP_Data ...
11 CREATE USER AuftragQSP_Data IDENTIFIED BY AuftragQSP_Data DEFAULT TABLESPACE USERS TEMPORARY T
12 GRANT CREATE SESSION, RESOURCE, CREATE VIEW, CREATE MATERIALIZED VIEW, CREATE SYNONYM TO Auftr
13 connect Emulation/Emulation;
14
15 CREATE OR REPLACE PACKAGE msaccess_utilities AS
16 identity NUMBER(10);
17 truncount NUMBER(10):=0;
18 var_number NUMBER(10):=0;
19
20 FUNCTION left_(p_string VARCHAR2, p_length NUMBER) RETURN VARCHAR2;
21 FUNCTION val(p_expr VARCHAR2) RETURN VARCHAR2;
22 FUNCTION sqr(p_num NUMBER) RETURN NUMBER;
23 FUNCTION oct(p_num VARCHAR2) RETURN VARCHAR2;
24 FUNCTION str(p_num VARCHAR2) RETURN VARCHAR2;
25 FUNCTION strcomp(p_first_string VARCHAR2, p_second_string VARCHAR2,
26 FUNCTION pi RETURN NUMBER;
27 FUNCTION format_numeric_expr(p_expr VARCHAR2, p_percent BOOLEAN);
28 FUNCTION hour_(p_date DATE) RETURN NUMBER;
29 FUNCTION isempty(p_expr VARCHAR2) RETURN NUMBER;
30 FUNCTION fix(p_num NUMBER) RETURN NUMBER;
31 FUNCTION substr_from_end(p_string VARCHAR2, p_length NUMBER, p_end NUMBER DEFAULT 1) RETURN VA
32 FUNCTION ismissing(p_expr VARCHAR2) RETURN NUMBER;
33 FUNCTION minute_(p_date DATE) RETURN NUMBER;
34 FUNCTION asc_(p_num VARCHAR2) RETURN NUMBER;
35 FUNCTION isdate(p_expr VARCHAR2) RETURN NUMBER;
36 FUNCTION hex(p_num VARCHAR2) RETURN VARCHAR2;
37 FUNCTION atn(p_num NUMBER) RETURN NUMBER;
38 FUNCTION midb(p_string VARCHAR2, p_start NUMBER, p_length NUMBER) RETURN VARCHAR2;
```

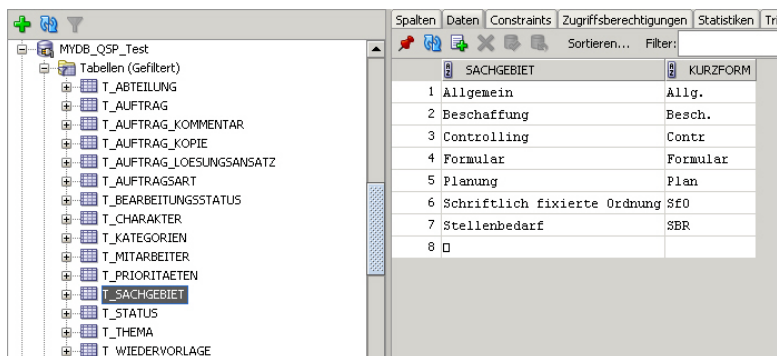
Abb. 6: Mit SQL*Developer generierte DDL

Diese Datei können Sie im SQL*Developer bearbeiten und dann ausführen. Für die Ausführung in unserem Haus wurden folgende Befehlsfolgen angepasst:

1. Die CREATE USER Befehle wurden bzgl. des Kennwortes und des DEFAULT TABLESPACES angepasst.
2. Bei der Analyse der MS Access Datenbank wurde festgestellt, dass in den Abfragen nur Standard-SQL verwendet wird. Daher haben wir den CREATE USER emulation und das CREATE PACKAGE msaccess_utilities, sowie sämtliche Referenzen darauf, entfernt. In diesem Package werden MS Access Funktionen nachgebildet.
3. Handelt es sich um eine MS Access Datenbank mit verknüpften Tabellen, wird für jede MS Access Datenbank ein eigenes Schema angelegt. Für die darin befindlichen Tabellen werden die Rechte SELECT, INSERT, UPDATE, DELETE und REFERENCES an PUBLIC granted. Befinden sich des weiteren in der MS Access Quelldatenbank Abfragen auf verknüpfte Tabellen, so wird in der DDL-Datei ein CREATE VIEW Befehl für jede Abfrage generiert. Hierbei wird aber leider der Owner der Basistabellen vergessen. Durch das Konsolidieren der Tabellen und Views in ein Schema haben wir diese Unschönheiten umgangen.
4. Module aus der MS Access Quelldatenbank werden als CREATE PROCEDURE Befehle in die DDL-Datei generiert. Die Prozeduren bestehen dabei aus dem Code des MS Access Moduls als Kommentar und beinhalten ansonsten keinen Code. Das ist als Merker, dass hier ggf. etwas zu tun ist, nicht schlecht. Wir haben bei der Analyse der MS Access Module allerdings festgestellt, dass der Code durch APEX-Funktionalitäten (Authentifizierungsschema, Wertelisten, usw.) abgelöst werden kann. Daher haben wir auch diese Prozedurrümpfe entfernt.

Nach Ausführen der angepassten DDL-Datei ist das MS Access Schema nach Oracle migriert. Nun kann im SQL*Developer eine Verbindung zu diesem Schema in einer Oracle Datenbank hergestellt werden.

Für das Verschieben der Daten von MS Access nach Oracle wurden vom Migrationsassistenten Loader-Controllfiles, Datendateien und ein Befehlsskript generiert. Bei den in unserem Hause durchgeführten Migrationen entstanden hier Probleme mit Datumsspalten und Spalten, die Zahlen mit Nachkommastellen beinhalten. In diesen Fällen müssen ggf. die Loader-Controllfiles bzgl. der Datums- und Zahlenformate angepasst werden. Leider ließ sich nicht nachvollziehen, wann und warum diese Probleme auftreten. In machen Fällen wurden so geartete Felder ohne Probleme verschoben, in manchen Fällen nicht. Nach den ggf. durchzuführenden Anpassungen und dem Lauf des Befehlsskriptes sind dann auch die Daten in Oracle vorhanden.



SACHGEBIET	KURZFORM
1 Allgemein	Allg.
2 Beschaffung	Besch.
3 Controlling	Contr
4 Formular	Formular
5 Planung	Plan
6 Schriftlich fixierte Ordnung	SFO
7 Stellenbedarf	SBR
8	

Abb. 7: Nach Oracle verschobene Daten

APEX-Tätigkeiten

Für die Migration der MS Access Anwendung nach APEX muss ein Workspace vorhanden sein. In diesem Workspace wird dann ein Migrationsprojekt erstellt. Beim Erstellen dieses Migrationsprojektes muss dann die SQL-Datei angegeben werden, die der MS Access Exporter für die Anwendung erstellt hat.

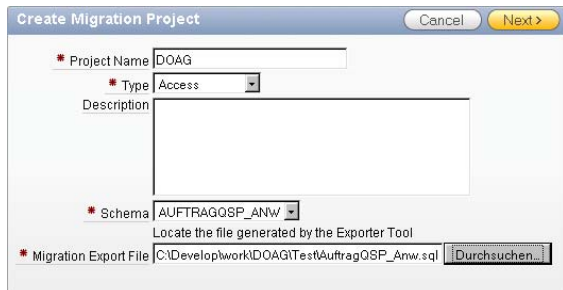


Abb. 8: Migrationsprojekt erstellen

Nach dem Import dieser Datei stehen die aus der Anwendung eingelesenen Objekte im Migrationsprojekt zur Verfügung. Diese müssen nun geprüft und für das Generieren einer Anwendung markiert werden.

Objects	Count	Valid	Invalid	Included	Database
Tables	0	0	-15	0	AuftragQSP_Data
Tables	15	15	0	15	AuftragQSP_Data
Queries	4	4	0	4	AuftragQSP_Data
Queries	0	0	0	0	AuftragQSP_Data
Forms	7	2	5	2	AuftragQSP_Data
Forms	0	2	5	0	AuftragQSP_Data
Reports	2	2	0	2	AuftragQSP_Data
Reports	0	0	0	0	AuftragQSP_Data
Modules	2	0	0	0	AuftragQSP_Data
Modules	0	0	0	0	AuftragQSP_Data
Pages	0	0	0	0	AuftragQSP_Data
Pages	0	0	0	0	AuftragQSP_Data

Abb. 9: Eingelesene Objekte

Durch das Zusammenlegen der MS Access Datenbanken in ein Schema entstanden hierbei für unsere Migrationen ungewollte „Seiteneffekte“. Diese zeigten sich dahingehend, dass alle Objekte mehrfach erschienen und zwar pro MS Access Datenbank einmal. Bei der Prüfung der Objekte können ebenfalls Probleme auftauchen, wenn selbst geschriebene VB-Funktionen als Bestandteil der SQL-Befehle verwendet werden, die für den Aufbau von Berichten oder Formularen in der MS Access Anwendung benutzt werden. Da diese VB-Funktionen nicht migriert werden, erscheinen diese Objekte als „Ungültig“. Hier sind dann die ungültigen VB-Funktionen zu prüfen und vor der Generierung der Anwendung durch PL/SQL-Code zu ersetzen.

Sind alle Objekte geprüft und gültig, können sie für das Generieren markiert werden. Generiert werden können:

- Die Formulare und Berichte, so wie sie aus der MS Access Anwendung erfasst wurden.
- Eine Wartungsanwendung, die aus Berichten und Formularen auf die migrierten Tabellen besteht.

Danach entstehen Anwendungen, die als Grundlage für das Weiterentwickeln - Stichwort VB-Code - bzw. das „Anhübschen“ verwendet werden können.

Fazit

Eine Migration einer MS Access Anwendung in eine Oracle Umgebung mit APEX ist je nach Umfang der MS Access Anwendung mit mehr oder weniger Aufwand machbar. Die Kosten halten sich im Rahmen, da die für die Migration verwendete Software Bestandteil der Datenbank-Lizenz und somit kostenfrei zu erhalten ist.

Für unser Haus haben wir festgestellt, dass eine 1:1 Migration (migrieren, sieht gut aus und tut) bei den bei uns verwendeten MS Access Anwendungen nicht zu erreichen ist. Vielmehr haben alle unsere Migrationen ergeben, dass gewisse Nacharbeiten erforderlich sind. Daher wird von unserem Haus der Weg der Migration der Daten von MS Access nach Oracle und das „Neumachen“ der Anwendung bevorzugt, auch wenn das manchmal etwas aufwändiger sein kann.

Kontaktadresse:**Markus Kühn**

Landesbausparkasse Baden-Württemberg
Jägerstrasse 36
D-70174 Stuttgart

Telefon: +49 (0) 711-183 2915
Fax: +49 (0) 711-183 49 2915
E-Mail: markus.kuehn@lbs-bw.de
Internet: www.lbs-bw.de