

HERZLICH WILLKOMMEN

Migration von Oracle 9i nach 11g

DOAG – SIG Database 12.09.2013 in Berlin

Axel vom Stein

axel.vomstein@bss-bohnenberg.de

Hauptsitz: Solingen

weitere Standorte: Ilmenau (Deutschland), Oedelem (Belgien),
Kattowitz (Polen), Wels (Österreich)

Geschäftsführer: Harald Bohnenberg, Martin Gräb, Vasilios Dossis

Gegründet: 1991

seit 2006: Unternehmen der ROFA – Gruppe

Rechtsform: GmbH

Jahresumsatz: ca. 40 Mio. EUR

Mitarbeiteranzahl: ca. 140

davon Ingenieure: ca. 90

Realisierte Projekte: ca. 600



- **Entwurf und Feinplanung von intralogistischen Systemen und Anlagen**
- **schlüsselfertiger Abwicklung als GU**
- **Flowpicker (Hochleistungskommissionierung)**
- **Hochregal und Kleinteilelagern**
- **Lagerverwaltungssystemen (BSS-L)**
- **Materialflussrechnern (BSS-M)**
- **fahrerlosen Transportsystemen**
- **Pick-by-Light / Pick-by-Voice**
- **Staplerleitsystemen**
- **Datenfunk über WLAN**
- **SPS – Technik (BSS-S)**
- **Visualisierungen**
- **etc.**



- **Technischer Projektleiter (bei BSS seit 2001)**

- **Oracle:**
 - **seit 2001**
 - **zunächst als Entwickler (PL/SQL, C#, Delphi)**
 - **dann mehr und mehr Richtung Datenbankadministration**

- **Schwerpunkte:**
 - **Oracle auf Windows – der Mittelstand hat i.d.R. kein Linux**
 - **Oracle Standard Edition**
 - **Sicherheitskonzepte (Unmanaged Standby, Dataguard, Failsafe)**
 - **Migrationen**
 - **Standardisierung der BSS – Software**
 - **alles was sich automatisieren lässt**

- 1.) Vorüberlegungen**
- 2.) Durchführung**
- 3.) Zusammenfassung**

1.) Vorüberlegungen

Warum – Wie

Herausforderungen

Probleme

2.) Durchführung

3.) Zusammenfassung

Warum migrieren:

- **Hardware und/oder Betriebssystemtausch**
- **Sicherheitskonzept erweitern**
- **große Systemerweiterung auf Applikationsebene**
- **ein neues Oracle-Release**

(in den seltensten Fällen ist dies bei KMU der treibende Faktor, es werden i.d.R. ein und manchmal auch zwei Releases übersprungen)

Wie migrieren:

- **eine „frische“ Datenbank auf einem neuen Rechner**
- **ein neues Oracle-Home mit einer „frischen“ Datenbank auf bisherigem Rechner**

→ Vorteil Fallback ist relativ einfach

Immer:

- **Downtime so gering wie möglich halten**
- **störungsfreier Betrieb nach erfolgter Migration**
- **Fallback berücksichtigen**

Oftmals:

- **Im Rahmen der Migration das System optimieren/erweitern**
- **Nur die User/Schemata migrieren, die wirklich noch gebraucht werden**
- **Passwörter mit übernehmen (aber nicht für alle User)**
- **Kunde möchte möglichst wenig Unterstützung liefern/leisten**
- **Testsystem migrieren, falls vorhanden**
- **Testsystem aufbauen, falls nicht vorhanden**
- **Frontend ebenfalls optimieren/erweitern/erneuern**

Daran wird man wohl denken:

- **Optimizer – ein Klassiker**
 - z.B. **Select-Statements („group by“ ohne „order by“)**
init.ora Parameter: `_gby_hash_aggregation_enabled=false`
- **Optimizer – Allgemein (z.B. bei einer Quell – DB 9.2.0.8):**
init.ora Parameter: `optimizer_features_enable=9.2.0.8`
- **Ob das sinnvoll ist, hängt von vielen Faktoren ab ...**
... und ist einen eigenen Vortrag wert ...
- **Welche Werte sind erlaubt?**
http://docs.oracle.com/cd/E18283_01/server.112/e17110/initparams164.htm
- **Mehr Input über die erlaubten Werte (Database Performance Tuning Guide):**
http://docs.oracle.com/cd/E18283_01/server.112/e16638/toc.htm

Daran wird man wohl denken:

- neue/geänderte Rollen z.B. CONNECT in 9i

CREATE VIEW

CREATE SESSION

CREATE TABLE

CREATE SYNONYM

ALTER SESSION

CREATE SEQUENCE

CREATE CLUSTER

CREATE DATABASE LINK

CONNECT ab 10g **nur noch** CREATE SESSION

➔ besser eine eigene Rolle verwenden

- ab 11g kann das Passwort CASE Sensitive sein (was will der Kunde ...)
- Schemata übergreifende Berechtigungen/FK's etc.

Aber denkt man auch an diese Dinge?

- **Externe Tabellen werden nicht mit exportiert**
- **Objekte in Oracle eigenen Schemata (SYS,SYSTEM etc. – ob Absicht oder nicht)**
- **bei der Verwendung von Libraries und Directories ggf. Pfade anpassen**
- **Queues für Enque/Dequeue aktivieren**
- **UTL_TCP, UTL_SMTP, UTL_MAIL, UTL_HTTP, oder UTL_INADDR**
→ **wer diese Packages nutzt, muss bei 11g die XML-DB installieren**
- **11g: Network ACL's werden für die obigen Packages benötigt**

... und an diese?

```
SQL> show parameter plsql_warnings
plsql_warnings      string      ENABLE:ALL
```

```
SQL> CREATE OR REPLACE PROCEDURE Mist AS
2 BEGIN
3   Dbms_Output.Put_Line('So nicht übersetzbar...');
4 EXCEPTION
5   WHEN OTHERS THEN
6     NULL;
7 END;
8 /
```

•SP2-0804: Procedure created with compilation warnings

```
SQL> show errors;
Errors for PROCEDURE MIST:
```

```
PLW-06009: Prozedur „MIST" OTHERS Handler endet nicht in RAISE oder RAISE_APPLICATION_ERROR
```


So geht's:

```
alter system set plsql_warnings='ENABLE:ALL,DISABLE:6009,DISABLE:6010' scope = both;
```

... und besonders an diese ?

- bei einem Export mit 9i und Import in 10g/11g wird der Buffer_Pool von Tabellen/Indices immer als Default angelegt – unabhängig von den vorherigen Einstellungen (Keep, Recycle, Default)
- Default Profil hat sich geändert, Auswirkung z.B. auf DB-Console, merkt man ggf. erst nach 180 Tagen ...

```
SELECT * FROM dba_profiles t WHERE PROFILE = 'DEFAULT'
```



PROFILE	RESOURCE_NAME	RESOURCE_TYPE	LIMIT
DEFAULT	FAILED_LOGIN_ATTEMPTS	PASSWORD	10
DEFAULT	PASSWORD_LIFE_TIME	PASSWORD	180
DEFAULT	PASSWORD_LOCK_TIME	PASSWORD	1
DEFAULT	PASSWORD_GRACE_TIME	PASSWORD	7

Full	Schema
<ul style="list-style-type: none">+ keine/wenige vorbereitende Skripte+ Gefahr etwas zu vergessen gering- ungutes Gefühl, da auch Objekte aus dem Schema SYSTEM importiert, bzw. Views invalid wurden, die nicht durch Recompile valid wurden- kann nicht alles Full ist nicht wirklich Full	<ul style="list-style-type: none">- viele vorbereitende Skripte- Gefahr etwas zu vergessen hoch+ kontrollierter Ablauf, nur das was man auch auswählt / implementiert+ kann alles, wenn man es implementiert

1.) Vorüberlegungen

2.) Durchführung

Ausgangslage

Idee

Ergebnis

Umsetzung

Sonstiges

3.) Zusammenfassung

Fazit

Quellen

Besonderheit einer BSS Datenbank:

- **Datenbanken sind i.d.R. nicht riesig (ca. 80 – 150 GB)**
- **die gesamte Logik des Materialflusses und Lagerverwaltungssystem liegt innerhalb der Datenbank**
- **die Logik ist in unterschiedlichste Schemata aufgeteilt (d.h. diverse Tabellen mit diversen FK's)**
- **Kopplungen zu den unterschiedlichsten Systemen wie SAP, proAlpha, SPS'en, Pick-By-Light, Pick-By-Voice, Etikettierern, usw. sind auch innerhalb der Datenbank**
- **alle Prozesse werden durch Oracle – Jobs (dbms_job) abgebildet, die über Advanced Queuing miteinander kommunizieren**

Erste Idee:

Erstelle Skripte, die aus einer bestehender Datenbank alle Informationen sammeln, die für einen Schema – Import in eine neue Datenbank benötigt werden.

Zweite Idee:

Alle Skripte, die irgendwo auf dem Filesystem lagen durch ein PL/SQL – Package (ExportUtl.pck) ersetzen, das möglichst universell zu nutzen ist. PL/SQL bietet mehr Möglichkeiten der Kapselung und Wiederverwendbarkeit.

Ziel:

Das Package soll Skripte erzeugen, die auf dem Zielsystem angewandt werden können und somit möglichst keine manuellen Eingriff erfordern.

Hilfsmittel:

- **Package utl_file**
- **Oracle Views**
- **interne Oracle Tabellen (z.B. User\$, etc.)**
- **Bedingte Kompilierung für Release unabhängige Verwendung**
(ab 10.2 supported, nutzbar ab 9.2.0.6 mit hidden init.ora Parameter)

Mit ExportUtl.pck erzeugte Dateien:

<input type="checkbox"/>	01_01_Pre_DOAGPROD_TbsOverviewForImport	txt	6,8 k
<input type="checkbox"/>	02_00_Pre_DOAGPROD_Run	sql	3,5 k
<input type="checkbox"/>	02_01_Pre_DOAGPROD_CreateAllBssProfiles	tmp	4,6 k
<input type="checkbox"/>	02_02_Pre_DOAGPROD_CreateAllBssRolesWithSysRigh..	tmp	11,1 k
<input type="checkbox"/>	02_03_Pre_DOAGPROD_CreateAllBssDirectories	tmp	1,0 k
<input type="checkbox"/>	02_04_Pre_DOAGPROD_CreateAllBssDatabaseLinks	tmp	392 b
<input type="checkbox"/>	02_05_Pre_DOAGPROD_CreateAllBssUserWithSysRigh..	tmp	205,7 k
<input type="checkbox"/>	02_06_Pre_DOAGPROD_CreateAllBssLibraries	tmp	544 b
<input type="checkbox"/>	02_07_Pre_DOAGPROD_CreateAllBssSynonyms	tmp	1,5 k
<input type="checkbox"/>	02_08_Pre_DOAGPROD_CreateBssSysView	tmp	735 b
<input type="checkbox"/>	02_09_Pre_DOAGPROD_GrantObjectRightsOnSysAndS..	tmp	48,4 k
<input type="checkbox"/>	02_10_Pre_DOAGPROD_GrantAllSysAndSysoperUser	tmp	2,4 k
<input type="checkbox"/>	02_11_Pre_DOAGPROD_SwitchOffOraParameter	tmp	524 b

Vorbereitung

<input checked="" type="checkbox"/>	03_01_Imp_DOAGPROD_Run	cmd	1,5 k
<input type="checkbox"/>	03_02_Imp_DOAGPROD_Parameter	imp	6,1 k

Import

<input type="checkbox"/>	04_00_Post_DOAGPROD_Run	sql	2,3 k
<input type="checkbox"/>	04_01_Post_DOAGPROD_RecompileAll	tmp	718 b
<input type="checkbox"/>	04_02_Post_DOAGPROD_ActivateAllQueues	tmp	19,4 k
<input type="checkbox"/>	04_03_Post_DOAGPROD_SwitchOffBssJobs	tmp	13,0 k
<input type="checkbox"/>	04_04_Post_DOAGPROD_SwitchOnOraParameter	tmp	523 b
<input type="checkbox"/>	04_05_Post_DOAGPROD_BufferCachePoolAssignment	tmp	49,7 k

Nachbereitung

Exemplarischer Inhalt der erzeugten Dateien:

TBS-Name	Typ	File-Name	Extent-Management	Allocation_Typ	Belegt-MB	Max-MB	Uniform-Size
DATA-LARGE	PERMANENT	F:\ORACLE\ORADATA\DOAGPROD\DATA-LARGE-01.DBF	LOCAL	UNIFORM	1064	2048	32 M
	PERMANENT	F:\ORACLE\ORADATA\DOAGPROD\DATA-LARGE-02.DBF	LOCAL	UNIFORM	96	2048	32 M
DATA-MEDIUM	PERMANENT	F:\ORACLE\ORADATA\DOAGPROD\DATA-MEDIUM-01.DBF	LOCAL	UNIFORM	768	2048	1 M
DATA-SMALL	PERMANENT	F:\ORACLE\ORADATA\DOAGPROD\DATA-SMALL-01.DBF	LOCAL	UNIFORM	95	2048	64 K
INDX-LARGE	PERMANENT	G:\ORACLE\ORADATA\DOAGPROD\INDX-LARGE-01.DBF	LOCAL	UNIFORM	1956	2048	8 M
	PERMANENT	G:\ORACLE\ORADATA\DOAGPROD\INDX-LARGE-02.DBF	LOCAL	UNIFORM	136	2048	8 M
INDX-SMALL	PERMANENT	G:\ORACLE\ORADATA\DOAGPROD\INDX-SMALL-01.DBF	LOCAL	UNIFORM	8	2048	64 K
INIT	PERMANENT	G:\ORACLE\ORADATA\DOAGPROD\INIT-01.DBF	LOCAL	UNIFORM	9	2048	64 K
LOGBUCH-LARGE	PERMANENT	G:\ORACLE\ORADATA\DOAGPROD\LOGBUCH-LARGE-01.DBF	LOCAL	UNIFORM	768	2048	64 M

```
prompt Info: CREATE USER EXPORT
CREATE USER          "EXPORT"
PROFILE              "BSS_BA"
IDENTIFIED BY VALUES "SYSB19227E127864"
DEFAULT TABLESPACE "DATA-MEDIUM"
TEMPORARY TABLESPACE "TEMP-ABL"
ACCOUNT UNLIMITED;

prompt Info: GRANT BSS_CONNECT TO USER EXPORT
GRANT "BSS_CONNECT" TO "EXPORT";

prompt Info: GRANT EXP_FULL_DATABASE TO USER EXPORT
GRANT "EXP_FULL_DATABASE" TO "EXPORT";
```

```
prompt Info: GRANT SELECT ON SYS.DBA_DATA_FILES TO EXPORT
GRANT SELECT ON "SYS"."DBA_DATA_FILES" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_DIRECTORIES TO EXPORT
GRANT SELECT ON "SYS"."DBA_DIRECTORIES" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_FREE_SPACE TO EXPORT
GRANT SELECT ON "SYS"."DBA_FREE_SPACE" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_INDEXES TO EXPORT
GRANT SELECT ON "SYS"."DBA_INDEXES" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_LIBRARIES TO EXPORT
GRANT SELECT ON "SYS"."DBA_LIBRARIES" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_OBJECTS TO EXPORT
GRANT SELECT ON "SYS"."DBA_OBJECTS" TO "EXPORT";

prompt Info: GRANT SELECT ON SYS.DBA_PROFILES TO EXPORT
GRANT SELECT ON "SYS"."DBA_PROFILES" TO "EXPORT";

....
```

spool .\Log_02_Pre_DOAGPROD_All.log

Prompt ##### Run Script 02_01_Pre_DOAGPROD_CreateAllBssProfiles.tmp

Prompt

@02_01_Pre_DOAGPROD_CreateAllBssProfiles.tmp

Prompt ##### Run Script 02_02_Pre_DOAGPROD_CreateAllBssRolesWithSysRights.tmp

Prompt

@02_02_Pre_DOAGPROD_CreateAllBssRolesWithSysRights.tmp

...

Exemplarischer Inhalt der erzeugten Dateien:

```
prompt Info: CREATE ACL Interface-PickByLight.xml with Principal BSS_PKG
```

```
BEGIN
```

```
  Sys.Dbms_Network_Acl_Admin.Create_Acl  
  (  
    Acl          => 'Interface-PickByLight.xml',  
    Description => 'TCP/IP Schnittstelle SpeedyPick II',  
    Principal    => 'BSS_PKG',  
    Is_Grant     => true,  
    Privilege    => 'connect'  
  );
```

```
END;
```

```
/
```

```
prompt Info: ASSIGN ACL Interface-PickByLight.xml for Host(10.110.4.141) Lower-Port (3001) Upper-Port(3001)
```

```
BEGIN
```

```
  Sys.Dbms_Network_Acl_Admin.Assign_Acl  
  (  
    Acl          => 'Interface-PickByLight.xml',  
    Host         => '10.110.4.141',  
    Lower_Port   => 3001,  
    Upper_Port   => 3001  
  );
```

```
END;
```

```
/
```

**....CreateNetworkAcl...
(Quelle 11g: aus DB ermittelbar)**

Was beim Aufruf der Skripte passiert, ist anhand der Namen meist selbsterklärend, folgende Hinweise zu ausgewählten Skripten:

...SwitchOffOraParameter:

```
ALTER SYSTEM SET job_queue_processes=0;
```

Notwendig damit beim Import kein Oracle-Job anlaufen kann.

... ActivateAllQueues:

```
DBMS_AqAdm.Start_Queue('KND_QUE.MFR_PFT_QUEUE', TRUE, TRUE);
```

Einschalten von Enqueue / Dequeue einer Queue.

...SwitchOffBssJobs:

Broken – Flag für Jobs setzen.

...CreateBssSysView:

```
SELECT t.Indx          AS FileIndex,  
       t.Fname_Krbmsft AS FileName  
FROM Sys.x$krbmsft t
```

Mit der Prozedur `Sys.Dbms_Backup_Restore.SearchFiles` und obiger View kann der Inhalt eines Verzeichnisses ausgelesen werden. (Feature ist nicht dokumentiert, wird von RMAN genutzt (`CATALOG START WITH ...`), um zu prüfen, ob es sich um Oracle – Files handelt).

...BufferCachePoolAssignment:

Setzen der richtigen Cache Zuordnungen (Recycle, Keep, Default) nur bei 9i Export notwendig.

Die Datenbank ist sehr gesprächig:

Db_Data_Files, Db_Free_Space, Db_Tablespaces,

Db_Temp_Files, v\$Temp_Extent_Pool

Db_Profiles, Db_Synonyms,

Db_Queues, Db_Objects, Db_Libraries, Db_Directories, Link\$,

User\$, Db_Users, Db_TS_Quotas,

Db_Role_Privs, Db_Tab_Privs, Db_Sys_Privs,

Db_Tables, Db_Indexes, Db_External_Tables,

v\$Database, v\$PwFile_Users, v\$Instance, v\$Parameter,

11g: Db_Network_Acls, Db_Network_Acl_Privileges

Wie ermittelt man alle User / Rollen, die zur Applikation gehören?

```
select user#, name, type#, ctime from user$
```

	USER#	NAME	TYPE#	CTIME
1	3927	_NEXT_USER	0	14.12.2010 14:19:51
2	1	PUBLIC	0	14.12.2010 14:19:52
3	2	CONNECT	0	14.12.2010 14:19:52
4	3	RESOURCE	0	14.12.2010 14:19:52
5	4	DBA	0	14.12.2010 14:19:52
6	0	SYS	1	14.12.2010 14:19:52
7	5	SYSTEM	1	14.12.2010 14:19:52
8	9	EXP_FULL_DATABASE	0	14.12.2010 14:19:58
9	10	IMP_FULL_DATABASE	0	14.12.2010 14:19:58
10	6	SELECT_CATALOG_ROLE	0	14.12.2010 14:19:58
11	31	PERFSTAT	1	14.12.2010 14:44:25
12	37	SYSMAN	1	06.01.2011 15:14:02
13	38	MGMT_USER	0	06.01.2011 15:14:07
14	39	MGMT_VIEW	1	06.01.2011 15:15:07
15	32	BSSINSTALL	1	06.01.2011 15:19:00
16	33	__Oracle_Ende__	0	06.01.2011 15:20:00
17	3880	BSS_ABLAUF_ROLE	0	12.10.2012 22:56:03
18	3861	BSS_DBA_ROLE	0	12.10.2012 22:56:03
19	3862	BSS_ENTWICKLER_ROLE	0	12.10.2012 22:56:03
20	3863	BSS_PLATZHALTER_ROLE	0	12.10.2012 22:56:04
21	3880	BSS_USV	1	12.10.2012 22:56:06

Idee:

- Ergänze eine Dummy-Rolle!

__Oracle_Ende__


- Führe ein Update auf die Spalte CTIME durch!
(alles älter als diese Rolle sind für die Migration relevante Dinge ...)

Achtung:

Nachträgliche Installation

**Bei allen Select – Statements (sofern Anlegezeitstempel bekannt)
einschränken nach folgendem Schema (Beispiel für Rollen):**

```
SELECT u.Name,
       u.Ctime,
       Ende.Ctime,
       Ende.Name
FROM Sys.User$ u,
     Sys.User$ Ende
WHERE u.Type# = 0
      AND u.Ctime > Ende.Ctime
```



	NAME	CTIME	CTIME	NAME
1	BSS_ABLAUF_ROLE	12.10.2012 22:56:03	06.01.2011 15:20:00	__Oracle_End__
2	BSS_DBA_ROLE	12.10.2012 22:56:03	06.01.2011 15:20:00	__Oracle_End__
3	BSS_ENTWICKLER_ROLE	12.10.2012 22:56:03	06.01.2011 15:20:00	__Oracle_End__
4	BSS_PLATZHALTER_ROLE	12.10.2012 22:56:04	06.01.2011 15:20:00	__Oracle_End__
5	PROJECT_USER_ROLE	12.10.2012 22:58:51	06.01.2011 15:20:00	__Oracle_End__

Komplexeres Beispiel (Objektberechtigungen pro User auf Sys/System):

```
SELECT s.Privilege, s.Owner, s.Table_Name, s.Grantee, o.Object_Type
FROM Dba_Users db,
     DBA_Objects o,
     (
       SELECT *
       FROM Dba_Tab_Privs
     ) s,
     Sys.User$ u,
     Sys.User$ Ende
WHERE s.Grantee IN ( SELECT d.Username
                   FROM Dba_users d,
                   Sys.User$ u,
                   Sys.User$ Ende
                   WHERE u.User# = d.User_Id
                        AND u.Type# = 1
                        AND u.Ctime > Ende.Ctime
                        AND Ende.Name = '__Oracle_Ende__'
                   )
AND s.Grantor = u.Name
AND s.Grantee = db.Username
AND s.Owner      = o.Owner
AND s.Table_Name = o.Object_Name
-- PACKAGE BODY bereits durch PACKAGE erschlagen, hier ggf. CONSUMER GROUPE, etc. exkludieren
AND o.Object_Type NOT IN ('PACKAGE BODY')
-- diese Queue-Views existieren in der neuen Ziel-DB nicht
AND o.Object_name NOT LIKE 'QT%BUFFER'
AND u.Type# = 1
AND u.Ctime < Ende.Ctime
AND ende.Name = '__Oracle_Ende__'
ORDER BY s.Grantee, s.Owner, s.Table_Name;
```

Verfahren funktioniert für:

- Directories
- User
- Rollen
- Objektberechtigungen
- Libraries
- DB-Links
- TBS – Quota
- Synonyme
- Queues
- Buffer Cache Zuordnungen

Verfahren funktioniert leider nicht für: Profile, Network-ACL's

Passwörter: ab 11g CASE sensitiv möglich

wenn CASE sensitiv:

- init.ora Parameter `sec_case_sensitive_logon=true`
- mit anderem Hash Algorithmus gespeichert
- nicht mehr in `Dbu_Users` zu finden (bis 10g Spalte Password)
- aber in `Sys.User$` in der Spalte `Spare4`

Dbu_Users: Spalte Password_Version

Password_Version	Spalte Password	Spalte Spare4
10G	Gefüllt	NULL
10G 11G	Gefüllt	Gefüllt
11G	NULL	Gefüllt

Bedeutung für die Skripte:

```
CREATE USER          "HUGO"
PROFILE              "ENTWICKLER"
IDENTIFIED BY VALUES '6AE6CB3B9AF65383'
DEFAULT TABLESPACE "DATAMEDIU"
TEMPORARY TABLESPACE "TEMPBSS"
ACCOUNT UNLOCK;

CREATE USER          "HUGO"
PROFILE              "ENTWICKLER"
IDENTIFIED BY VALUES 'S:28692361521F501C5DA569D78C83D5637C6819CB0B6BE53196F3E5F95C8D'
DEFAULT TABLESPACE "DATAMEDIU"
TEMPORARY TABLESPACE "TEMPBSS"
ACCOUNT UNLOCK;

CREATE USER          "HUGO"
PROFILE              "ENTWICKLER"
IDENTIFIED BY VALUES '6AE6CB3B9AF65383;S:28692361521F501C5DA569D78C83D5637C6819CB0B6BE53196F3E5F95C8D'
DEFAULT TABLESPACE "DATAMEDIU"
TEMPORARY TABLESPACE "TEMPBSS"
ACCOUNT UNLOCK;
```

Will man CASE sensitiv für jedes Passwort erzwingen:

- alle User mit Password_Version = 10G müssen das Passwort ändern
- Einsatz einer Passwort Verify Funktion, um CASE sensitiv zu erzwingen

Hilfreiche Notes: (Doc ID 1307555.1, 463999.1, 429465.1)

- schneller, kleiner, flexibler, API (DBMS_DATAPUMP)
- Query – Klausel sehr hilfreich, um bestimmte Daten nicht zu importieren (alte Logbücher, für Entwicklungsumgebung Teilmengen an Daten, etc.)
`Query=Hugo.Test_Tbl:"WHERE Sequence < 10000"`
- In den Versionen 10.2.0.1 bis 10.2.0.4 und 11.1.0.6 werden Oracle – Jobs nicht bei einem Schema Import importiert (Oracle Bug 7185544 – Note 875904.1).
- Gut versteckt:
Oracle Database Security Guide 11.2
Kapitel 4 – Creating an Access Control List for External Network Services

Note: You cannot import or export the access control list settings by using the Oracle Database import or export utilities such as Oracle Data Pump.

1.) Vorüberlegungen

2.) Durchführung

3.) Zusammenfassung

Fazit

Quellen

- **ExportUtl.pck ist immer weiter entwickelt worden und seit der ersten Migration sehr erfolgreich im Einsatz.**
- **Die Dummy – Rolle „`__Oracle_Ende__`“ wird auch genutzt, um eine Datenbank kontrolliert zu leeren.**
- **Mit 10g wurde ein Job implementiert, der einmal täglich per Datapump einen Export zieht und in diesem Zuge auch dieses Package nutzt.**
- **Es ist einfach geworden aus einer aktuellen Produktivdatenbank schnell eine Test oder – Entwicklungsdatenbank z.B. für Erweiterungen bzw. Schulungszwecke zu generieren.**

Database Utilities:

11gR2: Table 22-3 Objects Imported in Each Mode

http://docs.oracle.com/cd/E11882_01/server.112/e22490/original_import.htm

10gR2: Table 19-3 Objects Exported and Imported in Each Mode

http://docs.oracle.com/cd/B19306_01/server.102/b14215/exp_imp.htm

9iR2:http://docs.oracle.com/cd/B10501_01/server.920/a96652/ch01.htm#1013411

Bei jedem neuen Release:

New Features Guide 11g Release 2 (11.2)

http://docs.oracle.com/cd/E11882_01/server.112/e41360/toc.htm

New Features Guide 12c Release 1 (12.1)

http://docs.oracle.com/cd/E16655_01/server.121/e17906/chapter1.htm

