ORDIX AG

DBMS_METADATA und DBMS_METADATA_DIFF im Praxiseinsatz

DOAG Konferenz + Ausstellung
18.11.2014 Nürnberg

Philipp Loer
info@ordix.de
www.ordix.de

Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

Projekthintergrund (I)

ORDIX AG

- Akquisition eines Kunden durch ein weitaus größeres Unternehmen
- Übernahme großer Teile des IT-Systems des Kleinen durch den Großen

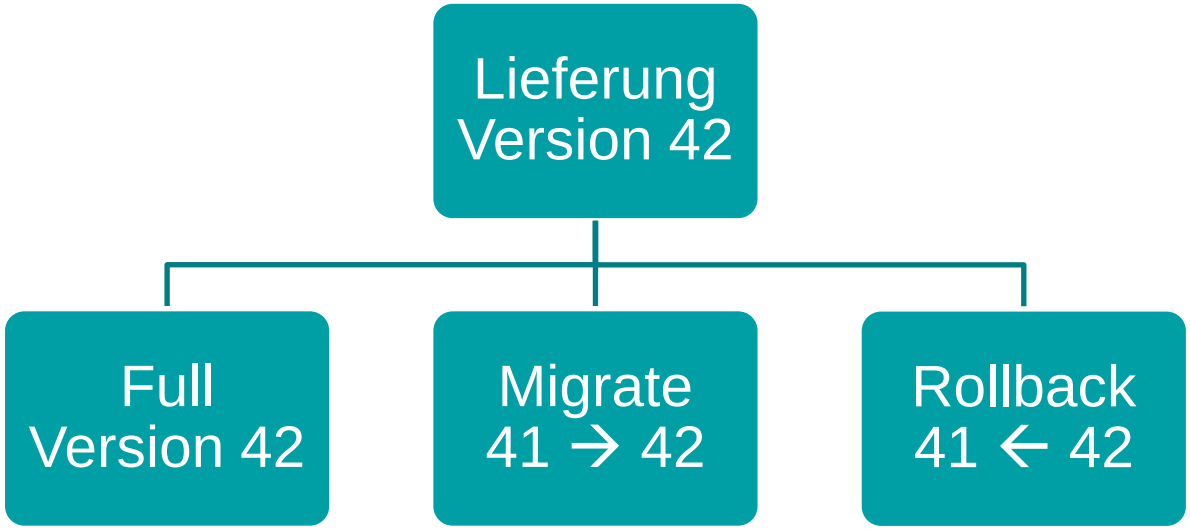
- Softwarelieferungen (auch Datenbanken) in einer fest vorgegebenen Struktur - dem „Vendor Standard“
- Vorgaben des Vendor Standards:
  - Nur SQL (DML, DDL, DCL) und PL/SQL-Skripte erlaubt
  - Jedes Objekt in einer eigenen Datei

- Lieferzyklus: 1x pro Woche
- Aktuell fünf Datenbanken produktiv, weitere in Planung

Projekthintergrund (II)

ORDIX AG

- Jede Auslieferung beinhaltet
  - Eine Full-Lieferung der kompletten Datenbank
  - Eine Migrationslieferung von der letzten Version auf die aktuelle
  - Eine Rollback-Lieferung von der aktuellen Version auf die zuletzt gültige

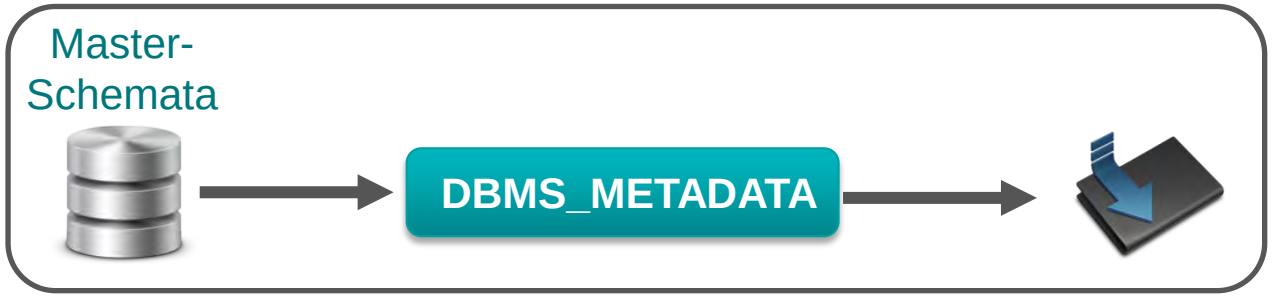


Projekthintergrund (III)

ORDIX AG

- Problem: Aufwand für eine manuelle Erstellung der Skripte
  - Hoher Zeitaufwand
  - Hohes Fehlerrisiko durch Skriptfehler
  - Hohes Fehlerrisiko hinsichtlich Abweichungen zwischen Full- und Migrate-Skripten

- Lösungsansatz:
  - Automatische Generierung aller Skripte mit `DBMS_METADATA`
  - Erstellung der Full-Lieferung aus dem Data Dictionary eines Master-Schemas



Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

DBMS_METADATA

ORDIX AG

- Erlaubt die Generierung von Objekteigenschaften aus dem Data Dictionary
- Generierung als
  - XML
  - DDL-Statement
  - SXML (ab 11g R2)
- Ausgabe kann angepasst werden
  - Schemaname
  - Tablespacename
  - Speicherparameter
  - u.v.m.

GET_DDL-Funktion

ORDIX AG

- Benötigt in der Regel drei Argumente:
  - Objekttypen (Table, View, Index etc.)
  - Objektnamen
  - Schema des Objektes (optional)

```
DBMS_METADATA.GET_DDL (
   object_type IN VARCHAR2,
   name IN VARCHAR2,
   schema IN VARCHAR2 DEFAULT NULL
...
)RETURN CLOB;
```

GET_DDL

Beispiel für Tabellen

ORDIX AG

```
SELECT dbms_metadata.get_ddl('TABLE','DEPT','SCOTT') FROM dual;
```

```
CREATE TABLE "SCOTT"."DEPT"
  (
    "DEPTNO" NUMBER(2,0),
    "DNAME"  VARCHAR2(14),
    "LOC"    VARCHAR2(13),
    CONSTRAINT "PK_DEPT" PRIMARY KEY ("DEPTNO")
    USING INDEX PCTFREE 10 ... TABLESPACE "USERS" ENABLE
  )
  SEGMENT CREATION IMMEDIATE PCTFREE 10 ...
 STORAGE
  (
    INITIAL 65536 NEXT 1048576 ...
  )
  TABLESPACE "USERS"
```

GET_DDL

5 Statements liefern die Metadaten für ein komplettes Schema

ORDIX AG

Um alle Objekte des Schemas zu erhalten:

```
SELECT dbms_metadata.GET_DDL(v.object_type,v.object_name,'SCOTT')
FROM   dba_objects v
WHERE  owner = 'SCOTT';
```

Um alle Grants des Schemas zu erhalten:

```
SELECT dbms_metadata.get_granted_ddl('system_grant','') from dual;
SELECT dbms_metadata.get_granted_ddl('role_grant','') from dual;
SELECT dbms_metadata.get_granted_ddl('object_grant','') from dual;
```

Für die Tablespaces:

```
SELECT dbms_metadata.get_ddl('TABLESPACE', tablespace_name )
from dba_tablespaces;
```

GET_XML

ORDIX AG

- Darstellung des Objektes in XML-Form
- Enthält viele Oracle-interne Parameter
  (ca. 300 bei der Tabelle `SCOTT.DEPT`)
- Ist nur dazu geeignet, das Objekt in einer anderen Oracle-Datenbank bzw. einem anderen Schema erneut zu erstellen
- Die Objekterstellung erfolgt über die Prozedur `DBMS_METADATA.PUT`

GET_XML
Beispiel

ORDIX AG

```
SELECT dbms_metadata.get_xml('TABLE','DEPT','SCOTT') FROM dual;
```

```
"<?xml version="1.0"?><ROWSET><ROW><TABLE_T>
 <VERS_MAJOR>1</VERS_MAJOR>
 <VERS_MINOR>3 </VERS_MINOR>
...
  <ANALYZETIME>2012/03/23 22:03:40</ANALYZETIME>
  <KERNELCOLS>3</KERNELCOLS>
  <PROPERTY>536870912</PROPERTY>
  <PROPERTY2>0</PROPERTY2>
  <SPARE1>6</SPARE1>
  <SPARE2>1</SPARE2>
  <SPARE3>83</SPARE3>
  <OWNER_NAME2>SCOTT</OWNER_NAME2>
  <PCT_FREE>10</PCT_FREE>
 ...
</TABLE_T></ROW></ROWSET>"
```

SET_TRANSFORM_PARAM (I)

ORDIX AG

Ermöglicht die Anpassung des von GET_DDL generierten SQL DDL

```
exec DBMS_METADATA.SET_TRANSFORM_PARAM
(DBMS_METADATA.SESSION_TRANSFORM,'STORAGE',false);

SELECT dbms_metadata.get_ddl('TABLE','DEPT','SCOTT') FROM dual;
```

```
CREATE TABLE "SCOTT"."DEPT"
   (     "DEPTNO" NUMBER(2,0),
         "DNAME" VARCHAR2(14),
         "LOC" VARCHAR2(13),
          CONSTRAINT "PK_DEPT" PRIMARY KEY ("DEPTNO")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  TABLESPACE "USERS"  ENABLE
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING [STORAGE CLAUSE]
  TABLESPACE "USERS"
```

SET_TRANSFORM_PARAM (II)

ORDIX AG

Parametername	Default
PRETTY	true
SQLTERMINATOR	false
CONSTRAINTS	true
REF_CONSTRAINTS	true
CONSTRAINTS_AS_ALTER	false
PARTITIONING	true
SEGMENT_ATTRIBUTES	true
STORAGE	true
TABLESPACE	true
BODY	true
FORCE	true
DEFAULT	-

DBMS_METADATA und context handles

ORDIX AG

- Bisher vorgestellte Verwendung von `DBMS_METADATA` basiert auf der Verwendung von Session-Variablen
- Die bereits beschriebene Funktion `SET_TRANSFORM_PARAM` ist auch zur Konfiguration eines context handles verwendbar
- Die im Folgenden beschriebenen Funktionen/Prozeduren sind im Gegensatz dazu ausschließlich mit einem context handle verwendbar

DBMS_METADATA-Funktionen

OPEN, CLOSE

ORDIX AG

- `OPEN`
  - Spezifiziert den Typ des Objektes das abgerufen werden soll
  - Rückgabewert ist ein context handle für eine Menge von Objekten, der in späteren Aufrufen verwendet werden kann
- `CLOSE`
  - Schließt diesen context handle wieder
  - Löscht die mit diesem context handle bestehende Konfiguration

DBMS_METADATA-Funktionen

SET_REMAP_PARAM

ORDIX AG

- SET_REMAP_PARAM
  - Ermöglicht die Veränderung der Bezeichner der von GET_DDL generierten SQL DDLs

```
DBMS_METADATA.SET_REMAP_PARAM (handle, 'REMAP_SCHEMA', 'SCOTT', 'ORDIX');

SELECT dbms_metadata.get_ddl(handle, 'TABLE','DEPT','SCOTT') FROM dual;
```

```
CREATE TABLE "ORDIX"."DEPT"
   (     "DEPTNO" NUMBER(2,0),
         "DNAME" VARCHAR2(14),
         "LOC" VARCHAR2(13),
          CONSTRAINT "PK_DEPT" PRIMARY KEY ("DEPTNO")
  USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
  TABLESPACE "USERS"  ENABLE
   ) SEGMENT CREATION IMMEDIATE
  PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255
 NOCOMPRESS LOGGING STORAGE ...
  TABLESPACE "USERS"
```

DBMS_METADATA-Funktionen

SET_REMAP_PARAM

ORDIX AG

Parametername	Besonderheiten
REMAP_DATAFILE	-
REMAP_NAME	-
REMAP_SCHEMA	Keine Änderung von Schemabezeichnern innerhalb von PL/SQL Code (Trigger, Stored Procedures etc.)
REMAP_TABLESPACE	-

DBMS_METADATA-Funktionen

Beispiel

ORDIX AG

```
md_handle       := dbms_metadata.OPEN('TABLE');
tr_handle       := dbms_metadata.add_transform(md_handle, 'MODIFY');

dbms_metadata.set_remap_param
(tr_handle, 'REMAP_SCHEMA', 'SCOTT', 'ORDIX');

dbms_metadata.set_remap_param
(tr_handle, 'REMAP_SCHEMA', 'ELLISON', 'DOAG');

ddl_handle := dbms_metadata.add_transform(md_handle, 'DDL');
dbms_metadata.set_transform_param( ddl_handle, 'PRETTY', FALSE);
dbms_metadata.set_transform_param(ddl_handle, 'SQLTERMINATOR', TRUE );

dbms_metadata.set_filter(md_handle, 'DEPT', p_object_name);
v_ddl := dbms_metadata.fetch_clob(md_handle);
dbms_metadata.CLOSE(md_handle);
```

DBMS_METADATA-Funktionen
ADD_TRANSFORM

ORDIX AG

- ADD_TRANSFORM
  - Legt fest, welche Umwandlungen für die Metadaten der FETCH_XXX-Funktionen/Prozeduren verwendet werden sollen
  - Ermöglichen eine Umwandlung der XML-Metadaten in:
    - SQL DDL
    - SXML
    - ALTER_XML
  - Erlaubt Anwendung der Prozeduren SET_TRANSFORM_PARAM und SET_REMAP_PARAM in Verbindung mit context handles

- `SET_FILTER`
  - Ermöglicht es einzelne Objekte zu filtern
  - Filtermöglichkeiten:
    - Objektname
    - Objekttyp
    - Schemaname
    - Name eines abhängigen Objektes
    - Name des Schemas eines abhängigen Objektes
    - Name des Grantors

DBMS_METADATA-Funktionen

Sonstige Funktionen

ORDIX AG

- SET_COUNT
  - Begrenzung der von Fetch_XXX zurückgegebenen Werte
- SET_PARSE_ITEM
  - Anpassung/Ermittlung der Eigenschaften eines Objektattributes
  - Z.B. ob ein Trigger aktiviert (enable) ist
  - Oder die Anpassung des pct_used-Wertes
- GET_QUERY
  - Liefert die Abfrage der FETCH_XX-Prozeduren/Funktionen an das Data Dictionary
  - Reine Debug-Funktionalität

Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

Projekthintergrund (IV)

ORDIX AG

- Vendor-Standard
  - Full-Lieferung der kompletten Datenbank: `DBMS_METADATA` ✓
  - Migrationslieferung von der letzten Version auf die aktuelle: `DDL-Trigger + DBMS_METADATA` ✓
  - Eine Rollback-Lieferung von der aktuellen Version auf die zuletzt gültige

- Generierung eines „DDL-Rollback“
  - Create-Statement → Drop-Statement
  - Drop-Statement → Create-Statement
  - Alter-Statement → Alter-Statement

DBMS_METADATA_DIFF

ORDIX AG

- DBMS_METADATA_DIFF
  - Verfügbar seit 11g R2
  - Vergleicht zwei von DBMS_METADATA generierte SXMLs und generiert hieraus Statements, um das eine Objekt dem anderen anzupassen

SXML

ORDIX AG

- Human-Readable-Version des bereits gezeigten XML durch
  - Verzicht auf Wiedergabe der strukturellen und logischen Komplexität des Data Dictionary
  - Verwendung von im SQL-Standard definierten Schlüsselwörtern (z.B. `Schema` statt `object_owner2`)
- Abbild eines DDL-Statement in XML-Form
- Vollständig dokumentiert
- Ermöglicht durch strukturellen Aufbau schnellen Vergleich

SXML
Beispiel

ORDIX AG

```
SELECT dbms_metadata.get_sxml('TABLE', EMP) from dual;
```

```
<TABLE xmlns="http://xmlns.oracle.com/ku" version="1.0">
<SCHEMA>SCOTT</SCHEMA>
<NAME>EMP</NAME>
<RELATIONAL_TABLE>
<COL_LIST>
   <COL_LIST_ITEM>
   <NAME>EMPNO</NAME>
   <DATATYPE>NUMBER</DATATYPE>
   <PRECISION>4</PRECISION>
   <SCALE>0</SCALE>
   <NOT_NULL/>
</COL_LIST_ITEM>
…
```

COMPARE_SXML

ORDIX AG

- COMPARE_SXML
  - Wird durch den Vergleich von zwei durch DBMS_METADATA generierten SXMLs erstellt
  - Beinhaltet alle Bestandteile des Objekts inkl. der Abweichungen zum zweiten Objekt

- ALTER_XML
  - Wird durch Umwandlung eines COMPARE_SXML erstellt
  - Enthält bereits alle DDL-Statements, um ein Objekt dem Anderen anzugleichen

```
CREATE TABLE ordix1 (a number, b char(10));
CREATE TABLE ordix2 (b char(20));

<TABLE xmlns="http://xmlns.oracle.com/ku" version="1.0">
   <SCHEMA>ORDIX</SCHEMA>
   <NAME value1="ORDIX1">ORDIX2</NAME>
   <RELATIONAL_TABLE>
      <COL_LIST>
         <COL_LIST_ITEM src="1">
            <NAME>A</NAME>
            <DATATYPE>NUMBER</DATATYPE>
         </COL_LIST_ITEM>
         <COL_LIST_ITEM>
            <NAME>B</NAME>
            <DATATYPE>VARCHAR2</DATATYPE>
            <LENGTH value1="10">20</LENGTH>
         </COL_LIST_ITEM>
      </COL_LIST>
...
   </RELATIONAL_TABLE>
</TABLE>
```

ALTER_XML

ORDIX AG

```
<ALTER_XML xmlns="http://xmlns.oracle.com/ku" version="1.0">
   <OBJECT_TYPE>TABLE</OBJECT_TYPE>
  ...
   <ALTER_LIST>
      <ALTER_LIST_ITEM>
         <PARSE_LIST>
            <PARSE_LIST_ITEM>
               <ITEM>NAME</ITEM>
               <VALUE>A</VALUE>
            </PARSE_LIST_ITEM>
            <PARSE_LIST_ITEM>
               <ITEM>CLAUSE_TYPE</ITEM>
               <VALUE>DROP_COLUMN</VALUE>
            </PARSE_LIST_ITEM>
         </PARSE_LIST>
         <SQL_LIST>
            <SQL_LIST_ITEM>
               <TEXT>ALTER TABLE "ORDIX"."ORDIX2" DROP("A")</TEXT>
...
```

Generierte Alter-DDLs

ORDIX AG

Ursprüngliche Objektanlage:

```
CREATE TABLE larry.ordix1 (a number, b char(15));
CREATE TABLE doag.ordix2 (b char(20));
```

Vorgehensweise

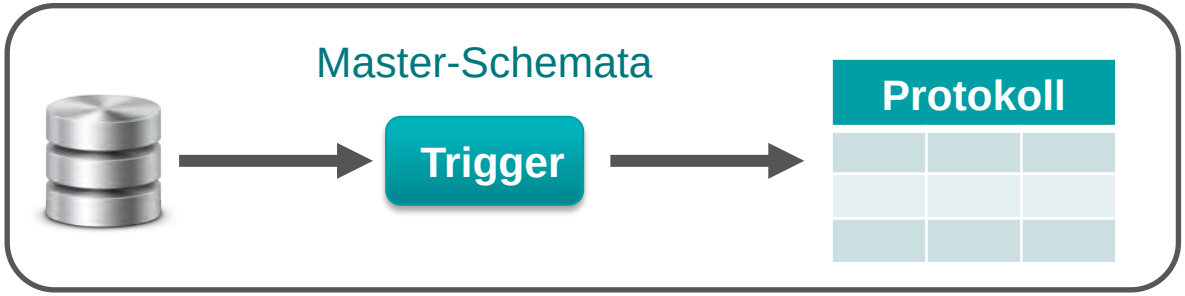
Generierte DDL-Statements:

```
ALTER TABLE LARRY.ORDIX1 DROP (A);
ALTER TABLE LARRY.ORDIX1 MODIFY (B CHAR(20));
ALTER TABLE LARRY.ORDIX1 RENAME TO ORDIX2;
```

DBMS_METADATA_DIFF im Projekteinsatz

ORDIX AG

- Speicherung
  - DDL-Trigger speichert den Zustand eines Objektes vor der Änderung in eine Tabelle
  - Speicherung erfolgt als SXML
- Generierung
  - Erzeugung eines SXML mit dem Zustand nach der Änderung
  - Vergleich der beiden Objekte mit `DBMS_METADATA_DIFF`
  - Erzeugung eines Alter-Statements durch `DBMS_METADATA` mit dem `Compare-SXML` aus `DBMS_METADATA_DIFF`



Projekthintergrund (V)

ORDIX AG

- Vendor-Standard
  - Full-Lieferung der kompletten Datenbank: `DBMS_METADATA` ✓
  - Migrationslieferung von der letzten Version auf die aktuelle: DDL-Trigger ✓
  - Eine Rollback-Lieferung von der aktuellen Version auf die zuletzt gültige ✓

Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

Projekthintergrund (VI)

ORDIX AG

- Bisherige Vorgehensweise erlaubt es, die DDL-Änderungen am Master-Schema zu verfolgen
- Während einer Schemamigration müssen aber auch DML-Operationen zwischen den einzelnen DDL-Statements durchgeführt werden
- Lösungsansatz: DML Statement Auditing

Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

Agenda

ORDIX AG

- Projekthintergrund
- DBMS_METADATA
- DBMS_METADATA_DIFF
- Standard Auditing
- Live-Demo
- Fazit

Probleme / Lessons Learned

ORDIX AG

- Skriptgenerierung
- After DDL-Trigger
- Rename Column-Problem
- Performante Datenmigrationen

ORDIX AG

Vielen Dank für Ihre Aufmerksamkeit!

Zentrale Paderborn
Westernmauer 12 - 16
33098 Paderborn
Tel.: 05251 1063-0

Seminarzentrum Wiesbaden
Kreuzberger Ring 13
65205 Wiesbaden
Tel.: 0611 77840-00

Zentrales Fax:
0180 1 67349 0
0180 1 ORDIX 0

Weitere Geschäftsstellen
in Köln, Münster und Neu-Ulm

E-Mail: info@ordix.de
Internet: http://www.ordix.de