

BaSQing in Oracle Delving into the Depths of the Internal Tables (Just for Fun)



An Introduction to
the Core Oracle Database Tables

BaSQing in Oracle Delving into the Depths of the Internal Tables (Just for Fun)



John Claxton (john_claxton@beaerospace.com)

B/E Aerospace

B/E Aerospace is the worldwide leading manufacturer of aircraft passenger cabin interior products

<http://www.beaerospace.com>

BaSQing in Oracle Delving into the Depths of the Internal Tables (Just for Fun)



What about you?

Do you consider yourself a:

IT Manager?

Database Administrator / Architect?

Developer?

Other?

BaSQing in Oracle Internals



▪ Q: What do the following all have in common?

-IS

-IT

-MIS

-CIO

BaSQing in Oracle Internals



▪ Q: What do the following all have in common?

- IS
- IT
- MIS
- CIO

Answer: INFORMATION!

BaSQing in Oracle Internals



- Data

- Data

- facts (as measurements or statistics)

- used as a basis for reasoning, discussion, or calculation

- *(slightly altered from Miriam-Webster see <http://www.m-w.com>)

BaSQing in Oracle Internals



- Data
- Information

BaSQing in Oracle Internals



- Data

- Information

 - “knowledge obtained from investigation, study, or instruction “

 - *(M-W)

BaSQing in Oracle Internals



- Data
- Information
- Knowledge

- Data

- Information

- Knowledge

- the fact or condition of knowing something with familiarity gained through experience or association

- *(M-W)

BaSQing in Oracle Internals



- Data
- Information
- Knowledge
- Wisdom

BaSQing in Oracle Internals



- Data
- Information
- Knowledge
- Wisdom
 - “ability to discern inner qualities and relationships “
 - *(M-W)

BaSQing in Oracle Internals



- Data
- Information
- Knowledge
- Wisdom
- Understanding

BaSQing in Oracle Internals



- Data
- Information
- Knowledge
- Wisdom
- Understanding
 - “the power to make experience intelligible by applying concepts and categories” *(M-W)

BaSQing in Oracle Internals



- Oracle – what is it?

- Oracle – what is it?
 - Relational Database

BaSQing in Oracle Internals



- Oracle – what is it?
 - Object Oriented Database

BaSQing in Oracle Internals



- Oracle – what is it?

- XML Database

BaSQing in Oracle Internals



- Oracle – what is it?

- Other?

BaSQing in Oracle Internals



- Oracle – what is it?
- Answer – YES!!!

BaSQing in Oracle Internals



Q: Where does it all begin?

BaSQing in Oracle Internals



Answer: The Beginning

BaSQing in Oracle Internals



Answer: The Beginning

... so lets start where we can

and roll back from there

BaSQing in Oracle Internals



Consider DBA_Objects:

Name	Null?	Type
OWNER		VARCHAR2 (30)
OBJECT_NAME		VARCHAR2 (128)
SUBOBJECT_NAME		VARCHAR2 (30)
OBJECT_ID		NUMBER
DATA_OBJECT_ID		NUMBER
OBJECT_TYPE		VARCHAR2 (19)
CREATED		DATE
LAST_DDL_TIME		DATE
TIMESTAMP		VARCHAR2 (19)
STATUS		VARCHAR2 (7)
TEMPORARY		VARCHAR2 (1)
GENERATED		VARCHAR2 (1)
SECONDARY		VARCHAR2 (1)
NAMESPACE		NUMBER
EDITION_NAME		VARCHAR2 (30)

BaSQing in Oracle Internals



So where does DBA_Objects (a view) come from? Hint: Check DBA_VIEWS ...

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2 (30)
VIEW_NAME	NOT NULL	VARCHAR2 (30)
TEXT_LENGTH		NUMBER
TEXT		LONG ←
TYPE_TEXT_LENGTH		NUMBER
TYPE_TEXT		VARCHAR2 (4000)
OID_TEXT_LENGTH		NUMBER
OID_TEXT		VARCHAR2 (4000)
VIEW_TYPE_OWNER		VARCHAR2 (30)
VIEW_TYPE		VARCHAR2 (30)
SUPERVIEW_NAME		VARCHAR2 (30)
EDITIONING_VIEW		VARCHAR2 (1)
READ_ONLY		VARCHAR2 (1)

BaSQing in Oracle Internals



So where does DBA_Objects (a view) come from?

```
SQL> select TEXT from dba_views where view_name='DBA_OBJECTS'
```

```
select u.name, o.name, o.subname, o.obj#, o.dataobj#,  
       decode(o.type#, 0, 'NEXT OBJECT', 1, 'INDEX', 2, 'TABLE', 3, 'CLUSTER',  
              4, 'VIEW', 5, 'SYNONYM', 6, 'SEQUENCE',  
              7, 'PROCEDURE', 8, 'FUNCTION', 9, 'PACKAGE',  
              11, 'PACKAGE BODY', 12, 'TRIGGER',  
              13, 'TYPE', 14, 'TYPE BODY',  
              19, 'TABLE PARTITION', 20, 'INDEX PARTITION', 21, 'LOB',  
              22, 'LIBRARY', 23, 'DIRECTORY', 24, 'QUEUE',  
              28, 'JAVA SOURCE', 29, 'JAVA CLASS', 30, 'JAVA RESOURCE',  
              32, 'INDEXTYPE', 33, 'OPERATOR',  
              34, 'TABLE SUBPARTITION', 35, 'INDEX SUBPARTITION',  
              40, 'LOB PARTITION', 41, 'LOB SUBPARTITION',  
              42, NVL((SELECT 'REWRITE EQUIVALENCE'  
                        FROM sum$ s  
                        WHERE s.obj#=o.obj#  
                              and bitand(s.xpflags, 8388608) = 8388608),  
                        'MATERIALIZED VIEW'),
```

BaSQing in Oracle Internals



So where does DBA_Objects (a view) come from? [cont.]

...

43, 'DIMENSION',
44, 'CONTEXT', 46, 'RULE SET', 47, 'RESOURCE PLAN',
48, 'CONSUMER GROUP',
51, 'SUBSCRIPTION', 52, 'LOCATION',
55, 'XML SCHEMA', 56, 'JAVA DATA',
57, 'EDITION', 59, 'RULE',
60, 'CAPTURE', 61, 'APPLY',
62, 'EVALUATION CONTEXT',
66, 'JOB', 67, 'PROGRAM', 68, 'JOB CLASS', 69, 'WINDOW',
72, 'WINDOW GROUP', 74, 'SCHEDULE', 79, 'CHAIN',
81, 'FILE GROUP', 82, 'MINING MODEL', 87, 'ASSEMBLY',
90, 'CREDENTIAL', 92, 'CUBE DIMENSION', 93, 'CUBE',
94, 'MEASURE FOLDER', 95, 'CUBE BUILD PROCESS',
'UNDEFINED'),

So where does DBA_Objects (a view) come from?
[cont.]

...

```
o.ctime, o.mtime,  
to_char(o.stime, 'YYYY-MM-DD:HH24:MI:SS'),  
decode(o.status, 0, 'N/A', 1, 'VALID', 'INVALID'),  
decode(bitand(o.flags, 2), 0, 'N', 2, 'Y', 'N'),  
decode(bitand(o.flags, 4), 0, 'N', 4, 'Y', 'N'),  
decode(bitand(o.flags, 16), 0, 'N', 16, 'Y', 'N'),  
o.namespace,  
o.defining_edition  
from sys."_CURRENT_EDITION_OBJ" o, sys.user$ u
```

BaSQing in Oracle Internals



So where does DBA_Objects (a view) come from? [cont.]

...

```
where o.owner# = u.user#
      and o.linkname is null
      and (o.type# not in (1 /* INDEX - handled below */,
                          10 /* NON-EXISTENT */))
      or
      (o.type# = 1 and 1 = (select 1
                          from sys.ind$ i
                          where i.obj# = o.obj#
                          and i.type# in (1, 2, 3, 4, 6, 7, 9))))
      and o.name != '_NEXT_OBJECT'
      and o.name != '_default_auditing_options_'
      and bitand(o.flags, 128) = 0
```

So where does DBA_Objects (a view) come from?
[cont.]

...

union all

```
select u.name, l.name, NULL, to_number(null), to_number(null),  
      'DATABASE LINK',  
      l.ctime, to_date(null), NULL, 'VALID','N','N', 'N', NULL, NULL  
from sys.link$ l, sys.user$ u  
where l.owner# = u.user#
```

BaSQing in Oracle Internals



The BSQ Files

Where they are: `$ORACLE_HOME/rdbms/admin`

daw.bsq	dcore.bsq	ddm.bsq	ddst.bsq
denv.bsq	dexttab.bsq	dfmap.bsq	djava.bsq
dImnr.bsq	dmanage.bsq	dobj.bsq	doptim.bsq
dpart.bsq	dplsql.bsq	drac.bsq	drep.bsq
dsec.bsq	dsqlddl.bsq	dsummgt.bsq	dtools.bsq
dtxnspc.bsq	recover.bsq	sql.bsq	

BaSQing in Oracle Internals



The BSQ Files

What they are

daw.bsq - Analytic Workspace catalog tables

dcore.bsq - Core database objects

ddm.bsq - Data Mining objects

ddst.bsq - Daylight Savings Time objects

denv.bsq - Environment (profile, resource, jobs, etc)

dexttab.bsq - External objects

dfmap.bsq - Map objects (files, elements, etc)

djava.bsq - Java objects

dlnmr.bsq - Log Miner objects

dmanage.bsq - SYSAUX Management objects

dobj.bsq - Dir, type, collection, etc objects

doptim.bsq - Statistics objects

BaSQing in Oracle Internals



The BSQ Files

What they are (cont):

dpart.bsq - LOB Partition objects

dpysql.bsq - PL/SQL objects

drac.bsq - TSM (Transparent Service Migration) objects

drep.bsq - Replication objects

dsec.bsq - Security objects (users sys, public; default roles; privileges; etc)

dsqlddl.bsq - Link, Drop User Cascade, Recyclebin, Context, etc

dsummgt.bsq - OLAP objects (sum, slog, snap, etc)

dtools.bsq - Special Roles (export_full_database, metaview, etc)

dtxnspc.bsq - Default UNDO, TEMPORARY tablespaces etc

recover.bsq - RMAN, backup and recovery

sql.bsq - Calls all the others

BaSQing in Oracle Internals



And now – the rest of the story ...

BaSQing in Oracle Internals



QUESTIONS?

?

?

?

?

?

?

BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



BaSQing in Oracle Internals



In the good old days ...

We used to be able to “hack” a users password by capturing the value out of DBA_USERS from the PASSWORD column and then issuing:

```
SQL> Alter User <UserName> Identified By Values  
'<PasswordHash>';
```

BaSQing in Oracle Internals



Sadly, when we query the DBA_USERS table now ...

```
SQL> select username, password from dba_users;
```

USERNAME	PASSWORD

SYSTEM	
SYS	
ORACLE_OCM	
SYSMAN	
...	
DBSNMP	
MGMT_VIEW	
OUTLN	
...	

The Password column is empty! No hash to hold onto.

BaSQing in Oracle Internals



So, using the knowledge we gained from exploring the BSQ files (specifically dcore.bsq).

```
SQL> desc sys.user$
```

Name	Null?	Type
-----	-----	-----
USER#	NOT NULL	NUMBER
NAME	NOT NULL	VARCHAR2(30)
TYPE#	NOT NULL	NUMBER
PASSWORD		VARCHAR2(30)
DATATS#	NOT NULL	NUMBER
...		

Hmmm. There is the NAME and the PASSWORD.

BaSQing in Oracle Internals



So lets see what's in that table:

```
SQL> select name, password from sys.user$;
SYS          F743170344CBD5223
PUBLIC
CONNECT
RESOURCE
DBA
SYSTEM      DC9994F2982CC644
...
TESTUSER    51DF590169B640C42
```

And we also learn that ROLES live in the SYS.USER\$ table. That's an added bonus.

BaSQing in Oracle Internals



Now we can use the 'Alter User...Identified By Values...' syntax:

```
SQL> alter user TESTUSER identified by mynewpassword;
```

```
SQL> connect TESTUSER/mynewpassword
```

```
SQL> ... do some work as TESTUSER ...
```

```
SQL> Alter user TESTUSER identified by '51DF590169B640C42';
```

BaSQing in Oracle Internals



BaSQing in Oracle Internals

