What's New In Database 18c You Wont' Hear About From Oracle

Daniel A. Morgan
dmorgan@siriuscom.com
+1 206-669-2949

16 May, 2018

Unsafe Harbor

- This room is an unsafe harbor
- You can rely on the information in this presentation to help you protect your data, your databases, your organization, and your career
- No one from Oracle has previewed this presentation
- No one from Oracle knows what I'm going to say
- No one from Oracle has supplied any of my materials
- Everything I will present is existing, proven, functionality

Introduction

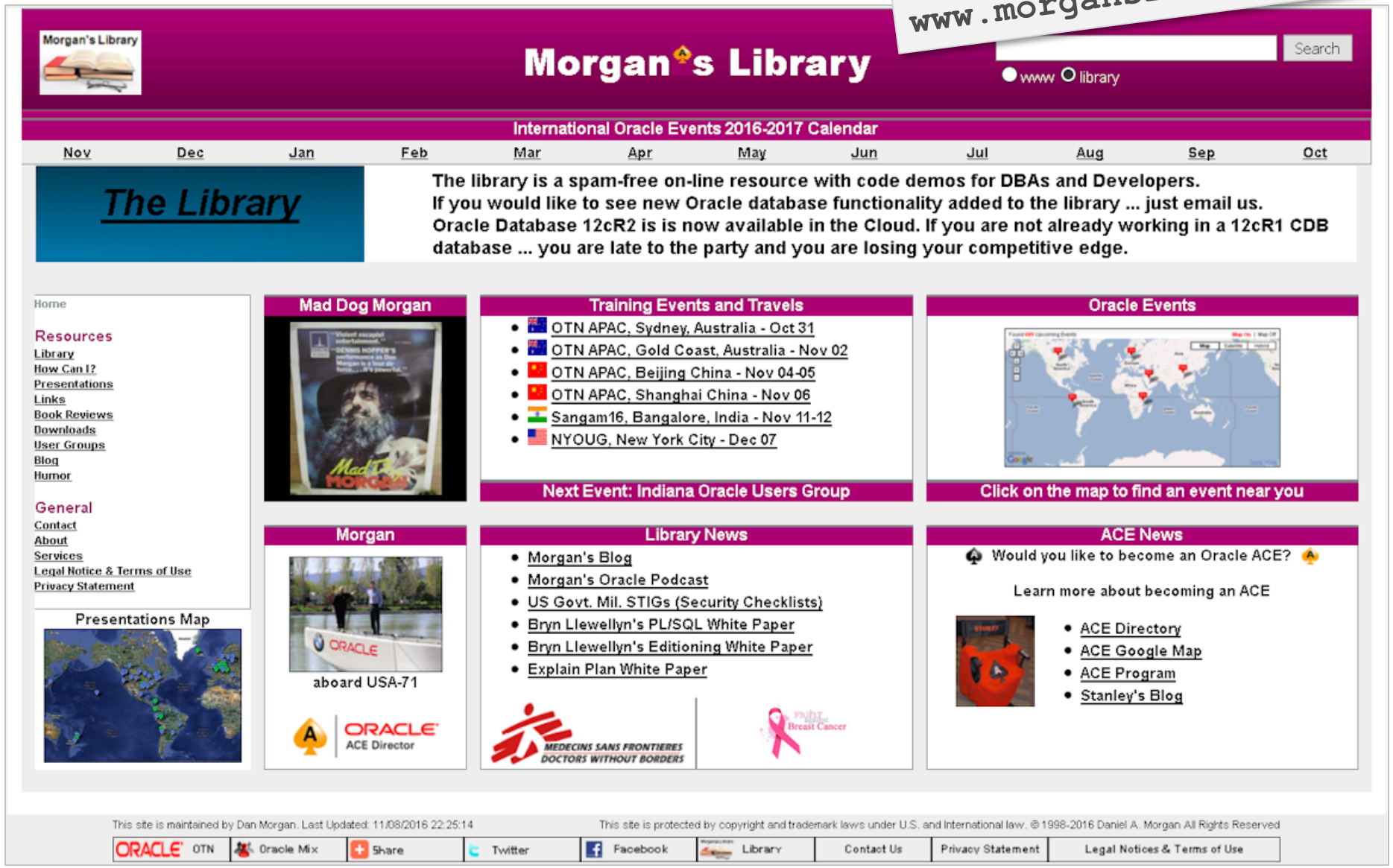
Daniel Morgan

- Oracle ACE Director Alumni
- Oracle Educator
  - Curriculum author and primary program instructor at University of Washington
  - Consultant: Harvard University
  - University Guest Lecturers
    - APAC: University of Canterbury (NZ)
    - EMEA: University of Oslo (Norway)
    - Latin America: Universidad Cenfotec, Universidad Latina de Panama, Technologico de Costa Rica
- IT Professional
  - First computer: IBM 360/40 in 1969: Fortran IV
  - Oracle Database since 1988-9 and Oracle Beta tester
  - The Morgan behind www.morganslibrary.org
  - Member Oracle Data Integration Solutions Partner Advisory Council
  - Vice President Twin Cities Oracle Users Group (Minneapolis-St. Paul)
  - Co-Founder International GoldenGate Oracle Users Group
- Principal Adviser: Sirius **Meta7**

System/370-145 system console

My Websites: Morgan's Library

www.morganslibrary.org



www.morganslibrary.org

Meta7 In Forbes Magazine

ForbesBrandVoice® What is this?

JAN 15, 2018 @ 05:00 AM 20,020

3 Essential DBA Career Priorities For 2018

OracleVoice

Simplify IT, Drive Innovation FULL BIO

Jeff Erickson, Oracle

Many database administrators (DBAs) will go into 2018 wondering if "self-driving" databases will weaken their career prospects. More likely, 2018 will be a year that database technology leaps forward and these valuable data experts take on other, more important responsibilities.

"History is repeating itself," says longtime DBA Dan Morgan, founder of Morgan's Library and principal adviser at tech firm Meta7. Morgan has seen the DBA role evolve amid a long series of technical advances in storage, management, and performance. And each advance asked DBAs to adjust the way they work.

Meta7 is a Division of Sirius

ORACLE Platinum Partner

Forbes AMERICA'S BEST MANAGEMENT CONSULTING FIRMS 2016

Forbes | 2017 AMERICA'S BEST MANAGEMENT CONSULTING FIRMS Powered by Statista

7th straight year CRN Top 50 Providers

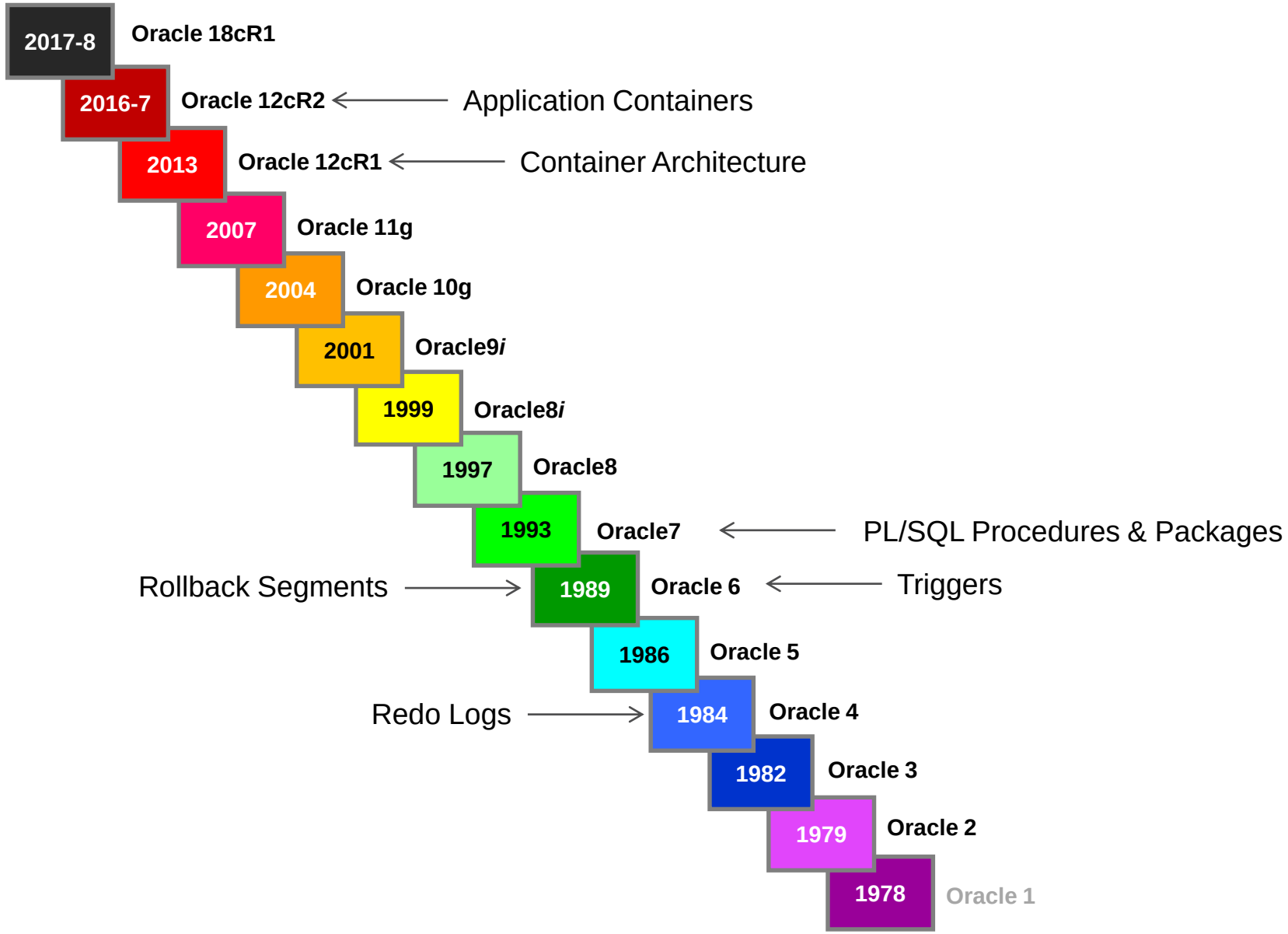
THE CHANNEL CO. CRN SOLUTION PROVIDER 500 2017

THE CHANNEL CO. CRN TECH ELITE 250 2017

THE CHANNEL CO. CRN MSP 500 2017

- Sirius acquired Forsythe Nov 1, 2017
- Combined we are a $3.5B consultancy and VAR
- World's largest IBM integrator
- Second largest security integrator in North America
- Our focus areas
  - Silicon up through Data Integration
  - Stability
  - Security
  - Scalability

A Brief History of the Oracle Database



Installation (1:2)

```
[opc@oem13c2-demo-db18c ~]$ sudo su - oracle
[oracle@oem13c2-demo-db18c ~]$ pwd
/home/oracle
[oracle@oem13c2-demo-db18c ~]$ ls -al
total 3372948
drwx------  8 oracle oinstall       4096 May 15 01:00 .
drwxr-xr-x. 5 root   root           4096 May  1 16:24 ..
-rw-r--r--  1 oracle oinstall        181 May 14 23:10 afiedt.buf
-rw-------  1 oracle oinstall      13202 May 16 02:49 .bash_history
-rw-r--r--  1 oracle oinstall         18 Mar 22  2017 .bash_logout
-rw-r--r--  1 oracle oinstall        175 May  9 18:02 .bash_profile
-rw-r--r--  1 oracle oinstall       1383 May  9 19:42 .bashrc
-rw-r--r--  1 root   root           135 Mar 13 18:23 .bashrc2018-03-13_18:23:24
-rw-r--r--  1 root   root           207 Mar 13 18:24 .bashrc2018-03-13_18:24:49
-rw-r--r--  1 root   root           788 Mar 13 18:42 .bashrc2018-03-13_18:42:36
drwxr-xr-x  3 root   root          4096 Mar 13 18:33 bkup
drwxr-xr-x  7 root   root          4096 Jan 26  2017 database
-rw-r-----  1 oracle oinstall      26662 May 12 18:37 dbca_122_container.rsp
-rw-r-----  1 oracle oinstall      26577 May  8 16:50 dbca_noncontainer.rsp
-rw-r--r--  1 root   root          5500 Mar 13 18:44 dbsetup.out.2872
-rwxr-xr-x  1 oracle oinstall      14204 Jan 24 00:43 dbsetup.sh
-rw-r--r--  1 oracle oinstall       4657 May  6 00:29 initparams.txt
-rwxr-xr-x  1 root   root          2892 Jan 24 00:43 dinject-sshkeys.sh
-rw-r--r--  1 oracle oinstall        171 Nov 15 18:39 .kshrc
-rw-rw-r--  1 oracle oinstall 3453696911 May  9 16:15 linuxx64_12201_database.zip
drwxr-xr-x  4 oracle oinstall       4096 Jan  9 22:32 .mozilla
drwxr-xr-x  2 oracle oinstall       4096 May  9 18:15 .oracle_jre_usage
drwx------  2 oracle oinstall       4096 Mar 13 18:19 .ssh
drwxr-xr-x  2 oracle oinstall       4096 Mar 13 18:19 tmp
-rw-------  1 oracle oinstall      10376 May 15 01:00 .viminfo
-rw-------  1 oracle oinstall         64 May 12 18:32 .Xauthority
```

Installation (2:2)

```
[oracle@oem13c2-demo-db18c ~]$ cd database
[oracle@oem13c2-demo-db18c database]$ ls -al
total 44
drwxr-xr-x  7 root   root     4096 Jan 26  2017 .
drwx------  8 oracle oinstall 4096 May 15 01:00 ..
drwxr-xr-x  4 root   root     4096 Jan 26  2017 install
drwxrwxr-x  2 root   root     4096 Jan 26  2017 response
drwxr-xr-x  2 root   root     4096 Jan 26  2017 rpm
-rwxr-xr-x  1 root   root     8771 Jan 26  2017 runInstaller
drwxrwxr-x  2 root   root     4096 Jan 26  2017 sshsetup
drwxr-xr-x 14 root   root     4096 Jan 26  2017 stage
-rwxr-xr-x  1 root   root      500 Feb  6  2013 welcome.html
[oracle@oem13c2-demo-db18c database]$
```

Read Only Oracle Home (1:4)

- This is one of the most important new Oracle 18c features and is a game changer with respect to how database software is installed
- It is something that was needed for decades for security and now has appeared to satisfy the requirements of the Oracle Cloud and Docker
- Docker containers are read-only ... so how can you deploy an Oracle Database in a Docker container if every ALTER SYSTEM that alers the spfile is non-persistent?
  - Of course you can create symbolic links to the spfile, to sqlnet.ora, listener.ora, tnsnames.ora, the password file, etc.
  - But it is incredibly clumsy
- The Cloud also benefits from a read only home when looked at from the standpoint of Oracle wanting to make claims for security and high availability in the Oracle Cloud

Read Only Oracle Home (2:4)

Read Only Oracle Home (3:4)

- $ORACLE_HOME/bin
  - executable: roohctl

```
[oracle@oem13c2-demo-db18c bin]$ pwd
/u01/app/oracle/product/18.0.0/dbhome_1/bin
[oracle@oem13c2-demo-db18c bin]$ ls -al *ctl
-rwxr-x--x 1 oracle oinstall 234586 Mar 13 18:23 agtctl
-rwxr-x--x 1 oracle oinstall   1578 Feb  8 08:45 aqxmlctl
-rwxr-x--x 1 oracle oinstall 254444 Mar 13 18:24 wdrdactl
-rwxr-x--x 1 oracle oinstall 178299 Mar 13 18:24 lsnrctl
-rwxr-x--- 1 oracle oinstall  35759 Feb  7 18:55 mtactl
-rwxr-xr-x 1 oracle oinstall  33548 Nov 27 08:12 wolfsctl
-rwxr-xr-x 1 oracle oinstall  14558 Feb  8 08:45 onsctl
-rwxr-xr-x 1 oracle oinstall   5440 Nov 23 06:18 wrhpctl
-rwxr-x--- 1 oracle oinstall   4631 Feb  8 08:45 roohctl
-rwxr-xr-x 1 oracle oinstall  11460 Feb  8 08:45 srvctl
[oracle@oem13c2-demo-db18c bin]$
```


Read Only Oracle Home (4:4)

```
[oracle@oem13c2-demo-db18c bin]$ roohctl -help
Usage:  roohctl [<flag>] [<command> <option>]
Following are the possible flags:
        -help

Following are the possible commands:
        -enable Enable Read-only Oracle Home
                [-nodeList List of nodes in a cluster environment]
```

- Now we have 4 different environment variables to improve our deployments
- ORACLE_BASE
  - Used to externalize the mutable files outside of the ORACLE_HOME directory structure
- ORACLE_HOME
  - The name provided when installing using OUI and DBCA ... findable in the Inventory
- ORACLEBASE_HOME
  - Mutable SQL*NET config, log and trace files and the /assistant DBCA templates
- ORACLEBASE_CONFIG
  - Mutable configuration files (.ora and .dat usually found in /dbs)

Users

New: 12cR2

APEX_050100
APEX_INSTANCE_ADMIN_USER
APEX_LISTENER
APEX_REST_PUBLIC_USER
DBJSON
DBSFWUSER
GGSYS
HRREST
OBE
ORDS_METADATA
ORDS_PUBLIC_USER
REMOTE_SCHEDULER_AGENT
RESTFUL
SYS$UMF
SYSRAC
XDBEXT
XDBPM
XFILES

Dropped

SPATIAL_WFS_USR

New: 18cR1

APEX_PUBLIC_USER
APEX_REST_PUBLIC_USER
FLOW_FILES

Dropped

SPATIAL_CSW_ADMIN_USR

Roles

New

None

Dropped

JAVA_DEPLOY
SPATIAL_CSW_ADMIN
XFILES_ADMINISTRATOR
XFILES_USER

System Privileges

New

READ ANY ANALYTIC VIEW CACHE
TEXT DATASTORE ACCESS
WRITE ANY ANALYTIC VIEW CACHE

Dropped

EXEMPT DDL REDACTION POLICY
EXEMPT DML REDACTION POLICY

Initialization Parameters

New

ADG_ACCOUNT_INFO_TRACKING
AWR_PDB_MAX_PARALLEL_SLAVES
DBFIPS_140
FORWARD_LISTENER
INMEMORY_AUTOMATIC_LEVEL
INMEMORY_OPTIMIZED_ARITHMETIC
INMEMORY_PREFER_XMEM_MEMCOMPRESS
INMEMORY_PREFER_XMEM_PRIORITY
INMEMORY_XMEM_SIZE
MEMOPTIMIZE_POOL_SIZE
MULTISHARD_QUERY_DATA_CONSISTENCY
MULTISHARD_QUERY_PARTIAL_RESULTS
OPTIMIZER_IGNORE_HINTS
OPTIMIZER_IGNORE_PARALLEL_HINTS
PARALLEL_MIN_DEGREE
PDB_TEMPLATE
PRIVATE_TEMP_TABLE_PREFIX
RESOURCE_MANAGER_CPU_ALLOCATION
STANDBY_PDB_SOURCE_FILE_DBLINK
STANDBY_PDB_SOURCE_FILE_DIRECTORY
TDE_CONFIGURATION
UNIFIED_AUDIT_SYSTEMLOG
WALLET_ROOT

Changed Values

DB_BLOCK_CHECKING

Desupported / Changed Values

DBA_REGISTERED_MVIEW_GROUPS

GLOBAL_CONTEXT_POOL_SIZE
MAX_ENABLED_ROLES
OPTIMIZER_ADAPTIVE_FEATURES
PARALLEL_AUTOMATIC_TUNING
PARALLEL_IO_CAP_ENABLED
PARALLEL_SERVER
PARALLEL_SERVER_INSTANCES
STANDBY_ARCHIVE_DEST
USE_INDIRECT_DATA_BUFFERS
UTL_FILE_DIR

Dropped Built-In PL/SQL Packages

- All of Streams Change Data Capture (CDC)
  - DBMS_CDC_EXPDP
  - DBMS_CDC_EXPVDP
  - DBMS_CDC_IMPDP
  - DBMS_CDC_IMPDPV
  - DBMS_CDC_IPUBLISH
  - DBMS_CDC_ISUBSCRIBE
  - DBMS_CDC_PUBLISH
  - DBMS_CDC_SUBSCRIBE
  - DBMS_CDC_SYS_IPUBLISH
  - DBMS_CDC_DPUTIL
  - DBMS_CDC_UTILITY
- DBMS_XMLQUERY
- DBMS_XMLSAVE
- Oracle Multimedia and DICOM

Temporary Tables (1:3)

- Global Temporary Tables are persistent tables defined in the data dictionary but created in the temporary tablespace

```
CREATE GLOBAL TEMPORARY TABLE gtt_zip (
zip_code      VARCHAR2(5),
by_user       VARCHAR2(30),
entry_date    DATE)
ON COMMIT DELETE ROWS;
```

```
CREATE GLOBAL TEMPORARY TABLE gtt_zip3 (
zip_code      VARCHAR2(5),
by_user       VARCHAR2(30),
entry_date    DATE)
ON COMMIT PRESERVE ROWS;
```

- Private Temporary Tables have similar characteristics but are created in memory

```
CREATE PRIVATE TEMPORARY TABLE ora$ptt_ocdr(
rid   NUMBER(10),
rname VARCHAR2(20))
ON COMMIT PRESERVE DEFINITION
ON COMMIT DELETE ROWS AS
SELECT * FROM servers;
```

```
CREATE PRIVATE TEMPORARY TABLE uwclass.ora$ptt_ocpr(
ON COMMIT DROP DEFINITION
ON COMMIT PRESERVE ROWS AS
SELECT * FROM uwclass.servers;
```

Temporary Tables (2:3)

- The Oracle docs are incomplete about PTTs so keep the following in mind when use them
  - A PTT's name must be prefixed with the parameter string value for "private_temp_table_prefix". If you don't like the Oracle Corp default, and I don't (too many bytes) change it

```
SQL> show parameter private

NAME                           TYPE    VALUE
------------------------------ ------- ---------
private_temp_table_prefix      string  ORA$PTT_
```

  - You cannot create a PTT as SYS and possibly with other privileged accounts. If you try to do so the error message you get will be misleading: Ignore it and move to a non-privileged schema.

```
SQL> sho user
USER is "SYS"

SQL> CREATE PRIVATE TEMPORARY TABLE ora$ptt_msg_fail
  2  ON COMMIT PRESERVE DEFINITION
  3  ON COMMIT DELETE ROWS AS
  4* SELECT * FROM user_objects;
ON COMMIT DELETE ROWS AS
*
ERROR at line 3:
ORA-00922: missing or invalid option
```

Temporary Tables (3:3)

- All DDL contains 2 implicit commits
- If you create a Temporary Table with ON COMMIT DELETE ROWS ... any DDL will empty the temporary table(s)

Built-In Functions (1:3)

- APPROX_COUNT
  - Returns the approximate count of an expression. With MAX_ERROR the function returns the maximum error between the actual and approximate count.

```
APPROX_COUNT(<expression> [, 'MAX_ERROR']) RETURN NUMBER;

SELECT department_id, job_id, APPROX_COUNT(*)
FROM employees
GROUP BY department_id, job_id
HAVING APPROX_RANK (PARTITION BY department_id
ORDER BY APPROX_COUNT(*) DESC) <= 10;
```

- APPROX_RANK
  - Returns the approximate rank from an optional PARTITION BY clause followed by a mandatory ORDER BY ... DESC clause. The PARTITION BY key must be a subset of the GROUP BY key. The ORDER BY clause must include either APPROX_COUNT or APPROX_SUM.

```
APPROX_MEDIAN(<expression> [PARTITION BY <partition_by_clause> [ORDER BY <order_by_clause> DESC])

SELECT department_id, job_id, APPROX_COUNT(*)
FROM employees
GROUP BY department_id, job_id
HAVING APPROX_RANK (PARTITION BY department_id ORDER BY APPROX_COUNT(*) DESC) <= 10;
```

Built-In Functions (2:3)

- APPROX_SUM
  - Returns the approximate sum of an expression. If you supply MAX_ERROR as the second argument, then the function returns the maximum error between the actual and approximate sum. You must use this function with a corresponding APPROX_RANK function in the HAVING clause. If a query uses APPROX_COUNT, APPROX_SUM, or APPROX_RANK, then the query must not use any other aggregation functions.

```
APPROX_COUNT(<expression> [, 'MAX_ERROR']) RETURN NUMBER;

SELECT department_id, job_id, APPROX_SUM(salary)
FROM employees
GROUP BY department_id, job_id
HAVING APPROX_RANK (PARTITION BY department_id
ORDER BY APPROX_SUM(salary) DESC) <= 10;
```

Built-In Functions (3:3)

- ROUND_TIED_TO_EVEN
  - Returns n rounded to integer places according to the following rules:
    - 1. If integer is positive, n is rounded to integer places to the right of the decimal point
    - 2. If integer is not specified, then n is rounded to 0 places
    - 3. If integer is negative, then n is rounded to integer places to the left of the decimal point

```
ROUND_TIES_TO_EVEN(n [, INTEGER DESC])

SQL> SELECT round_ties_to_even(0.05, 1) "ROUND_EVEN"
  2  FROM dual;

ROUND_EVEN
----------
         0

SQL> SELECT round_ties_to_even(41.572,-1) "ROUND_EVEN"
  2  FROM dual;

ROUND_EVEN
----------
        40
```

```
SQL> SELECT round_ties_to_even(41.572,1) "ROUND_EVEN"
  2  FROM dual;

ROUND_EVEN
----------
      41.6

SQL> SELECT round_ties_to_even(41.572,2) "ROUND_EVEN"
  2  FROM dual;

ROUND_EVEN
----------
     41.57
```

New Built-In Packages

- DBMS_AWR_PROTECTED
- DBMS_ISCHEDFW
- DBMS_ISCHED_AGENT
- DBMS_ISCHED_UTL
- DBMS_MEMOPTIMIZE
- DBMS_PDB_APP_CON
- DBMS_SODA
- DBMS_SQLSET
- DBMS_STATS_INTERNAL_AGG
- DBMS_WORKLOAD_CAPTURE_I
- DBMS_WORKLOAD_REPLAY_I
- DBMS_WRR_REPORT
- DBMS_XDS_INT
- SCHEDULER$_QP_19951

DBMS_MEMOPTIMIZE

- Provides an interface for managing data in the memoptimize pool which is an SGA cache that stores table data and hash index related to the Memoptimized Rowstore
- The package provides the following functionality
  - POPULATE
    - Populate the Memoptimized Rowstore hash index with the data related to a specific table
  - DROP_OBJECT
    - Removes data from the Memoptimized Rowstore hash index related to a specific table

DBMS_SODA

- A PL/SQL package implementing Simple Oracle Document Access (SODA)
- SODA allows use of the Oracle Database as a NoSQL document store
- The core abstraction provided by SODA is that of document collections
- The DBMS_SODA package allows developers to create, list, and delete document collections with PL/SQL, and to perform CRUD (create, replace, update, delete) operations on documents
- All DDL functions are encapsulated within this package
- The package contains the following objects
  - CREATE_COLLECTION
  - DROP_COLLECTION
  - LIST_COLLECTION_NAMES
  - OPEN_COLLECTION

DBMS_SQLSET

- The DBMS_SQLTUNE package provides an interface to manage SQL tuning sets
- This package provides the same subprograms, although in some cases with slightly different names, as the SQL tuning set subprograms in DBMS_SQLTUNE
- The difference is that DBMS_SQLSET does not require the Oracle Tuning Pack license

Disaster Recovery with Data Guard (1:5)

- The DBMS_DRS built-in package has 27 new objects
- DBMS_DRS is no documented and supported for DBAs and Developers but these new functions provide an excellent window into changes Oracle is making to Data Guard Physical Standbys
  - ADD_DATABASE
    - Add a standby database to a broker configuration. database_ci is the connection identifier
  - ADD_FAR_SYNC
    - Add a far sync instance to a broker configuration. far_sync_ci is the connection identifier
  - CHECK_CONNECT
    - Check network connectivity to the specified member
  - CREATE_CONFIGURATION
    - Creates a broker configuration. The primary database will be automatically added to the configuration by this procedure. Must be called on a primary database.
  - DISABLE_FS_FAILOVER
    - Disables Fast Start Failover

Disaster Recovery with Data Guard (2:5)

- DO_OBSERVE
  - Observer operation API - observer's operation to control FSFO since 12.2. (replaces Ping, ReadyToFailover, and StateChangeRecorded)
- DUMP_BROKER
  - Dumps critical internal data of the broker process to a file
- DUMP_OBSERVER
  - Dumps critical internal data of client-side observer process to a file
- ENABLE_CONFIGURATION
  - Enables broker management of a Data Guard configuration. It must be called on the primary database. Return 0 means enable was successful, otherwise returns an error number.
- ENABLE_DATABASE
  - Used to enable broker management of a database within the broker configuration. It must be called on the primary database.
- ENABLE_FAR_SYNC
  - Enable broker management of a far sync instance within the broker configuration. It must be called on the primary database.

Disaster Recovery with Data Guard (3:5)

- ENABLE_FS_FAILOVER
  - Used to enable fast-start failover
- REMOVE_CONFIGURATION
  - Removes a broker configuration. It must be called on the primary database.
- REMOVE_DATABASE
  - Used to remove a database from the broker configuration. It must be called on the primary database.
- REMOVE_FAR_SYNC
  - Removes a far sync instance from the broker configuration. It must be called on the primary database.
- REPLACE_MEMBER_NAME_IN_PROPS
  - Replaces a member name with another member name in all broker properties
- RESET_CONFIGURATION_PROPERTY
  - Resets configuration-level property, not database or far sync instance property, to its default value.

Disaster Recovery with Data Guard (4:5)

- RESET_DATABASE_PROPERTY
  - Resets a database configurable property to its default value
- RESET_FAR_SYNC_PROPERTY
  - Resets a far sync instance configurable property to its default value
- SET_CONFIGURATION_PROPERTY
  - Used to set configuration-level property (not a database or far sync property). Can be used to set both integer and character string properties.
- SET_DATABASE_PROPERTY
  - Used to set a database configurable property. Can be used to set both integer and character string properties.
- SET_FAR_SYNC_PROPERTY
  - Used to set a far sync instance's configurable property. Can be used to set both integer and character string properties.
- SET_PROTECTION_MODE
  - Changes the protection mode to the mode specified. To prevent including database restart logic this procedure does not support the promotion of the protection mode from maximum performance to maximum protection.

Disaster Recovery with Data Guard (5:5)

- STOP_OBSERVER
  - Stops the fast-start failover observers in a data guard broker configuration
- WAIT
  - Waits up to the number of seconds specified by the max_wait_time argument for the event specified by the event_type parameter to prevail

Hierarchical Profiling (1:2)

- DBMS_HPROF
  - ANALYZE
    - 2 new overloads for analyzing the raw profiler output and produces hierarchical profiler information in database tables

Overload 4

```
dbms_hprof.analyze(
trace_id      IN NUMBER,
summary_mode  IN BOOLEAN      DEFAULT FALSE,
trace         IN VARCHAR2     DEFAULT NULL,
skip          IN PLS_INTEGER DEFAULT 0,
collect       IN PLS_INTEGER DEFAULT NULL,
run_comment   IN VARCHAR2     DEFAULT NULL,
profile_uga   IN BOOLEAN      DEFAULT NULL,
profile_pga   IN BOOLEAN      DEFAULT NULL)
RETURN NUMBER;
```

Overload 5

```
dbms_hprof.analyze(
trace_id     IN  NUMBER,
report_clob OUT CLOB,
trace        IN  VARCHAR2     DEFAULT NULL,
skip         IN  PLS_INTEGER DEFAULT 0,
collect      IN  PLS_INTEGER DEFAULT NULL,
profile_uga IN  BOOLEAN      DEFAULT NULL,
profile_pga IN  BOOLEAN      DEFAULT NULL);
```

  - CREATE TABLES
    - Creates the table dbmshp_trace_data and sequence dbmshp_tracenumber sequence

```
dbms_hprof.start_profiling(force_it IN BOOLEAN DEFAULT FALSE);

exec dbms_hprof.create_tables(TRUE);
```

Hierarchical Profiling (2:2)

- DBMS_HPROF
  - START_PROFILING
    - 1 new overload for starting PL/SQL profiling

```
dbms_hprof.start_profiling(
max_depth   IN PLS_INTEGER DEFAULT NULL,
profile_uga IN BOOLEAN     DEFAULT NULL,
profile_pga IN BOOLEAN     DEFAULT NULL,
sqlmonitor  IN BOOLEAN     DEFAULT TRUE,
run_comment IN VARCHAR2    DEFAULT NULL)
RETURN NUMBER;
```

Polymorphic Table Functions (1:4)

- PTFs are a new type of table function, a function that returns a collection of rows, whose return type is determined by the arguments passed into the PTF
- The new PTFs provides an efficient and scalable framework to extend the analytical capabilities of the Oracle Database
- A query writer is able to call these functions without knowing the details of the implementation and the PTF doesn't need to know about the details or how the function is being executed or whether the input rows are partitioned or ordered
- PTFs are useful when SQL developers and database administrators want to provide generic extensions which work for arbitrary input tables or queries
- Making possible queries like this producing JSON as output

```
SELECT * FROM to_doc(scott.dept)

{"DEPTNO":10, "DNAME":"ACCOUNTING", "LOC":"NEW YORK"}
{"DEPTNO":20, "DNAME":"RESEARCH", "LOC":"DALLAS"}
{"DEPTNO":30, "DNAME":"SALES", "LOC":"CHICAGO"}
{"DEPTNO":40, "DNAME":"OPERATIONS", "LOC":"BOSTON"}
```

Polymorphic Table Functions (2:4)

- The DBMS_TF package was initially released in version 12.2 and is now extended with new capabilities in 18.1
  - Contains types, constants, and subprograms that can be used by Polymorphic Table Functions (PTFs)
  - Provides server and client services to get rows from the database and send back new rows

12.2

```
GET_COL
GET_ENV
GET_XID
GET_ROW_SET
PUT_COL
PUT_ROW_SET
SUPPORTED_TYPE
TRACE
```

18.1

```
COLUMN_TYPE_NAME
COL_TO_CHAR
CSTORE_EXISTS
CSTORE_GET
ROW_REPLICATION
ROW_TO_CHAR
XSTORE_CLEAR
XSTORE_EXISTS
XSTORE_GET
XSTORE_REMOVE
XSTORE_SET
```

Polymorphic Table Functions (3:4)

- Examples from the web of dbms_tf usage

```
CREATE PACKAGE to_doc_p AS
 FUNCTION desc(tab IN OUT dbms_tf.table_t,cols IN dbms_tf.columns_t DEFAULT NULL) RETURN dbms_tf.describe_t;
END to_doc_p;
```

```
CREATE PACKAGE BODY to_doc_p AS
 FUNCTION desc(tab IN OUT dbms_tf.table_t,cols IN dbms_tf.columns_t DEFAULT NULL) RETURN dbms_tf.describe_t AS
 BEGIN
   FOR i IN 1 .. tab.column.COUNT LOOP
     CONTINUE WHEN NOT DBMS_TF.supported_type(tab.column(i).description.TYPE);
     IF cols IS NULL THEN
       tab.column(i).for_read := TRUE;
       tab.column(i).pass_through := FALSE;
       CONTINUE;
     END IF;
     FOR j IN 1 .. cols.COUNT LOOP
       IF (tab.column(i).description.name = cols(j)) THEN
         tab.column(i).for_read     := TRUE;
         tab.column(i).pass_through := FALSE;
       END IF;
     END LOOP;
   END LOOP;
   RETURN dbms_tf.describe_t(new_columns => dbms_tf.columns_new_t(1 =>
          dbms_tf.column_metadata_t(name =>'DOCUMENT')));
 END;
END;
```

Polymorphic Table Functions (4:4)

- Examples from the web of dbms_tf usage

```
dbms_tf.get_col(
ColumnId   IN            NUMBER,
Collection IN OUT NOCOPY "<V2_TABLE_1>");
pragma interface(c, Get_Col);
```

```
CREATE OR REPLACE PROCEDURE fetch_rows AUTHID CURRENT_USER IS
 col1 dbms_tf.tab_clob_t;
 col2 dbms_tf.tab_colb_t;
 out1 dbms_tf.tab_clob_t;
 out2 dbms_tf.tab_clob_t;
BEGIN
  dbms_tf.get_col(1, col1);
  dbms_tf.get_col(2, col2);

  FOR i IN 1 .. col1.COUNT LOOP
    out1(i) := 'ECHO-' || col1(i);
  END LOOP;

  FOR i IN 1 .. col2.COUNT LOOP
    out2(i) := 'ECHO-' || col2(i);
  END LOOP;

  dbms_tf.put_col(1, out1);
  dbms_tf.put_col(2, out2);
END fetch_rows;
/
```

Real Application Testing: Capture (1:3)

- New Public Capabilities
  - ENCRYPT and DECRYPT Capture (new procedures)
  - START_CAPTURE (new parameters)

```
dbms_workload_capture.encrypt_capture(
src_dir    IN VARCHAR2,
dst_dir    IN VARCHAR2,
encryption IN VARCHAR2 DEFAULT 'AES256'); -- options: 'AES128', 'AES192', 'AES256'

exec dbms_workload_capture.encrypt_capture('SRCDIR', 'TGTDIR', 'AES256');
```

```
dbms_workload_capture.decrypt_capture(
src_dir IN VARCHAR2,
dst_dir IN VARCHAR2);

exec dbms_workload_capture.decrypt_capture('SRCDIR', 'TGTDIR');
```

Real Application Testing: Capture (2:3)

- New Public Capabilities
  - START_CAPTURE (new parameters)
    - PLSQL_MODE
      - TOP_LEVEL: only top-level PL/SQL calls are captured
      - EXTENDED: both top-level PL/SQL calls and SQL called from PL/SQL are captured
    - ENCRYPTION
      - NULL: no encryption
      - AES128
      - AES 192
      - AES256

```
dbms_workload_capture.start_capture(
name              IN VARCHAR2,
dir               IN VARCHAR2,
duration          IN NUMBER   DEFAULT NULL,
default_action    IN VARCHAR2 DEFAULT 'INCLUDE',
auto_unrestrict   IN BOOLEAN  DEFAULT TRUE,
capture_sts       IN BOOLEAN  DEFAULT FALSE,
sts_cap_interval  IN NUMBER   DEFAULT 300,
plsql_mode        IN VARCHAR2 DEFAULT 'TOP_LEVEL',
encryption        IN VARCHAR2 DEFAULT NULL);
```

Real Application Testing: Capture (3:3)

- New Private Capabilities
  - GET_STATE (1 if capturing the current session; otherwise 0)

```
dbms_workload_capture.get_state RETURN BINARY_INTEGER;

SQL> SELECT dbms_workload_capture.get_state
  2  FROM dual;

 GET_STATE
----------
         0
```

  - START_BATCH_CAPTURE (same syntax as START_CAPTURE)
  - SWITCH_BUCKET
    - Signals all connected sessions to store workload captures into a new bucket. By default, SWITCH_BUCKET will create an AWR snapshot for the workload captured in the current bucket.

```
dbms_workload_capture.switch_bucket(create_snapshot IN BOOLEAN DEFAULT TRUE);

exec dbms_workload_capture.switch_bucket(FALSE);
```

Real Application Testing: Replay (1:4)

- New Public Capabilities
  - ASSIGN_GROUP_TO_INSTANCE
    - Assigns a group of capture files to be processed by a particular node in a RAC cluster

```
dbms_workload_replay.assign_group_to_instance(
group_id         IN NUMBER,
instance_number IN NUMBER);

exec dbms_workload_replay.assign_group_to_instance(6, 2);
```

  - LOAD_LONG_SQLTEXT
    - Loads captured SQL statements whose length is greater than 1000 characters

```
dbms_workload_replay.load_long_sqltext(capture_id IN NUMBER);

exec dbms_workload_replay.load_long_sqltext(11);
```

Real Application Testing: Replay (2:4)

- New Public Capabilities
  - SET_SQL_MAPPING
    - Specifies SQL statements to be skipped or replaced in replay

```
-- overload 1
dbms_workload_replay.set_sql_mapping(
schedule_cap_id      IN NUMBER,
sql_id               IN VARCHAR2,
operation            IN VARCHAR2,
replacement_sql_text IN VARCHAR2);

-- overload 2
dbms_workload_replay.set_sql_mapping(
sql_id               IN VARCHAR2,
operation            IN VARCHAR2,
replacement_sql_text IN VARCHAR2);
```

Real Application Testing: Replay (3:4)

- New Private Capabilities
  - GET_STATE (1 if capturing the current session; otherwise 0)

```
dbms_workload_replay.get_state RETURN BINARY_INTEGER;

SQL> SELECT dbms_workload_capture.get_state
  2  FROm dual;

 GET_STATE
----------
         0
```

  - GROUP_WORKLOAD
    - Finds a grouping for the workload, resolves conflictsm and merges them into groups based on the number of files until the number of groups is less than the max. Results are stored in WRR$_WORKLOAD_GROUPS and WRR$_REPLAY_LOGIN_QUEUE.

```
dbms_workload_replay.group_workload(max_groups IN NUMBER);

exec dbms_workload_replay.group_workload(10);
```

Real Application Testing: Replay (4:4)

- New Private Capabilities
  - LOAD_TRACKED_COMMITS: Overload 1
    - Overload 1Commits data for a given replay id

```
dbms_workload_replay.load_tracked_commits(replay_id IN NUMBER);

exec dbms_workload_replay.load_tracked_commits(8);
```

  - LOAD_TRACKED_COMMITS: Overload 2
    - Load tracked commits data for all replays in a given directory object

```
dbms_workload_replay.load_tracked_commits(replay_dir IN VARCHAR2);

exec dbms_workload_replay.load_tracked_commits('u03/apps/oracle/replay');
```

```
*
ERROR at line 1:
ORA-00028: your session has been killed
```

Thank You

META7™

Questions? dmorgan@siriuscom.com