

# Bab 5

## Membuat Data Dictionary View

Pada bab ini akan dibahas mengenai langkah-langkah administrasi database yang meliputi :

- Mengkonstruksi view untuk data dictionary
- Menggunakan data dictionary
- Menyiapkan lingkungan operasi PL/SQL menggunakan script administratif
- Melakukan administrasi terhadap store procedure dan paket

### 5.1. Struktur Data Dictionary

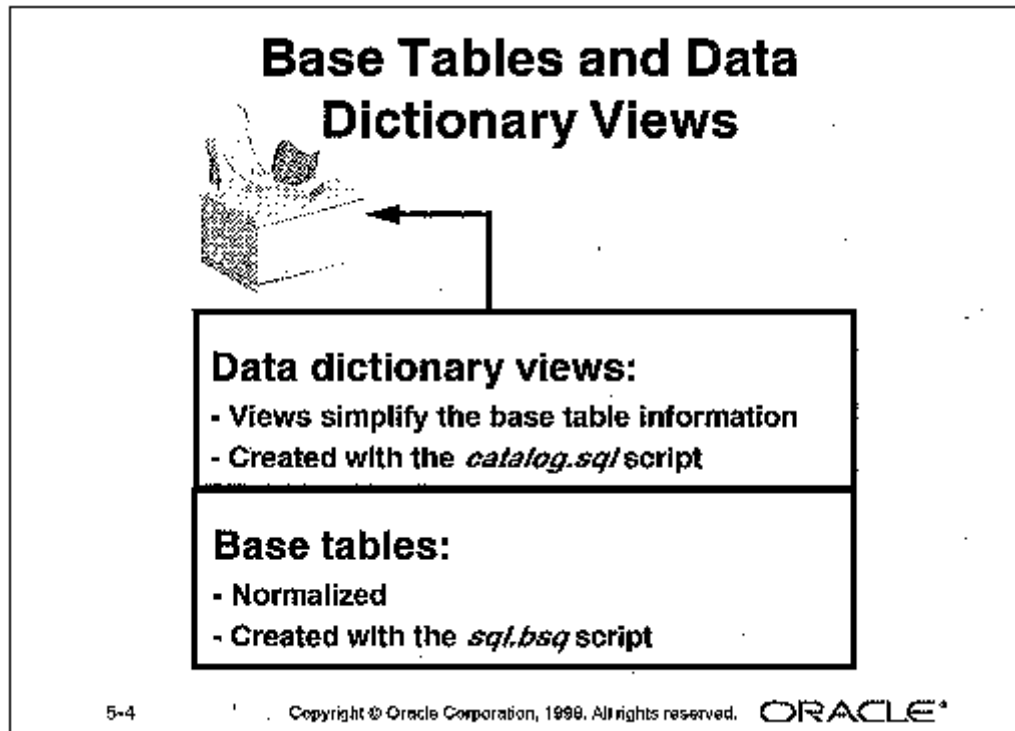
Secara umum data dictionary berisi informasi tentang :

- Struktur database secara fisik dan logika.
- Nama, definisi, dan alikasi ruang memory ayng diperluakn oleh schema obyek.
- Berisi integrity constraint
- Database tentang user dan privilege
- Auditing.

Data dictionary merupakan bagian yang penting dalam database Oracle. Data dictionary berbentuk beberapa tabel dan view yang dipakai sebagai referensi untuk menyediakan informasi tentang database secara terpadu.

Tabel-tabel dalam data dictionary dibuat berdasarkan file script *sql.bsq* pada saat pembuatan database. Data dictionary merupakan sumber informasi sentral untuk server Oracle dan untuk database user maupun administrator.

Data dictionary di-update oleh server Oracle pada saat dieksekusinya perintah DDL. Selanjutnya perintah DML yang menghasilkan keputusan untuk menambah dimensi tabel maka akan mengupdate data dictionary.



Data dictionary ditempatkan pada tablespace SYSTEM dan dimiliki oleh user SYS, yang terdiri dari dua bagian yaitu :

- *Base Table*

Pondasi dari data dictionary adalah terdiri dari beberapa table dasar, dimana server Oracle selalu membaca dan menulis pada tabel ini. User database jarang sekali melakukan akses secara langsung terhadapnya karena dilakukan normalisasi dan informasinya dilakukan encoding.

Sebagai contoh jika anda melakukan query terhadap tabel IND\$ untuk mendapatkan informasi tentang definisi index didalam database, atau menampilkan tabel OBJ\$ yang berisi informasi definisi obyek didalam database.

Perintah DDL seperti INSERT, UPDATE, dan DELETE tidak pernah bisa digunakan untuk meng-update tabel dasar data dictionary secara langsung, kecuali hanya tabel AUD\$.

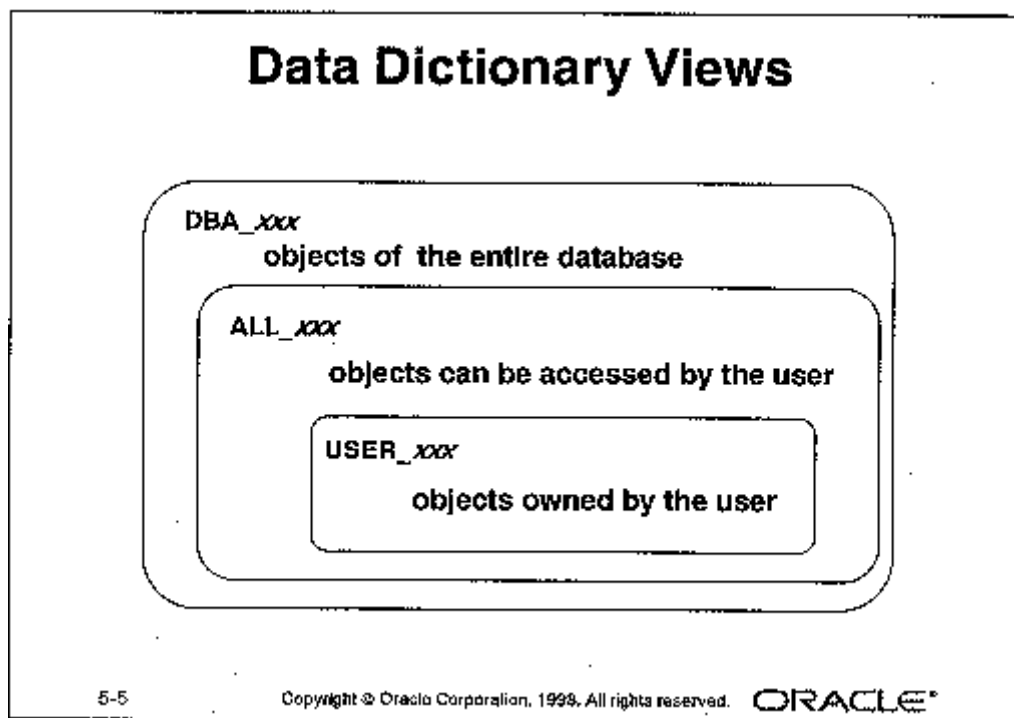
- *Data Dictionary View*

Data dictionary view dibuat pada saat menjalankan script *catalog.sql*

View ini mendecode dan merangkum informasi dari tabel dasar data dictionary. Agar user dapat mengakses data dictionary dengan mudah maka dibuat public synonym. Umumnya user mengakses data dictionary lewat view dari pada mengakses tabel secara langsung.

## 5.2. Katagori View Data Dictionary

View data dictionary dibagi menjadi tiga katagori yaitu USER, ALL, dan DBA. Pada beberapa hal view-view ini berisi informasi yang sama/mirip dan untuk membedakan diantara ketiganya adalah menggunakan prefix.



- *Prefix USER*

View ini dapat diakses oleh sembarang user dan umumnya mereferensikan obyek yang dimiliki oleh user tersebut. Sebagai contoh USER\_TABLES berisi

informasi semua tabel yang dimiliki oleh user tersebut. View ini memberikan informasi kepada ALL\_views.

- *Prefix ALL*

View dengan prefix ALL dapat diakses oleh sembarang user dan biasanya terdapat kolom OWNER. View ini memberikan informasi obyek yang dapat diakses oleh user yang diberi grant public atau explicit dari privilege atau role, termasuk obyek yang dimiliki oleh user tersebut.

- *Prefix DBA*

View dengan prefix DBA memberikan informasi mengenai semua obyek didalam database dan biasanya termasuk kolom OWNER. View ini dapat diakses oleh database administrator atau user yang di grant dengan privilege sistem SELECT ANY TABLE. Agar semua user yang di beri privilege SELECT ANY TABLE dapat mengaksesnya maka dibuat synonym untuk view ini.

Pada tabel berikut ini diberikan beberapa view yang dikelompokkan dalam katagori jenis informasi yang dihasilkan oleh masing-masing view, seperti berikut ini :

<b>View</b>	<b>Keterangan</b>
dictionary dict_columns	Informasi umum (general overview)
dba_tables dba_objects dba_lobs dba_tab_columns dba_constraints	Informasi yang berhubungan dengan obyek user seperti tabel, constraint, obyek dan kolom yang besar.
dba_users dba_sys_privs dba_roles	Informasi tentang privilege dan role user
dba_extents dba_free_space dba_segments	Alokasi ruang memory untuk obyek database
dba_rollback_segs	Struktur database secara umum

dba_data_files dba_tablespaces	
dba_audit_trail dba_audit_objects dba_audit_obj_opts	Informasi auditing

Untuk mendapatkan gambaran tentang view data dictionary, misalkan kolom-kolom yang dimilikinya, dan dynamic performance fiew, maka dapat diquery lewat view DICTIONARY atau DICT\_COLUMNS, seperti contoh berikut ini :

```
SVRMGR> SELECT * FROM dictionary
        2 WHERE table_name LIKE '%TABLE%';
```

```
TABLE_NAME                                COMMENTS
-----
ALL_ALL_TABLES                            Description of all object and relational
                                           tables accessible to the user
ALL_NESTED_TABLES                         Description of nested tables in tables
                                           accessible to the user
ALL_OBJECT_TABLES                         Description of all object tables
                                           accessible to the user
ALL_PART_TABLES
ALL_TABLES                                Description of relational tables
                                           accessible to the user
ALL_UPDATABLE_COLUMNS                     Description of all updatable columns
DBA_ALL_TABLES                             Description of all object and relational
                                           tables in the database
DBA_NESTED_TABLES                         Description of nested tables contained
                                           in all tables
DBA_OBJECT_TABLES                         Description of all object tables in the
                                           database
DBA_PART_TABLES
DBA_QUEUE_TABLES
DBA_TABLES                                Description of all relational tables in
                                           the database
DBA_TABLESPACES                           Description of all tablespaces
```

```
DBA_UPDATABLE_COLUMNS      Description of dba updatable columns
USER_ALL_TABLES            Description of all object and relational
                           tables owned by the user's
USER_NESTED_TABLES        Description of nested tables contained
                           in the user's own tables
USER_OBJECT_TABLES        Description of the user's own object
                           tables
USER_PART_TABLES
USER_QUEUE_TABLES
USER_TABLES                Description of the user's own relational
                           tables
USER_TABLESPACES          Description of accessible tablespaces
USER_UPDATABLE_COLUMNS    Description of updatable columns
TABLE_PRIVILEGES          Grants on objects for which the user is
                           the grantor, grantee, owner, or an
                           enabled role or PUBLIC is the grantee
GV$FIXED_TABLE            Synonym for GV_$FIXED_TABLE
GV$TABLESPACE             Synonym for GV_$TABLESPACE
V$FIXED_TABLE             Synonym for V_$FIXED_TABLE
V$TABLESPACE             Synonym for V_$TABLESPACE
27 rows selected.
```

Catatan : Data dictionary view **DICTIONARY** biasanya dipanggil menggunakan synonymnya yaitu **DICT**, seperti contoh berikut ini :

```
SQL> SELECT column_name, comments
       2 FROM dict_columns
       3 WHERE table_name='DBA_TABLES';
```

```
COLUMN_NAME                COMMENTS
-----
AVG_SPACE                  The average available free
                           space in the table
```

<i>CHAIN_CNT</i>	<i>The number of chained rows in the table</i>
<i>AVG_ROW_LEN</i>	<i>The average row length, including row overhead</i>
<i>AVG_SPACE_FREELIST_BLOCKS</i>	<i>The average freespace of all blocks on a freelist</i>
<i>NUM_FREELIST_BLOCKS</i>	<i>The number of blocks on the freelist</i>
<i>DEGREE</i>	<i>The number of threads per instance for scanning the table</i>
<i>INSTANCES</i>	<i>The number of instances across which the table is to be scanned</i>
<i>CACHE</i>	<i>Whether the table is to be cached in the buffer cache</i>
<i>TABLE_LOCK</i>	<i>Whether table locking is enabled or disabled</i>
<i>SAMPLE_SIZE</i>	<i>The sample size used in analyzing this table</i>
<i>LAST_ANALYZED</i>	<i>The date of the most recent time this table was analyzed</i>
<i>PARTITIONED</i>	<i>Is this table partitioned? YES or NO</i>
<i>IOT_TYPE</i>	<i>If index-only table, then IOT_TYPE is IOT or IOT_OVERFLOW else NULL</i>
<i>TEMPORARY</i>	<i>Can the current session only see data that it place in this object itself?</i>
<i>NESTED</i>	<i>Is the table a nested table?</i>
<i>BUFFER_POOL</i>	<i>The default buffer pool to be used for table blocks</i>
<i>OWNER</i>	<i>Owner of the table</i>
<i>TABLE_NAME</i>	<i>Name of the table</i>
<i>TABLESPACE_NAME</i>	<i>Name of the tablespace containing the table</i>
<i>CLUSTER_NAME</i>	<i>Name of the cluster, if any, to which the table belongs</i>

<i>IOT_NAME</i>	Name of the index-only table, if any, to which the overflow entry belongs
<i>PCT_FREE</i>	Minimum percentage of free space in a block
<i>PCT_USED</i>	Minimum percentage of used space in a block
<i>INI_TRANS</i>	Initial number of transactions
<i>MAX_TRANS</i>	Maximum number of transactions
<i>INITIAL_EXTENT</i>	Size of the initial extent in bytes
<i>NEXT_EXTENT</i>	Size of secondary extents in bytes
<i>MIN_EXTENTS</i>	Minimum number of extents allowed in the segment
<i>MAX_EXTENTS</i>	Maximum number of extents allowed in the segment
<i>PCT_INCREASE</i>	Percentage increase in extent size
<i>FREELISTS</i>	Number of process freelists allocated in this segment
<i>FREELIST_GROUPS</i>	Number of freelist groups allocated in this segment
<i>LOGGING</i>	Logging attribute
<i>BACKED_UP</i>	Has table been backed up since last modification?
<i>NUM_ROWS</i>	The number of rows in the table
<i>BLOCKS</i>	The number of used blocks in the table
<i>EMPTY_BLOCKS</i>	The number of empty (never used) blocks in the table

37 rows selected.



## 5.2. Membuat View Data Dictionary

Pada saat database selesai dibuat, maka script *catalog.sql* dan *catproc.sql* harus dieksekusi oleh user SYS. Pada sistem operasi UNIX, script tersebut berada didalam direktory \$ORACLE\_HOME/rdbms/admin. Sedangkan pada sistem operasi Windows NT/2000 berada didalam direktory %ORACLE\_HOME%\rdbms80\admin.

Fungsi dari masing-masing script dapat dilihat pada tabel berikut ini :

Script	Kegunaan
catalog.sql	Membuat view data dictionary yang umumnya digunakan.
catproc.sql	Menjalankan semua script yang diperlukan PL/SQL pada server.

- *Script catalog.sql*

Script *catalog.sql* digunakan untuk membuat view dari tabel dasar, view pada *dynamic-performance-view* dan synonymnya. Script *catalog.sql* juga menjalankan beberapa script lain yang dipergunakan untuk membuat view dan object lainnya untuk utility Server Manager. Script *catalog.sql* juga menjalankan *standart.sql* yang digunakan untuk membuat lingkungan PL/SQL. Script *standart.sql* mendeklarasi tipe, eksepsi, dan subprogram yang secara otomatis dapat digunakan oleh setiap program PL/SQL. Sebagai contoh script *standart.sql* mendeklarasikan fungsi *built-in* yang disebut BITAND yang digunakan untuk menampilkan hasil dan operasi bit.

```
function BITAND (LEFT binary_integer, RIGH binary_integer)
    return binary_integer;
```

- Script *catproc.sql*

Script *catproc.sql* menghasilkan beberapa fungsi PL/SQL, dan script ini juga membuat beberapa paket PL/SQL yang dipakai untuk mengembangkan fungsi RDBMS. Script *catproc.sql* juga membuat beberapa view tambahan untuk

keperluan opsi antrian lebih lanjut, tablespce untuk recovery *point-in-time*, dan penggunaan beberapa LOB.

### 5.3. Menggunakan Script Administrativ

Direktory \$ORACLE\_HOME/rdbms/admin pada UNIX atau direktory %ORACLE\_HOME%\rdbms80\admin pada Windows NT/2000 berisi script SQL untuk keperluan administrativ, yang dibagi menjadi empat katagori file sesuai dengan konvensi yang diberikan pada nama depan script tersebut, yang ditampilkan pada tabel berikut ini :

Konvensi nama	Keterangan
<i>cat*.sql</i>	Informasi katalog dan data dictionary
<i>dbms*.sql</i>	Spesifikasi paket database
<i>prvt*.plb</i>	Kode paket wrapped database
<i>utl*.sql</i>	View dan tabel untuk utilitas database

- *Script utl\*.sql*

Script utl\*.sql dieksekusi untuk keperluan jika database perlu untuk menambah view atau tabel. Sebagai contoh script utlsamp.sql akan membuat beberapa tabel contoh seperti EMP, DEPT, SALGRADE, dan BONUS, dalam user SCOTT.

- *Script cat\*.sql*

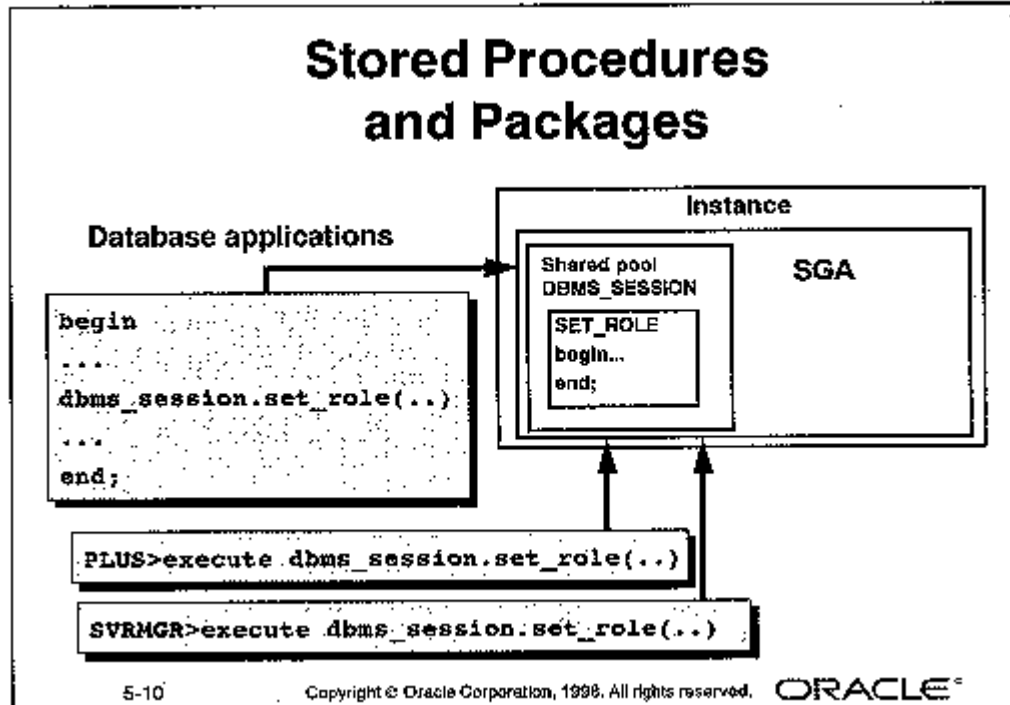
Script cat\*.sql digunakan untuk membuat view data dictionary. Selain script *ctalog.sql* dan *catproc.sql* yang telah dibahas didepan, ada script yang digunakan untuk membuat informasi utilitas Oracle. Sebagai contoh script *catrman.sql* digunakan untuk membuat tabel catalog recovery dan view untuk Recovery Manager. Sedangkan script *catnormn.sql* digunakan untuk men-drop tabel dan view diatas.

- *Script dbms\*.sql dan prvt\*.sql*

Script `dbms*.sql` dan `prvt*.sql` berisi informasi tentang paket Oracle predefined, yang dapat digunakan untuk mengembangkan fungsionalitas Oracle dengan memperlengkapi database fungsi dan *stored-procedure*. Program-program ini banyak digunakan untuk menyederhanakan tugas seorang administrator didalam mengelola database.

Hampir semua script dieksekusi pada saat mengeksekusi script `catproc.sql`, sedangkan beberapa script tambahan harus dieksekusi tersendiri oleh administrator. Sebagai contoh bahwa `dbmspool.sql` untuk membantu agar dapat menampilkan ukuran suatu obyek pada shared pool dan menandainya untuk keperluan menyimpan atau tidak menyimpan dalam tugas mengurangi fragmentasi shared pool.

Catatan : Hampir semua script harus dieksekusi oleh user SYS. Database administrator harus menguji beberapa script untuk menentukan user yang mana yang dapat menjalankan script.



### **5.3. Administrasi Stored Procedure dan Paket**

User dapat menyimpan unit program PL/SQL didalam database dan mengeksekusinya menggunakan Oracle tool seperti SQL\*plus, Server Manager, Enterprise Manager, atau menjalankannya lewat aplikasi Oracle. Stored procedure berisi prosedur dan fungsi. Pekt merupakan suatu unit program PL/SQL.

Store procedure dan paket merupakan obyek database yangdibuat dan dihapus dari schema user dengan perintah CREATE dan DROP. Sebagai contoh procedure DBMS\_SESSION.SET\_ROLE dapat dieksekusi dengan SQL\*Plus, Server Manager dan dari aplikasi Oracle.

Jika telah biasa maka bagi seorang administrator paket sangat membantu untuk melaksanakan tugas-tugas administratif database, disamping memiliki keuntungan antara lain :

- Disimpan dalam share pool sehingga mengurangi waktu yang dibutuhkan untuk membaca dibanding jika disimpan dalam bentuk script pada disk.
- Keamanan data dapat dijaga karena akses yang dilakukan telah ditentukan pada stored prosedur atau fungsi tersebut.
- Untuk eksekusi, beberapa user dapat membagi duplikat prosedur tesebut.