中国数据库联盟
All China Database Union

墨天轮

2023 JUN 30

ACDU China Tour

数据库前沿技术揭秘及应用

中国 · 深圳

Oracle 23c创新特性与SQL增强

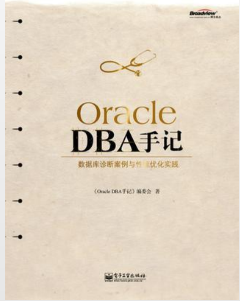
演讲人：杨廷琨

介绍

杨廷琨(yangtingkun)

- □ ACOUG副总裁
- □ 前Oracle ACED
- □ ITPUB数据库管理区版主
- □ 参与编写《Oracle数据库性能优化》、《Oracle DBA手记》、《Oracle DBA手记3》和《Oracle性能优化与诊断案例精选》
- □ 二十三年的一线DBA经验
- □ 个人BLOG中积累了2500篇原创技术文章
- □ 云和恩墨联合创始人兼CTO

ACOUG
All China Oracle User Group
中国 Oracle 用户组



目录

CONTENTS

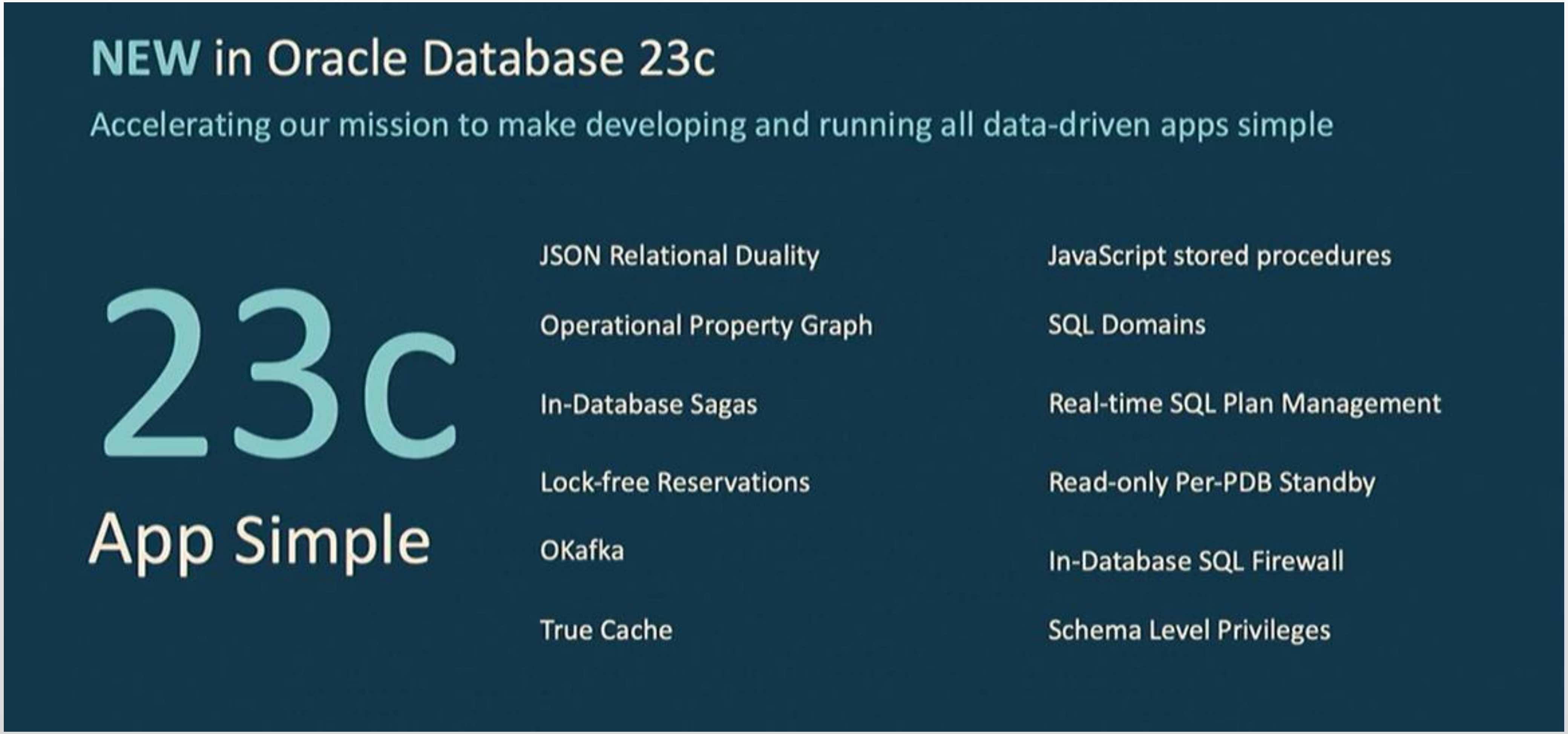
01 Oracle 23c创新特性

02 Oracle 23c SQL新特性

03 Oracle 23c升级策略

Oracle23c 概述

- 23c的主要目标：应用简化



NEW in Oracle Database 23c

Accelerating our mission to make developing and running all data-driven apps simple

23c

App Simple

JSON Relational Duality

Operational Property Graph

In-Database Sagas

Lock-free Reservations

OKafka

True Cache

JavaScript stored procedures

SQL Domains

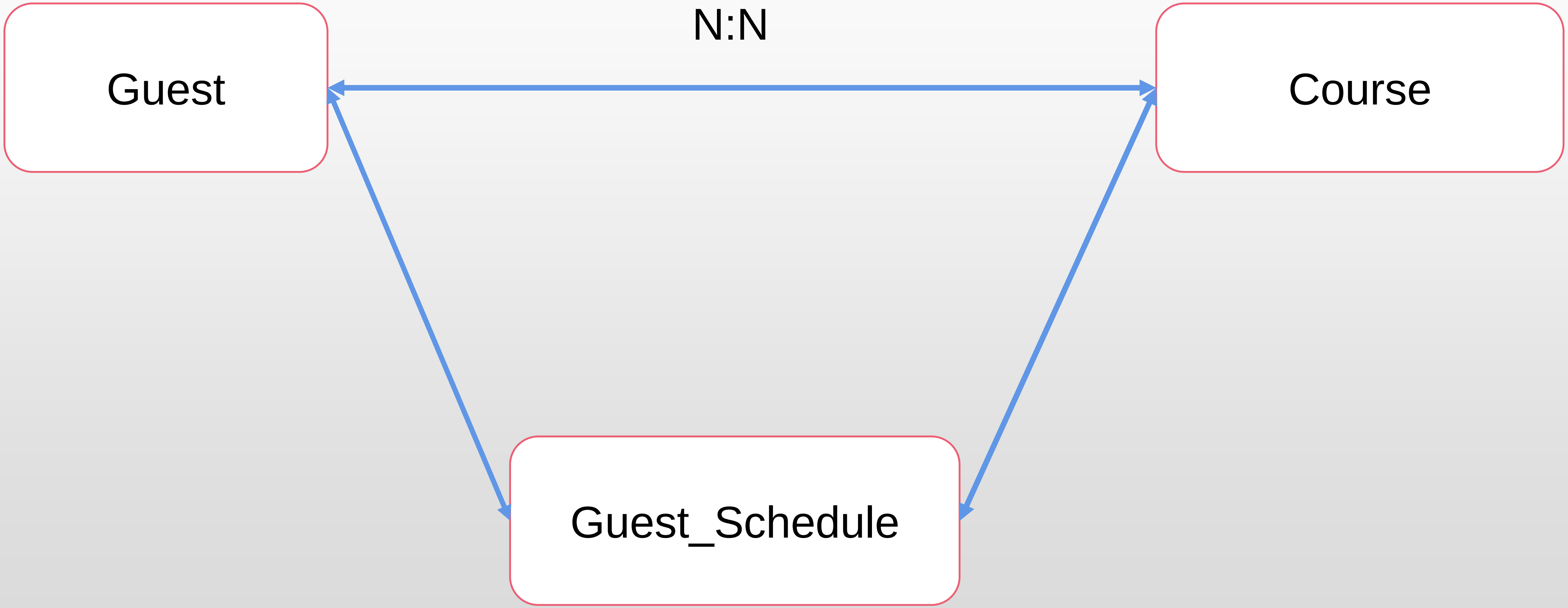
Real-time SQL Plan Management

Read-only Per-PDB Standby

In-Database SQL Firewall

Schema Level Privileges

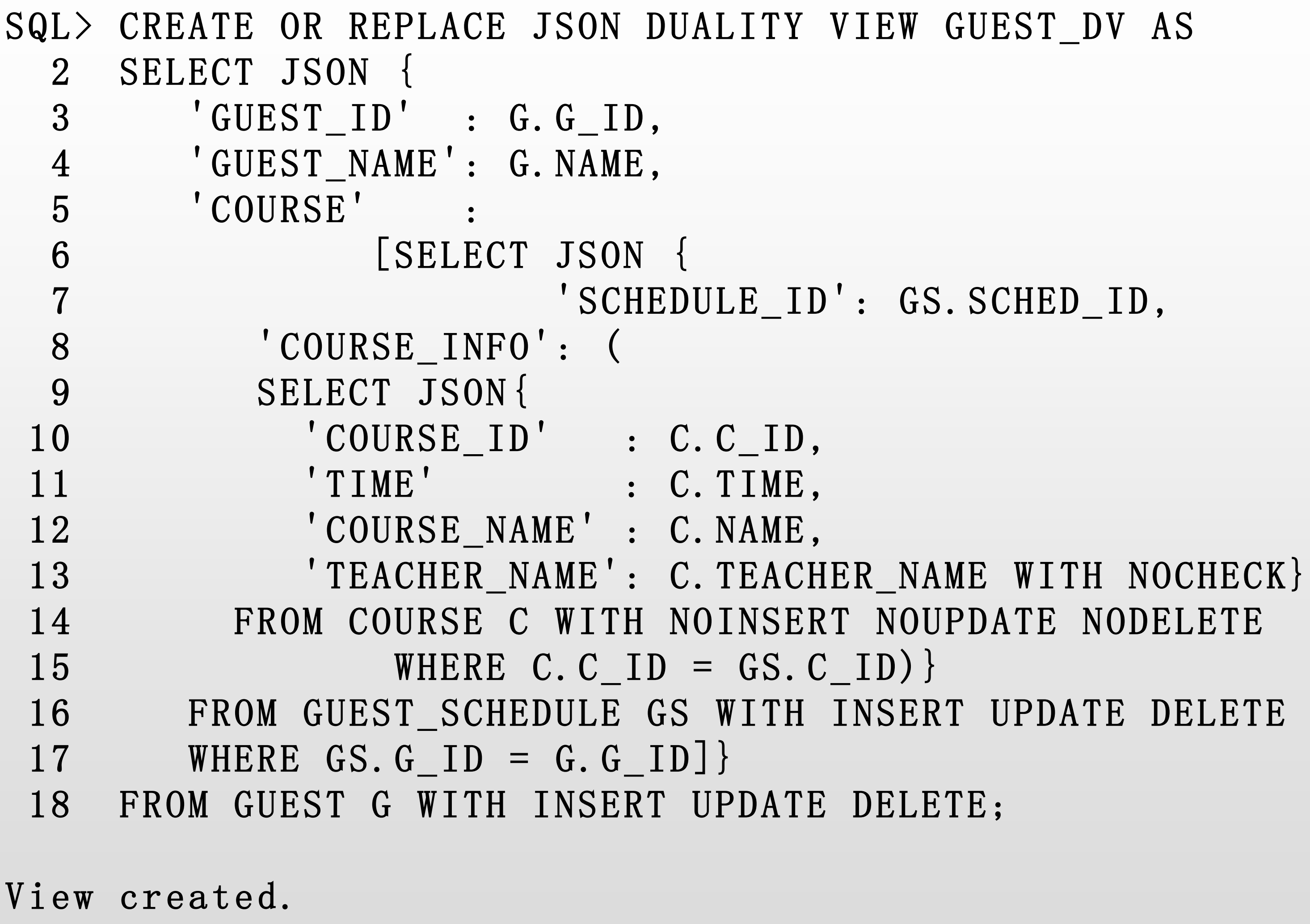
Oracle23c创新特性：JSON RELATIONAL DUALITY



Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> CREATE TABLE GUEST
  2  (G_ID NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY ,
  3  NAME VARCHAR2(30),
  4  CONSTRAINT PK_STUDENT PRIMARY KEY (G_ID));

Table created.

SQL> CREATE TABLE COURSE
  2  (C_ID NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
  3  NAME VARCHAR2(90),
  4  TIME VARCHAR2(30),
  5  TEACHER_NAME VARCHAR2(30),
  6  CONSTRAINT PK_COURSE PRIMARY KEY (C_ID));

Table created.

SQL> CREATE TABLE GUEST_SCHEDULE
  2  (SCHED_ID NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY,
  3  G_ID NUMBER,
  4  C_ID NUMBER,
  5  CONSTRAINT PK_STUD_SCHED PRIMARY KEY (SCHED_ID),
  6  CONSTRAINT FK_GUEST  FOREIGN KEY (G_ID) REFERENCES GUEST(G_ID),
  7  CONSTRAINT FK_COURSE FOREIGN KEY (C_ID) REFERENCES COURSE(C_ID));

Table created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY



```
SQL> CREATE OR REPLACE JSON DUALITY VIEW GUEST_DV AS
  2  SELECT JSON {
  3     'GUEST_ID'  : G.G_ID,
  4     'GUEST_NAME': G.NAME,
  5     'COURSE'    :
  6             [SELECT JSON {
  7                     'SCHEDULE_ID': GS.SCHED_ID,
  8        'COURSE_INFO': (
  9        SELECT JSON{
 10          'COURSE_ID'   : C.C_ID,
 11          'TIME'        : C.TIME,
 12          'COURSE_NAME' : C.NAME,
 13          'TEACHER_NAME': C.TEACHER_NAME WITH NOCHECK}
 14       FROM COURSE C WITH NOINSERT NOUPDATE NODELETE
 15            WHERE C.C_ID = GS.C_ID)}
 16     FROM GUEST_SCHEDULE GS WITH INSERT UPDATE DELETE
 17     WHERE GS.G_ID = G.G_ID]}
 18  FROM GUEST G WITH INSERT UPDATE DELETE;

View created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> CREATE OR REPLACE JSON DUALITY VIEW COURSE_DV AS
  2  SELECT JSON {
  3     'COURSE_ID'    : C.C_ID,
  4     'TIME'         : C.TIME,
  5     'COURSE_NAME'  : C.NAME,
  6     'TEACHER_NAME' : C.TEACHER_NAME}
  7     FROM COURSE C WITH INSERT UPDATE DELETE;

View created.

SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20001,
  3             "TIME"         : "2023-06-30 14:00:00",
  4             "COURSE_NAME"  : "新时代下数据库运维和DBA面临的挑战和机遇",
  5             "TEACHER_NAME" : "徐戟" }');

1 row created.

SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20002,
  3             "TIME"         : "2023-06-30 14:30:00",
  4             "COURSE_NAME"  : "Greenplum Database 7 性能提升的秘密武器",
  5             "TEACHER_NAME" : "汤韬" }');

1 row created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20003,
  3            "TIME"           : "2023-06-30 15:00:00",
  4            "COURSE_NAME"    : "MySQL 8.0 新特性解读",
  5            "TEACHER_NAME"   : "姜承尧" }');

1 row created.

SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20004,
  3            "TIME"           : "2023-06-30 15:30:00",
  4            "COURSE_NAME"    : "基于PostgreSQL内核增强和扩展的产品设计哲学",
  5            "TEACHER_NAME"   : "赵伟" }');

1 row created.

SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20005,
  3            "TIME"           : "2023-06-30 16:00:00",
  4            "COURSE_NAME"    : "深算院全自研数据库系统的设计与实践",
  5            "TEACHER_NAME"   : "王海峰" }');

1 row created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20006,
  3             "TIME"         : "2023-06-30 16:30:00",
  4             "COURSE_NAME"  : "openGauss 5.0.0的新特性解读",
  5             "TEACHER_NAME" : "熊小军" }');

1 row created.

SQL> INSERT INTO COURSE_DV
  2  VALUES ('{ "COURSE_ID"    : 20007,
  3             "TIME"         : "2023-06-30 17:00:00",
  4             "COURSE_NAME"  : "Oracle 23c创新技术与SQL增强",
  5             "TEACHER_NAME" : "杨廷琨" }');

1 row created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> INSERT INTO GUEST_DV VALUES ('
     {"GUEST_ID"   : 100000,
      "GUEST_NAME" : "墨天轮",
      "COURSE"     :
        [{"SCHEDULE_ID" : 101,
         "COURSE_INFO" : {"COURSE_ID"    : 20001,
                          "TIME"         : "2023-06-30 14:00:00",
                          "COURSE_NAME"  : "新时代下数据库运维和DBA面临的挑战和机遇",
                          "TEACHER_NAME" : "徐戟" }},
        {"SCHEDULE_ID" : 102,
         "COURSE_INFO" : {"COURSE_ID"    : 20005,
                          "TIME"         : "2023-06-30 16:00:00",
                          "COURSE_NAME"  : "深算院全自研数据库系统的设计与实践",
                          "TEACHER_NAME" : "王海峰" }}]}');

1 row created.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> SELECT * FROM GUEST;

      G_ID NAME
---------- ---------------------------------------------
    100000 墨天轮

SQL> SELECT NAME, TEACHER_NAME TEACHER, SUBSTR(TIME, 1, 16) TIME FROM COURSE;

NAME                                          TEACHER    TIME
--------------------------------------------- ---------- ------------------------------
新时代下数据库运维和DBA面临的挑战和机遇        徐戟       2023-06-30 14:00
Greenplum Database 7 性能提升的秘密武器       汤韬       2023-06-30 14:30
MySQL 8.0 新特性解读                          姜承尧     2023-06-30 15:00
基于PostgreSQL内核增强和扩展的产品设计哲学    赵伟       2023-06-30 15:30
深算院全自研数据库系统的设计与实践            王海峰     2023-06-30 16:00
openGauss 5.0.0的新特性解读                   熊小军     2023-06-30 16:30
Oracle 23c创新技术与SQL增强                   杨廷琨     2023-06-30 17:00

SQL> SELECT * FROM GUEST_SCHEDULE;

  SCHED_ID       G_ID       C_ID
---------- ---------- ----------
       101     100000      20001
       102     100000      20005
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> SELECT REPLACE(JSON_QUERY(DATA, '$.COURSE'), ',', ',' || CHR(10)) FROM GUEST_DV;

REPLACE(JSON_QUERY(DATA,'$.COURSE'),',',','||CHR(10))
--------------------------------------------------------------------
[{"SCHEDULE_ID":101,
"COURSE_INFO":{"COURSE_ID":20001,
"TIME":"2023-06-30 14:00:00",
"COURSE_NAME":"新时代下数据库运维和DBA面临的挑战和机遇",
"TEACHER_NAME":"徐戟"}},
{"SCHEDULE_ID":102,
"COURSE_INFO":{"COURSE_ID":20005,
"TIME":"2023-06-30 16:00:00",
"COURSE_NAME":"深算院全自研数据库系统的设计与实践",
"TEACHER_NAME":"王海峰"}}]

SQL> INSERT INTO GUEST_SCHEDULE VALUES (103, 100000, 20007);

1 row created.

SQL> COMMIT;

Commit complete.
```

Oracle23c创新特性：JSON RELATIONAL DUALITY

```
SQL> SELECT REPLACE(JSON_QUERY(DATA, '$.COURSE'), ',', ',' || CHR(10)) FROM GUEST_DV;

REPLACE(JSON_QUERY(DATA,'$.COURSE'),',',','||CHR(10))
---------------------------------------------------------------------------------------------
----------------------------------------------------------------------
[{"SCHEDULE_ID":101,
"COURSE_INFO":{"COURSE_ID":20001,
"TIME":"2023-06-30 14:00:00",
"COURSE_NAME":"新时代下数据库运维和DBA面临的挑战和机遇",
"TEACHER_NAME":"徐戟"}},
{"SCHEDULE_ID":102,
"COURSE_INFO":{"COURSE_ID":20005,
"TIME":"2023-06-30 16:00:00",
"COURSE_NAME":"深算院全自研数据库系统的设计与实践",
"TEACHER_NAME":"王海峰"}},
{"SCHEDULE_ID":103,
"COURSE_INFO":{"COURSE_ID":20007,
"TIME":"2023-06-30 17:00:00",
"COURSE_NAME":"Oracle 23c创新技术与SQL增强",
"TEACHER_NAME":"杨廷琨"}}]
```

目录

CONTENTS

01 Oracle 23c创新特性

02 Oracle 23c SQL新特性

03 Oracle 23c升级策略

Oracle23c新特性

- 查询省略FROM
- 表支持4096列
- Boolean类型
- GROUP BY别名
- SCHEMA级授权
- DDL支持EXISTS
- 构建多行记录
- 更新JOIN结果
- RETURN支持OLD
- SQL DOMAINS
- 无锁列值托管
- 自动事务终止

Oracle23c新特性：查询省略FROM

```
SQL> select banner_full from v$version;

BANNER_FULL
------------------------------------------------------------------------------------
Oracle Database 23c Free, Release 23.0.0.0.0 - Developer-Release
Version 23.2.0.0.0

SQL> select 100;

       100
----------
       100

SQL> select to_char(sysdate, 'yyyy-mm-dd');

TO_CHAR(SY
----------
2023-02-02

-----------------------------------------------------------------
| Id  | Operation        | Name | Rows  | Cost (%CPU)| Time     |
-----------------------------------------------------------------
|   0 | SELECT STATEMENT |      |     1 |     2   (0)| 00:00:01 |
|   1 |  FAST DUAL       |      |     1 |     2   (0)| 00:00:01 |
-----------------------------------------------------------------
```

Oracle23c新特性：表支持4096列

```
SQL> CREATE TABLE T_1001_COLUMNS (
  2  C0001 NUMBER,
  3  C0002 NUMBER,

.
.
.
999  C0998 NUMBER,
1000  C0999 NUMBER,
1001  C1000 NUMBER,
1002  C1001 NUMBER);
C1001 NUMBER)
*
ERROR at line 1002:
ORA-01792: maximum number of columns in a table or view is 1000

SQL> SHOW PARAMETER MAX_COLUMNS

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
max_columns                          string      STANDARD
```

Oracle23c新特性：表支持4096列

```
SQL> ALTER SYSTEM SET MAX_COLUMNS = EXTENDED;

System altered.

SQL> CREATE TABLE T_1001_COLUMNS (
  2  C0001 NUMBER,
  3  C0002 NUMBER,
  4  C0003 NUMBER,
.
.
.
999  C0998 NUMBER,
1000  C0999 NUMBER,
1001  C1000 NUMBER,
1002  C1001 NUMBER);

Table created.

SQL> SELECT COUNT(*) FROM USER_TAB_COLUMNS WHERE TABLE_NAME = 'T_1001_COLUMNS';

  COUNT(*)
----------
      1001
```

Oracle23c新特性：Boolean类型

```
SQL> CREATE TABLE T_BOOL (ID NUMBER, BOOL BOOLEAN);

Table created.

SQL> INSERT INTO T_BOOL VALUES (1, TRUE);

1 row created.

SQL> INSERT INTO T_BOOL VALUES (2, FALSE);

1 row created.

SQL> INSERT INTO T_BOOL VALUES (3, NULL);

1 row created.

SQL> INSERT INTO T_BOOL VALUES (4, 'T');

1 row created.

SQL> INSERT INTO T_BOOL VALUES (5, 0);

1 row created.
```

Oracle23c新特性：Boolean类型

```
SQL> SELECT ID FROM T_BOOL WHERE BOOL;

        ID
----------
         1
         4

SQL> SELECT * FROM T_BOOL;

        ID BOOL
---------- -----------
         1 TRUE
         2 FALSE
         3
         4 TRUE
         5 FALSE
```

Oracle23c新特性：GROUP BY别名

```
SQL> create table t as select rownum id, a.* from dba_objects a;

Table created.

SQL> SELECT TRUNC(CREATED, 'MM') MON, COUNT(*) FROM T
  2  GROUP BY TRUNC(CREATED, 'MM');

MON               COUNT(*)
------------ ----------
01-OCT-22         76378
01-DEC-22          5289
01-JAN-23         10164
01-FEB-23           683

SQL> SELECT TRUNC(CREATED, 'MM') MON, COUNT(*) FROM T
  2  GROUP BY MON;

MON               COUNT(*)
------------ ----------
01-OCT-22         76378
01-DEC-22          5289
01-JAN-23         10164
01-FEB-23           683
```

Oracle23c新特性：SCHEMA级授权

```
SQL> CREATE USER YANGTK_SELE IDENTIFIED BY SELECTONLY DEFAULT TABLESPACE USERS;

User created.

SQL> GRANT CREATE SESSION TO YANGTK_SELE;

Grant succeeded.

SQL> GRANT SELECT ANY TABLE ON SCHEMA YANGTK TO YANGTK_SELE;

Grant succeeded.

SQL> CONN YANGTK_SELE/SELECTONLY@pdb1
Connected.
SQL> SELECT * FROM SESSION_PRIVS;

PRIVILEGE
----------------------------------------
CREATE SESSION

SQL> SELECT * FROM SESSION_ROLES;

no rows selected
```

Oracle23c新特性：SCHEMA级授权

```
SQL> SELECT OWNER, TABLE_NAME FROM ALL_TABLES WHERE OWNER NOT IN ('SYS', 'SYSTEM', 'XDB',
'MDSYS', 'CTXSYS');

OWNER                          TABLE_NAME
------------------------------ ------------------------------
YANGTK                         STUDENT_SCHEDULER
YANGTK                         T
YANGTK                         T_1001_COLUMNS

SQL> SELECT TABLE_NAME, PRIVILEGE FROM USER_TAB_PRIVS;

TABLE_NAME                     PRIVILEGE
------------------------------ ----------------------------------------
YANGTK_SELE                    INHERIT PRIVILEGES

SQL> SELECT COUNT(*) FROM YANGTK.T;

  COUNT(*)
----------
     92514
```

Oracle23c新特性：SCHEMA级授权

```
SQL> CONN YANGTK/yangtk@pdb1
Connected.
SQL> CREATE TABLE T_NEW (ID NUMBER);

Table created.

SQL> CONN YANGTK_SELE/SELECTONLY@pdb1
Connected.
SQL> SELECT OWNER, TABLE_NAME FROM ALL_TABLES WHERE OWNER NOT IN ('SYS', 'SYSTEM', 'XDB',
'MDSYS', 'CTXSYS');

OWNER      TABLE_NAME
---------- --------------------
YANGTK     STUDENT_SCHEDULER
YANGTK     T
YANGTK     T_1001_COLUMNS
YANGTK     T_NEW
```

Oracle23c新特性：DDL支持EXISTS

```
SQL> SELECT * FROM TAB;

TNAME                                              TABTYPE        CLUSTERID
-------------------------------------------------- ------------- ----------
STUDENT_SCHEDULER                                  TABLE
T                                                  TABLE
T_1001_COLUMNS                                     TABLE
T_NEW                                              TABLE
T_BOOL                                             TABLE

SQL> DROP TABLE T_NOTEXIST;
DROP TABLE T_NOTEXIST
           *
ERROR at line 1:
ORA-00942: table or view does not exist


SQL> DROP TABLE IF EXISTS T_NOTEXIST;

Table dropped.

SQL> DROP TABLE IF EXISTS T_NEW;

Table dropped.
```

Oracle23c新特性：DDL支持EXISTS

```
SQL> SELECT * FROM TAB;

TNAME                                              TABTYPE        CLUSTERID
-------------------------------------------------- ------------- ----------
STUDENT_SCHEDULER                                  TABLE
T                                                  TABLE
T_1001_COLUMNS                                     TABLE
T_BOOL                                             TABLE
BIN$87V21hTqU+jgU6YWFKyjyQ==$0                     TABLE

SQL> CREATE TABLE T (ID NUMBER);
CREATE TABLE T (ID NUMBER)
             *
ERROR at line 1:
ORA-00955: name is already used by an existing object

SQL> CREATE TABLE IF NOT EXISTS T (ID NUMBER);

Table created.
```

Oracle23c新特性：构建多行记录

```
SQL> WITH S (A, B, C) AS (
  2  VALUES (1, 'A', SYSDATE),
  3  (2, 'B', TO_DATE('20230202', 'YYYYMMDD')),
  4  (3, 'ABCD', SYSDATE - 1))
  5  SELECT A, B, TO_CHAR(C, 'YYYYMMDD') FROM S;

         A B    TO_CHAR(
---------- ---- --------
         1 A    20230202
         2 B    20230202
         3 ABCD 20230201

-------------------------------------------------------------------------
| Id  | Operation        | Name | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------
|   0 | SELECT STATEMENT |      |    18E|    15E|     6   (0)| 00:00:01 |
|   1 |  VIEW            |      |    18E|    15E|     6   (0)| 00:00:01 |
|   2 |   VALUES SCAN    |      |       |       |            |          |
-------------------------------------------------------------------------
```

Oracle23c新特性：构建多行记录

```
SQL> CREATE TABLE T_MULROW (ID NUMBER, NAME VARCHAR2(30));

Table created.

SQL> INSERT INTO T_MULROW VALUES (1, 'A'), (2, 'B'), (3, 'ABC');

3 rows created.

------------------------------------------------------------------------------
| Id  | Operation                  | Name     | Rows  | Cost (%CPU)| Time     |
------------------------------------------------------------------------------
|   0 | INSERT STATEMENT           |          |    18E|     6   (0)| 00:00:01 |
|   1 |  LOAD TABLE CONVENTIONAL   | T_MULROW |       |            |          |
|   2 |   VALUES SCAN              |          |       |            |          |
------------------------------------------------------------------------------
```

Oracle23c新特性：更新JOIN结果

```
SQL> SELECT * FROM T_BOOL;

        ID BOOL
---------- -----------
         1 TRUE
         2 FALSE
         3
         4 TRUE
         5 FALSE

SQL> SELECT * FROM T_MULROW;

        ID NAME
---------- ------------------------------
         1 A
         2 B
         3 ABC
```

Oracle23c新特性：更新JOIN结果

```
SQL> UPDATE T_MULROW SET NAME = LOWER(NAME)
  2  FROM T_BOOL
  3  WHERE T_MULROW.ID = T_BOOL.ID
  4  AND BOOL;

1 row updated.

-------------------------------------------------------------------------------
| Id  | Operation           | Name     | Rows  | Bytes | Cost (%CPU)| Time     |
-------------------------------------------------------------------------------
|   0 | UPDATE STATEMENT    |          |     3 |    30 |     6   (0)| 00:00:01 |
|   1 |  UPDATE             | T_MULROW |       |       |            |          |
|*  2 |   HASH JOIN         |          |     3 |    30 |     6   (0)| 00:00:01 |
|*  3 |    TABLE ACCESS FULL| T_BOOL   |     3 |    12 |     3   (0)| 00:00:01 |
|   4 |    TABLE ACCESS FULL| T_MULROW |     3 |    18 |     3   (0)| 00:00:01 |
-------------------------------------------------------------------------------

SQL> SELECT * FROM T_MULROW;

        ID NAME
---------- ------------------------------
         1 a
         2 B
         3 ABC
```

Oracle23c新特性：更新JOIN结果

```
SQL> ALTER TABLE T_BOOL ADD PRIMARY KEY (ID);

Table altered.

SQL> ALTER TABLE T_MULROW ADD CONSTRAINT FK_ID FOREIGN KEY (ID) REFERENCES T_BOOL;

Table altered.

SQL> UPDATE
  2  (SELECT M.NAME, M.ID MID, B.ID BID FROM T_MULROW M, T_BOOL B WHERE M.ID = B.ID)
  3  SET NAME = LOWER(NAME)
  4  WHERE MID = BID;

3 rows updated.

--------------------------------------------------------------------------------
| Id  | Operation          | Name     | Rows  | Bytes | Cost (%CPU)| Time     |
--------------------------------------------------------------------------------
|   0 | UPDATE STATEMENT   |          |     3 |    18 |     3   (0)| 00:00:01 |
|   1 |  UPDATE            | T_MULROW |       |       |            |          |
|*  2 |   TABLE ACCESS FULL| T_MULROW |     3 |    18 |     3   (0)| 00:00:01 |
--------------------------------------------------------------------------------
```

Oracle23c新特性：RETURN支持OLD

```
SQL> SET SERVEROUT ON SIZE 100000
SQL> DECLARE
  2     V_NAME_O VARCHAR2(30);
  3     V_NAME_N VARCHAR2(30);
  4  BEGIN
  5     UPDATE T_MULROW SET NAME = UPPER(NAME) WHERE ID = 1
  6     RETURN OLD NAME, NEW NAME
  7     INTO V_NAME_O, V_NAME_N;
  8     DBMS_OUTPUT.PUT_LINE('OLD NAME: ' || V_NAME_O);
  9     DBMS_OUTPUT.PUT_LINE('NEW NAME: ' || V_NAME_N);
  10    COMMIT;
  11  END;
  12  /
OLD NAME: a
NEW NAME: A

PL/SQL procedure successfully completed.
```

Oracle23c新特性：SQL DOMAINS

```
SQL> CREATE DOMAIN ID_NUMBER AS VARCHAR2(18)
  2  CONSTRAINT C_LEN CHECK (LENGTH(ID_NUMBER) = 18
  3   AND LTRIM(SUBSTR(ID_NUMBER, 1, 17), '0123456789') IS NULL
  4   AND SUBSTR(ID_NUMBER, 18) IN ('X', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9')
  5   AND MOD(SUBSTR(ID_NUMBER,1,1)*7 + SUBSTR(ID_NUMBER,2,1)*9 + SUBSTR(ID_NUMBER,3,1)*10
  6   + SUBSTR(ID_NUMBER,4,1)*5 + SUBSTR(ID_NUMBER,5,1)*8 + SUBSTR(ID_NUMBER,6,1)*4
  7   + SUBSTR(ID_NUMBER,7,1)*2 + SUBSTR(ID_NUMBER,8,1) + SUBSTR(ID_NUMBER,9,1)*6
  8   + SUBSTR(ID_NUMBER,10,1)*3 + SUBSTR(ID_NUMBER,11,1)*7 + SUBSTR(ID_NUMBER,12,1)*9
  9   + SUBSTR(ID_NUMBER,13,1)*10 + SUBSTR(ID_NUMBER,14,1)*5 + SUBSTR(ID_NUMBER,15,1)*8
 10   + SUBSTR(ID_NUMBER,16,1)*4 + SUBSTR(ID_NUMBER,17,1)*2, 11) =
 11   MOD(12 - CASE (SUBSTR(ID_NUMBER, 18)) WHEN 'X' THEN 10 ELSE TO_NUMBER(SUBSTR(ID_NUMBER,
18)) END, 11))
 12  DISPLAY SUBSTR(ID_NUMBER, 1, 10) || '****' || SUBSTR(ID_NUMBER, 15)
 13  ORDER TO_NUMBER(SUBSTR(ID_NUMBER, 7, 11));

Domain created.

SQL> CREATE TABLE T_IDEN (ID NUMBER, NAME VARCHAR2(30), IDEN VARCHAR2(18) DOMAIN ID_NUMBER);

Table created.
```

Oracle23c新特性：SQL DOMAINS

```
SQL> INSERT INTO T_IDEN VALUES (1, 'A', '110101198001010010');

1 row created.

SQL> INSERT INTO T_IDEN VALUES (2, 'B', '220381197001010014');

1 row created.

SQL> INSERT INTO T_IDEN VALUES (3, 'C', '33010519900101002X');

1 row created.

SQL> INSERT INTO T_IDEN VALUES (4, 'D', '330105199001010021');
INSERT INTO T_IDEN VALUES (4, 'D', '330105199001010021')
*
ERROR at line 1:
ORA-02290: check constraint (YANGTK.SYS_C008298) violated


SQL> COMMIT;

Commit complete.
```

Oracle23c新特性：SQL DOMAINS

```
SQL> SELECT ID, NAME, IDEN, DOMAIN_DISPLAY(IDEN) D_ID
  2  FROM T_IDEN ORDER BY IDEN;

        ID NAME                 IDEN               D_ID
---------- -------------------- ------------------ --------------------------------
         1 A                    110101198001010010 1101011980****0010
         2 B                    220381197001010014 2203811970****0014
         3 C                    33010519900101002X 3301051990****002X

SQL> SELECT ID, NAME, IDEN, DOMAIN_DISPLAY(IDEN) D_ID
  2  FROM T_IDEN ORDER BY DOMAIN_ORDER(IDEN);

        ID NAME                 IDEN               D_ID
---------- -------------------- ------------------ --------------------------------
         2 B                    220381197001010014 2203811970****0014
         1 A                    110101198001010010 1101011980****0010
         3 C                    33010519900101002X 3301051990****002X
```

Oracle23c新特性：无锁列值托管

```
SQL> CREATE TABLE T_ESCROW (ID NUMBER PRIMARY KEY, ESC_LOCK NUMBER ESCROW, NORMAL_LOCK NUMBER);

Table created.

SQL> INSERT INTO T_ESCROW VALUES (1, 1, 1), (2, 2, 2), (3, 3, 3);

3 rows created.

SQL> COMMIT;

Commit complete.

SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          1           1
         2          2           2
         3          3           3
```

Oracle23c新特性：无锁列值托管

```
SQL> UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK + 1 WHERE ID = 1;

1 row updated.

--sqlplus登陆新的会话，用SQL标识符SQL2>来表示第二个会话进行的操作
SQL> SET SQLP 'SQL2> '
SQL2> UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK - 1 WHERE ID = 1;

1 row updated.

SQL2> COMMIT;

Commit complete.

SQL> COMMIT;

Commit complete.
```

Oracle23c新特性：无锁列值托管

```
SQL> UPDATE T_ESCROW SET NORMAL_LOCK = NORMAL_LOCK + 1 WHERE ID = 1;

1 row updated.

SQL> SELECT L.SID, L.TYPE, ID1, LMODE, CTIME, BLOCK FROM V$LOCK L, V$SESSION S
  2  WHERE L.SID = S.SID AND S.USERNAME = USER AND L.TYPE IN ('TX', 'TM');

       SID TY        ID1      LMODE      CTIME      BLOCK
---------- -- ---------- ---------- ---------- ----------
       506 TM     119484          3         13          2
       506 TX    1376279          6         13          2

SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          1           2
         2          2           2
         3          3           3

SQL> COMMIT;

Commit complete.
```

Oracle23c新特性：无锁列值托管

```
SQL> UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK + 1 WHERE ID = 1;

1 row updated.

SQL> SELECT L.SID, L.TYPE, ID1, LMODE, CTIME, BLOCK FROM V$LOCK L, V$SESSION S
  2  WHERE L.SID = S.SID AND S.USERNAME = USER AND L.TYPE IN ('TX', 'TM');

       SID TY        ID1      LMODE      CTIME      BLOCK
---------- -- ---------- ---------- ---------- ----------
       506 TX    1376285          6         19          2
       506 TM     119485          3         19          2
       506 TM     119484          3         19          2

SQL> SELECT OWNER, OBJECT_NAME FROM DBA_OBJECTS WHERE OBJECT_ID IN (119484, 119485);

OWNER                          OBJECT_NAME
------------------------------ ------------------------------
YANGTK                         T_ESCROW
YANGTK                         SYS_ESCROWJRNL_119484
```

Oracle23c新特性：无锁列值托管

```
SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          1           2
         2          2           2
         3          3           3

SQL> COMMIT;

Commit complete.

SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          2           2
         2          2           2
         3          3           3
```

Oracle23c新特性：无锁列值托管

```
SQL> UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK - 1 WHERE ID = 1;

1 row updated.

SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          2           2
         2          2           2
         3          3           3

SQL> ROLLBACK;

Rollback complete.

SQL> SELECT * FROM T_ESCROW;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          2           2
         2          2           2
         3          3           3
```

Oracle23c新特性：无锁列值托管

```
SQL> select * from t_escrow;

        ID   ESC_LOCK NORMAL_LOCK
---------- ---------- -----------
         1          2           2
         2          2           2
         3          3           3

SQL> select * from SYS_ESCROWJRNL_119484;

no rows selected

SQL> update t_escrow set esc_lock = esc_lock - 1 where id = 1;

1 row updated.

SQL> select * from SYS_ESCROWJRNL_119484;

ORA_SAGA_I ORA_TXN_ID$      ORA_STATUS$  ORA_STMT_TYPE$        ID ESC_LOC ESC_LOCK_RESERVED
---------- ---------------- ------------ ---------------- ------- ------- -----------------
           1500130070AA0100 ACTIVE       UPDATE                 1 -                       1
```

Oracle23c新特性：无锁列值托管

```
SQL> update t_escrow set esc_lock = esc_lock + 3 where id = 1;

1 row updated.

SQL> select * from SYS_ESCROWJRNL_119484;

ORA_SAGA_I ORA_TXN_ID$      ORA_STATUS$  ORA_STMT_TYPE$         ID ESC_LOC ESC_LOCK_RESERVED
---------- ---------------- ------------ ---------------- ------- ------- -----------------
           1500130070AA0100 ACTIVE       UPDATE                 1 -                       1
           1500130070AA0100 ACTIVE       UPDATE                 1 +                       3

SQL> commit;

Commit complete.

SQL> select * from SYS_ESCROWJRNL_119484;

no rows selected
```

Oracle23c新特性：无锁列值托管

```
SQL> CREATE TABLE T_ESCROW (ID NUMBER, ESC_LOCK VARCHAR2(30) ESCROW, NORMAL_LOCK NUMBER);
CREATE TABLE T_ESCROW (ID NUMBER, ESC_LOCK VARCHAR2(30) ESCROW, NORMAL_LOCK NUMBER)
                                                              *
ERROR at line 1:
ORA-55748: Escrow column is supported only on columns of types Oracle NUMBER, INTEGER, and
FLOAT.

SQL> CREATE TABLE T_ESCROW (ID NUMBER, ESC_LOCK NUMBER ESCROW, NORMAL_LOCK NUMBER);
CREATE TABLE T_ESCROW (ID NUMBER, ESC_LOCK NUMBER ESCROW, NORMAL_LOCK NUMBER)
*
ERROR at line 1:
ORA-55728: Escrow columns can only be specified on tables with a primary key.

SQL> UPDATE T_ESCROW SET ESC_LOCK = 0 WHERE ID = 1;
UPDATE T_ESCROW SET ESC_LOCK = 0 WHERE ID = 1
       *
ERROR at line 1:
ORA-55782: Operation is not supported on escrow columns.

SQL> UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK + 1 WHERE NORMAL_LOCK = 3;
UPDATE T_ESCROW SET ESC_LOCK = ESC_LOCK + 1 WHERE NORMAL_LOCK = 3
       *
ERROR at line 1:
ORA-55732: Escrow update should specify all the primary key columns
```

Oracle23c新特性：无锁列值托管

```
SQL> DROP TABLE T_ESCROW PURGE;
DROP TABLE T_ESCROW PURGE
           *
ERROR at line 1:
ORA-55764: Cannot drop/move an escrow table. First run alter table <table_name> modify
(<escrow_column_name> NOT ESCROW) and then drop/move the escrow table


SQL> ALTER TABLE T_ESCROW MODIFY (ESC_LOCK NOT ESCROW);

Table altered.

SQL> DROP TABLE T_ESCROW PURGE;

Table dropped.
```

Oracle23c新特性：自动事务终止

```
SQL> SHOW PARAMETER TXN

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
global_txn_processes                 integer     1
txn_high_priority_wait_target        integer     2147483647
txn_medium_priority_wait_target      integer     2147483647
txn_priority                         string      HIGH
SQL> ALTER SESSION SET TXN_PRIORITY = MEDIUM;

Session altered.

SQL> UPDATE T_BOOL SET BOOL = FALSE WHERE ID = 3;

1 row updated.

SQL> SELECT XID, STATUS, USED_UREC, START_TIME, TXN_PRIORITY, TXN_PRIORITY_WAIT_TARGET FROM
V$TRANSACTION;

XID              STATUS  USED_UREC START_TIME         TXN_PRI TXN_PRIORITY_WAIT_TARGET
---------------- ------  --------- ------------------ ------- ------------------------
0C0010004E0C0000 ACTIVE          1 02/14/23 16:44:07   MEDIUM                        0
```

Oracle23c新特性：自动事务终止

```
SQL2> ALTER SYSTEM SET TXN_HIGH_PRIORITY_WAIT_TARGET = 10;

System altered.

SQL2> SHOW PARAMETER TXN

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
txn_high_priority_wait_target        integer     10
txn_medium_priority_wait_target      integer     2147483647
txn_priority                         string      HIGH
SQL2> SET TIMING ON TIME ON
16:44:18 SQL2> UPDATE T_BOOL SET BOOL = TRUE WHERE ID = 3;

1 row updated.

Elapsed: 00:00:10.68
16:44:36 SQL2> SELECT XID, STATUS, USED_UREC, START_TIME, TXN_PRIORITY,
TXN_PRIORITY_WAIT_TARGET FROM V$TRANSACTION;

XID              STATUS  USED_UREC START_TIME           TXN_PRI TXN_PRIORITY_WAIT_TARGET
---------------- ------ ---------- -------------------- ------- ------------------------
15000B00311C0000 ACTIVE          1 02/14/23 16:44:25    HIGH                          10
```

Oracle23c新特性：自动事务终止

```
SQL> SELECT XID, STATUS, USED_UREC, START_TIME, TXN_PRIORITY, TXN_PRIORITY_WAIT_TARGET FROM
V$TRANSACTION;
SELECT XID, STATUS, USED_UREC, START_TIME, TXN_PRIORITY, TXN_PRIORITY_WAIT_TARGET FROM
V$TRANSACTION
*
ERROR at line 1:
ORA-03113: end-of-file on communication channel
Process ID: 2684285
Session ID: 1229 Serial number: 33384
```

目录
CONTENTS

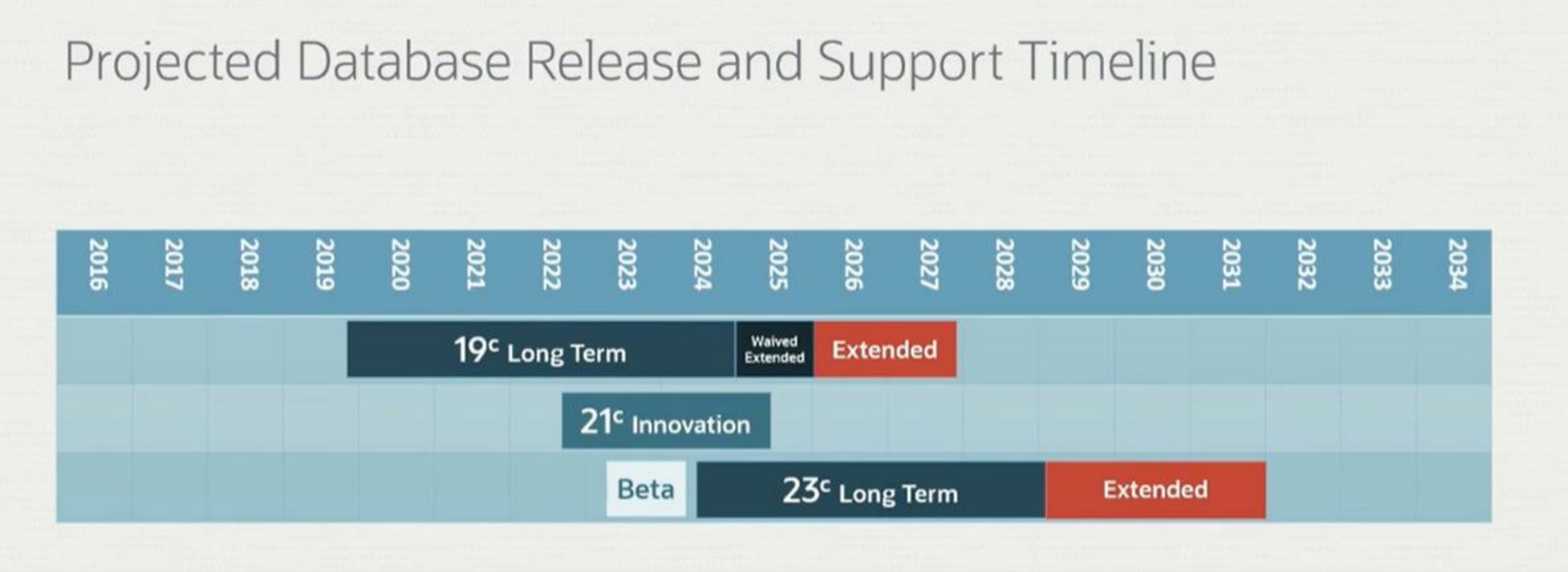
01 Oracle 23c创新特性

02 Oracle 23c SQL新特性

03 Oracle 23c升级策略

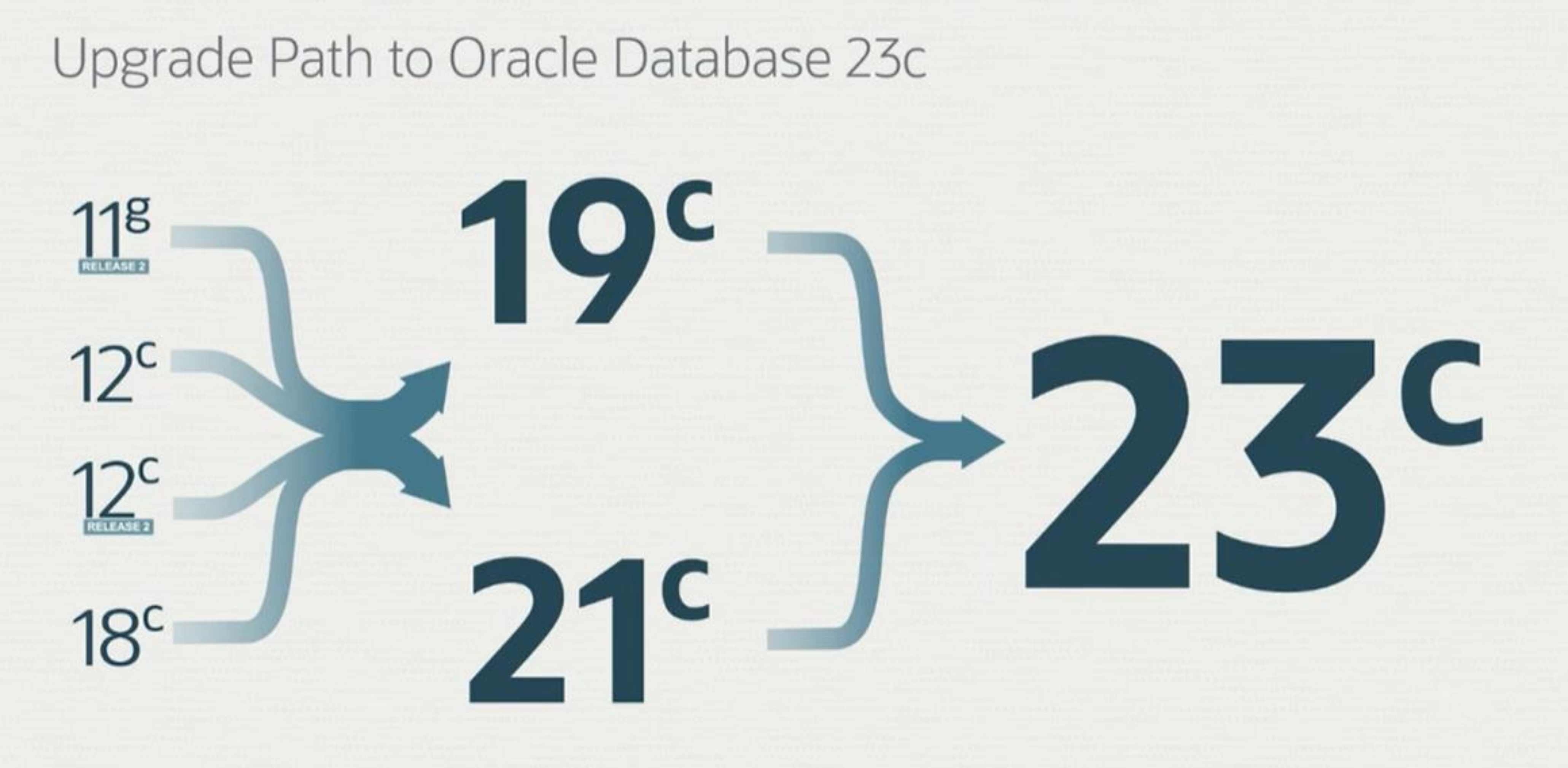
Oracle23c升级策略

- 23c是长期版本



Oracle23c升级策略

- 23c升级路径



谢谢观看

THANKS FOR WATCHING

中国数据库联盟
All China Database Union

墨天轮

点击查看合辑免费获取更多PPT：https://www.modb.pro/topic/655627

扫码关注更多ACDU大会活动
https://www.modb.pro/meetup

添加墨天轮小助手微信：modb666
备注常用数据库名称，入群获取更多技术干货