

# 数据同步

- 一、应用背景：企业有很多数据库，跨库抽取数据，从A库的表抽数据到B库的表，做增量或全量同步

## ▼ 二、数据库连接DATABASE LINK (DB-LINK)

- 1、数据库连接（也是数据库对象，有自己的名称）、是单向连接，旧库访问新库，在旧库创建link，需要用sys账号给scott账号授权
- 2、先创建新的库：开始--ORACLE文件--找红路灯图标文件（database congiguration assistant）具体步骤见操作文档

### ▼ 3、创建DBLINK需要有权限

- (1) 授权


代码演示：

```
GRANT CREATE PUBLIC DATABASE LINK TO SCOTT;  
GRANT CREATE DATABASE LINK TO SCOTT;  
GRANT DROP PUBLIC DATABASE LINK TO SCOTT;
```

- (2) 撤销权限

```
GRANT CREATE PUBLIC DATABASE LINK TO SCOTT;  
GRANT CREATE DATABASE LINK TO SCOTT;  
GRANT DROP PUBLIC DATABASE LINK TO SCOTT;
```

### ▼ 4、创建DBlink

-  DBLINK 是单向访问，从旧库访问新库，所有在旧库创建link

- 代码演示

代码演示：

```
CREATE PUBLIC DATABASE LINK Wind_link ---建立DBLINK  
CONNECT TO SCOTT IDENTIFIED BY "123456" ---数据库地址，help里找TNS名称  
USING '  
  DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP)(HOST = DESKTOP-G6T2GKN)(PORT = 1521))  
  (CONNECT_DATA =  
  (SERVER = DEDICATED)  
  (SERVICE_NAME = W_INDS))  
';
```

-  创建完成后，可以通过link在访问所连数据库的表数据

SELECT \* FROM EMP; ---本库的EMPS数据

SELECT \* FROM EMP@Wind\_link; ---新库W\_inds的EMP表数据

- 5、DBLINKE的使用：同步数据，将所连的表数据同步到本数据库中

## ▼ 三、数据同步（全量抽取）

### ▼ （一）全量同步：数据量不多，可以整张同步

- 1、全量抽取：将目标表的数据全部删除，再将源系统的数据全部插入目标表。此方法保证了数据的质量，但是对于数据量大的表而言，性能太差。

## ▪ 2、创建全量同步过程

---全量同步代码演示  
--创建一个过程


```
CREATE OR REPLACE PROCEDURE GET_ALL_EMP
AS
BEGIN
EXECUTE IMMEDIATE 'TRUNCATE TABLE EMP'; ---使用动态SQL先清空EMP表的数据
INSERT INTO EMP      ---向EMP表全量同步数据
SELECT EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,DEPTNO
FROM EMP@Wind_link;
COMMIT;
END;
```

## ▪ 3、调用同步过程

--调用过程  
BEGIN  
GET\_ALL\_EMP;  
END;

## ▼ 4、注意要点

- (1) 尽量不使用\*, \*不直观, 且有时会使数据插入不对应的字段。
- (2) 表最好别名, 给系统减轻识别负担, 不易产生分歧。
- (3) 工作中做全量抽取时, 通常先判断源表是否有数据, 再执行。如果源表没有数据了, 同步之后目标表也没有数据了, 只能从硬盘里找回数据, 成本代价高

-  全量更新的操作在实际项目中使用场景并不多, 更多的场景会使用增量更新

## ▼ (二) 增量同步: 每天新增数据多, 数据量大的业务明细数据做增量同步

- 1、增量抽取: 只需抽取新增的或修改的数据。此方法性能好, 但容易遗漏。有时源表更新的字段, 在目标表中不存在, 则不需要更新。以时间戳取增量, 对源表删除的数据无能为力
- 2、创建增量同步过程

【增量同步】每次输入参数查询出数据后更新数据  
--创建一个增量同步的过程  
CREATE OR REPLACE PROCEDURE GET\_PART\_EMP(P\_YEAR IN NUMBER)  
AS  
BEGIN  
DELETE FROM EMP WHERE TO\_CHAR(HIREDATE,'YYYY') = P\_YEAR;---支持重跑,  
删除已存在的数据, 再插入新数据完成更新  
COMMIT;  
INSERT INTO EMP  
SELECT EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,DEPTNO  
FROM EMP@Wind\_Link  
WHERE TO\_CHAR(HIREDATE,'YYYY') = P\_YEAR;  
END;

TRUNCATE TABLE EMP;  
SELECT \* FROM EMP;

- 3、调用同步过程

```
--调用过程，进行增量同步  
BEGIN  
  FOR I IN 1980..1990 LOOP  
    GET_PART_EMP(I);  
  END LOOP;  
END;
```

- ▼ 4、增量抽取方法

- 方法一：MERGE（不是所有数据库都通用）
- 方法二：先将目标表的记录在增量范围之内的删除，然后再将增量插入

- 方法三：使用游标，逐一判断每一条增量记录是需要更新还是插入

```
CREATE OR REPLACE PROCEDURE SP_EMPD(P_START_DATE VARCHAR2,
                                     P_END_DATE VARCHAR2) IS
    V_START_DATE DATE := TO_DATE(P_START_DATE, 'YYYY-MM-DD HH24:MI:SS');
    V_END_DATE DATE := TO_DATE(P_END_DATE, 'YYYY-MM-DD HH24:MI:SS');
    CURSOR C_UPDATE IS
        SELECT E.EMPNO,
               E.ENAME,
               E.JOB,
               E.MGR,
               E.HIREDATE,
               E.SAL,
               E.COMM,
               E.DEPTNO
        FROM EMP E
        WHERE E.LAST_UPDATE_DATE >= V_START_DATE
              AND E.LAST_UPDATE_DATE < V_END_DATE;
    CT NUMBER;
BEGIN
    FOR X IN C_UPDATE LOOP
        SELECT COUNT(1) INTO CT FROM EMPEMP A WHERE A.EMPNO = X.EMPNO;
        IF CT = 1 THEN
            UPDATE EMPEMP A
            SET A.ENAME = X.ENAME,
                A.JOB = X.JOB,
                A.MGR = X.MGR,
                A.HIREDATE = X.HIREDATE,
                A.SAL = X.SAL,
                A.COMM = X.COMM,
                A.DEPTNO = X.DEPTNO,
                A.LAST_UPDATE_DATE = SYSDATE
            WHERE A.EMPNO = X.EMPNO;
        ELSIF CT = 0 THEN
            INSERT INTO EMPEMP A
            (A.EMPNO,
            A.ENAME,
            A.JOB,
            A.MGR,
            A.HIREDATE,
            A.SAL,
            A.COMM,
            A.DEPTNO,
            A.LAST_UPDATE_DATE)
            VALUES
            (X.EMPNO,
            X.ENAME,
            X.JOB,
            X.MGR,
            X.HIREDATE,
            X.SAL,
            X.COMM,
            X.DEPTNO,
            SYSDATE);
```

```
END IF;  
END LOOP;  
COMMIT;  
END;
```

```
BEGIN  
SP_EMPD('2017-04-05 00:00:00','2017-04-06 00:00:00');  
END;
```

- 方法四：将增量记录逐一插入目标表，如果违反唯一索引，则更新

```
CREATE OR REPLACE PROCEDURE SP_EMPG(P_START_DATE VARCHAR2,
                                     P_END_DATE  VARCHAR2) IS
  V_START_DATE DATE := TO_DATE(P_START_DATE, 'YYYY-MM-DD HH24:MI:SS');
  V_END_DATE  DATE := TO_DATE(P_END_DATE, 'YYYY-MM-DD HH24:MI:SS');
  CURSOR C_UPDATE IS
    SELECT E.EMPNO,
           E.ENAME,
           E.JOB,
           E.MGR,
           E.HIREDATE,
           E.SAL,
           E.COMM,
           E.DEPTNO
    FROM EMP E
   WHERE E.LAST_UPDATE_DATE >= V_START_DATE
        AND E.LAST_UPDATE_DATE < V_END_DATE;

BEGIN
  FOR X IN C_UPDATE LOOP
    BEGIN
      INSERT INTO EMPEMP A
        (A.EMPNO,
         A.ENAME,
         A.JOB,
         A.MGR,
         A.HIREDATE,
         A.SAL,
         A.COMM,
         A.DEPTNO,
         A.LAST_UPDATE_DATE)
      VALUES
        (X.EMPNO,
         X.ENAME,
         X.JOB,
         X.MGR,
         X.HIREDATE,
         X.SAL,
         X.COMM,
         X.DEPTNO,
         SYSDATE);
    EXCEPTION
      WHEN DUP_VAL_ON_INDEX THEN
        UPDATE EMPEMP A
          SET A.ENAME    = X.ENAME,
              A.JOB      = X.JOB,
              A.MGR       = X.MGR,
              A.HIREDATE  = X.HIREDATE,
              A.SAL       = X.SAL,
              A.COMM      = X.COMM,
              A.DEPTNO    = X.DEPTNO,
              A.LAST_UPDATE_DATE = SYSDATE
         WHERE A.EMPNO = X.EMPNO;
```

```

END;
END LOOP;
END;

BEGIN
SP_EMPG('2017-04-05 00:00:00','2017-04-06 00:00:00');
END;

```

### ▼ (三)

#### ▼ 如果源表数据删除了怎么办？

- 源表的数据一般不会被删除，通常表后面会加一个删除标识字段，用来标识这条记录是否失效，即软删除（华为称为打PD），证明这条记录存在过。
- 如果源表有数据删除了，可以查看源表的日志，看哪些记录被删除了，在目标表相应记录上加上删除标识，但是此方法代价大，成本高。

#### ▼ 两张表A、B关联抽数如何抽取？

- 方法1：两张表全量关联，A表的时间或B表的时间在时间范围内
- 方法2：（A表的增量与B表的全量关联）UNION ALL（B表的增量与A表的全量关联）
- 方法3：（A的增量主键union all B增量主键group by 主键）left join A on 主键 left join B on 主键

### ▼ 四、merge into

- 有的时候表的数据会有滞后的情况，例如3月的财务数据5月份才添加到表中，这个时候上面那个步骤就不能很好的时候数据的新增了，那么我们可以使用MERGE INTO的写法。

#### ▼ merge into 语法

- 语法格式：MERGE（不是所有数据库都通用）  
MERGE INTO 目标表  
USING (增量)  
ON (匹配字段)  
WHEN MATCHED THEN UPDATE SET --UPDATE和SET之间不需要加表名  
WHEN NOT MATCHED THEN INSERT VALUES  
--INSERT和VALUES之间不需要加INTO 表名
- ● 若数据量大时，MERGE删除比DELETE好

#### ▼ (1) 结合merge into 创建同步过程

- 1、如果数据存在，则更新数据，如果数据不存在，则插入数据

## ▪ 2、代码演示

-----如果历史数据有更改，则需要更新一个时间段的数据，用merge into  
CREATE OR REPLACE PROCEDURE GET\_NEW\_EMP(START\_YEAR IN  
NUMBER,END\_YEAR IN NUMBER)

AS

BEGIN

DELETE FROM EMP

WHERE TO\_CHAR(HIREDATE,'YYYY') >= START\_YEAR

AND TO\_CHAR(HIREDATE,'YYYY') <= END\_YEAR; ---支持重跑

COMMIT;

MERGE INTO EMP A

USING (SELECT EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,DEPTNO

FROM EMP@WIND\_LINK

WHERE TO\_CHAR(HIREDATE, 'YYYY') >= START\_YEAR

AND TO\_CHAR(HIREDATE, 'YYYY') <= END\_YEAR) B

ON (A.EMPNO = B.EMPNO)

WHEN MATCHED THEN

UPDATE SET A.ENAME = B.ENAME,

A.JOB = B.JOB,

A.MGR = B.MGR,

A.HIREDATE = B.HIREDATE,

A.SAL = B.SAL,

A.COMM = B.COMM,

A.DEPTNO = B.DEPTNO

WHEN NOT MATCHED THEN

INSERT (A.EMPNO,A.ENAME,A.JOB,A.MGR,A.HIREDATE,A.SAL,A.COMM,A.DEPTNO)

VALUES

(B.EMPNO,B.ENAME,B.JOB,B.MGR,B.HIREDATE,B.SAL,B.COMM,B.DEPTNO);

COMMIT;

END;

## ▪ (2) 调用同步过程

---调用增量过程。进行增量同步

BEGIN

GET\_NEW\_EMP(1980,1990);

END;

SELECT \* FROM EMP

TRUNCATE TABLE EMP;