# DM7 上的日志挖掘

## 一、适用场景

相信许多 DBA 同学都碰到过这样的情况，由于各种原因需要对数据库进行不完全的恢复，但又不确定应该恢复到哪个时间点或者 SCN。在 DM7 中，用户可以使用 DBMS_LOGMNR 包对归档日志进行挖掘，重构出 DDL 和 DML 等操作，并通过获取的信息进行更深入的分析；同样，可以对归档日志文件进行恢复被误操作的数据，并进行故障跟踪，定位执行误操作的用户信息。目前 DBMS_LOGMNR 只支持归档日志进行分析，配置归档后，还需要将 dm.ini 中的 RLOG_APPEND_LOGIC 选项配置为 1 或 2。

注：需要在产生归档日志的数据库上进行分析，利用备份文件还原的测试库上无法进行日志挖掘，因为当数据库经过备份还原并恢复后，DB_MAGIC 就会发生改变，在还原库上进行日志分析时会报“[DBMS_LOGMNR.START_LOGMNR] 归档日志 MAGIC 错误”错误。

## 二、环境检查

注：本文实验环境：

DM7 数据库：DM Database Server x64 V7.1.6.33-Build(2017.12.11-87269)ENT

### 2.1、检查是否创建了系统包

如未创建，可以执行如下命令：

```
SP_CREATE_SYSTEM_PACKAGES(1);
```

### 2.2、检查是否开启了归档和追加日志

```
select para_name,para_value from v$dm_ini where para_name  in ('ARCH_INI','RLOG_APPEND_LOGIC');
```

注：RLOG_APPEND_LOGIC 需要设置为 1 或 2， 1 代表如果有主键列，记录 UPDATE 和 DELETE 操作时只包含主键列信息， 若没有主键列则包含所有列信息； 2 代表不论是否有主键列， 记录 UPDATE 和 DELETE 操作时都包含所有列的信息。

如果未开启归档和追加日志，可以参照如下进行设置：

-1-修改 dm.ini 中的参数

```
ARCH_INI = 1
RLOG_APPEND_LOGIC = 1
```

注：静态参数，需要重启数据库后生效

-2-dmarch.ini 配置本地归档

```
[ARCHIVE_LOCAL1]ARCH_TYPE
= LOCAL
ARCH_DEST = d:\dm7data\arch
```

```
ARCH_FILE_SIZE = 128
ARCH_SPACE_LIMIT = 0
```

注：归档文件配置完毕后，需要重启数据库后生效

# 三、开始日志挖掘

## 3.1、添加归档日志文件

### 3.1.1、查询数据库当前有哪些归档日志

```
SELECT NAME , FIRST_TIME , NEXT_TIME , FIRST_CHANGE# , NEXT_CHANGE# FROM  V$ARCHIVED_LOG;
```

### 3.1.2、添加一个或多个需要分析的归档日志文件

```
DBMS_LOGMNR.ADD_LOGFILE('/dmdb/xxx/xxx/xxx/ARCHIVE_LOCAL1_20171214120114980.log');
```

注：对于不在数据库默认归档路径下的归档日志，可以直接指定归档日志的绝对路径添加进分析列表。

### 3.1.3、查询通过 ADD_LOGFILE 添加的归档日志文件

可以查询 V$LOGMNR_LOGS 动态视图进行插入，如下：

```
SELECT LOW_SCN, NEXT_SCN, LOW_TIME, HIGH_TIME, LOG_ID, FILENAME FROM  V$LOGMNR_LOGS;
```

## 3.2、启动归档日志文件分析

```
DBMS_LOGMNR.START_LOGMNR(OPTIONS=>2128 , STARTTIME=>TO_DATE('2018-1-24 12:01:12','YYYY-MM-DD
HH24:MI:SS') , ENDTIME=>TO_DATE('2018-01-24 04:20:03','YYYY-MM-DD  HH24:MI:SS'));
```

注：可以指定日志文件分析的时间段或 SCN 范围，同样可以指定 Options 的功能组合，如 下：

Options：提供如下的可选模式，各模式可以通过 + 或者按位或来进行组合。其它位的值如 1、4、 8 等目前不支持，配置后不会报错，但是没有效果。

| Options | 对应值 | 说明 |
| --- | --- | --- |
| COMMITTED_DATA_ONLY | 2 | 仅从已交的事务的日志中挖掘信息 |
| DICT_FROM_ONLINE_CATALOG | 16 | 使用在线字典 |
| NO_SQL_DELIMITER | 64 | 拼写的 SQL 语句最后不添加分隔符 |
| NO_ROWID_IN_STMT | 2048 | 拼写的 SQL 语句中不包含 ROWID |

### 3.2.1、查看归档日志文件的分析结果

如要查看归档日志文件的分析结果，可以通过动态视图 V$LOGMNR_CONTENTS 进行查询，如下：

```
select
TIMESTAMP,START_TIMESTAMP,COMMIT_TIMESTAMP,OPERATION,OPERATION_CODE,ROLL_BACK,TABLE_NAME,ROW_
ID,USERNAME,DATA_OBJ#,DATA_OBJV#,SQL_REDO,REDO_VALUE,UNDO_VALUE
from V$LOGMNR_CONTENTS
where table_name='XXX';
```

注：可以根据需要指定追踪信息，如表名、用户名、时间段等,在数据库中执行的操作会被解析为单行元组的 SQL 操作，即在数据库中执行一条 update XX set name=xx where id<100;则在日志分析结果中会解析为一条条单行元组的 SQL 操作，如 update XX set name=xx where id=1；、update table_name set name=xx where id=2；等所有 id 小于 100 的 SQL 操作。

OPERATION 字段代表操作类型，主要包括 start、 insert、 update、delete、 commit、rollback 等语句

OPERATION_CODE 代表操作类型代码，1 表示插入操作， 2 表示删除操作， 3 表示更新操作， 6 表示事务起始语句， 7 表示提交操作， 9 表示批量更新， 36 表示回滚操作。

### 3.3、终止归档日志文件分析

```
DBMS_LOGMNR.END_LOGMNR();
```

注：进行日志挖掘过程中，在 V$LOGMNR_LOGS、V$LOGMNR_CONTENTS 等数据库动态性能视图上会产生分析数据，数据存储在 TEMP 临时表空间上，会话断开或终止归档日志文件分析后，数据会被清除。

看到这里，相信大家对 DBMS_LOGMNR 的基本用法有了一定的了解，如果大家想对归档日志挖掘进行进一步的学习，可以参见《DM7 系统包使用手册》第七章<DBMS_LOGMNR 包>。