**Oracle® Database Utilities  11g Release 2 (11.2) 》19 Using LogMiner to Analyze Redo Log Files**

Requirements

The following are requirements for the source and mining database, the data dictionary, and the redo log files that LogMiner will mine:

•Source and mining database

◦Both the source database and the mining database must be running on the same hardware platform.

◦The mining database can be the same as, or completely separate from, the source database.

◦The mining database must run the same release or a later release of the Oracle Database software as the source database.

◦The mining database must use the same character set (or a superset of the character set) used by the source database.

•LogMiner dictionary

◦The dictionary must be produced by the same source database that generates the redo log files that LogMiner will analyze.

•All redo log files:

◦Must be produced by the same source database

◦Must be associated with the same database RESETLOGS SCN.

◦Must be from a release 8.0 or later Oracle Database. However, several of the LogMiner features introduced as of release 9.0.1 work only with redo log files produced on an Oracle9i or later database. See "Supported Databases and Redo Log File Versions".

**LogMiner Dictionary Options**

•Using the Online Catalog

EXECUTE DBMS\_LOGMNR.START\_LOGMNR(-

   OPTIONS => DBMS\_LOGMNR.DICT\_FROM\_ONLINE\_CATALOG);

•Extracting a LogMiner Dictionary to the Redo Log Files

EXECUTE DBMS\_LOGMNR\_D.BUILD( -

   OPTIONS=> DBMS\_LOGMNR\_D.STORE\_IN\_REDO\_LOGS);

•Extracting the LogMiner Dictionary to a Flat File

UTL\_FILE\_DIR = /oracle/database

STARTUP

EXECUTE DBMS\_LOGMNR\_D.BUILD('dictionary.ora', -

   '/oracle/database/', -

    DBMS\_LOGMNR\_D.STORE\_IN\_FLAT\_FILE);

**Redo Log File Options**

•Manually

EXECUTE DBMS\_LOGMNR.ADD\_LOGFILE( -

   LOGFILENAME => '/oracle/logs/log1.f', -

   OPTIONS => DBMS\_LOGMNR.NEW);

EXECUTE DBMS\_LOGMNR.ADD\_LOGFILE( -

   LOGFILENAME => '/oracle/logs/log2.f', -

   OPTIONS => DBMS\_LOGMNR.ADDFILE);

select filename from v$logmnr\_logs;

**Starting LogMiner**

The following list is a summary of LogMiner settings that you can specify with the OPTIONS parameter to DBMS\_LOGMNR.START\_LOGMNR and where to find more information about them.

•DICT\_FROM\_ONLINE\_CATALOG — See "Using the Online Catalog"

•DICT\_FROM\_REDO\_LOGS — See "Start LogMiner"

•CONTINUOUS\_MINE — See "Redo Log File Options"

•COMMITTED\_DATA\_ONLY — See "Showing Only Committed Transactions"

•SKIP\_CORRUPTION — See "Skipping Redo Corruptions"

•NO\_SQL\_DELIMITER — See "Formatting Reconstructed SQL Statements for Re-execution"

•PRINT\_PRETTY\_SQL — See "Formatting the Appearance of Returned Data for Readability"

•NO\_ROWID\_IN\_STMT — See "Formatting Reconstructed SQL Statements for Re-execution"

•DDL\_DICT\_TRACKING — See "Tracking DDL Statements in the LogMiner Dictionary"

=========================================

如下面的例子，我们仅仅分析2013年6月8日的日志，：

EXECUTE dbms\_logmnr.start\_logmnr(

DictFileName => dictfilename=>'D:\..\practice\LOGMNR\dictionary.ora',

StartTime =>to\_date('2013-6-8 00:00:00','YYYY-MM-DD HH24:MI:SS')

EndTime =>to\_date(''2013-6-8 23:59:59','YYYY-MM-DD HH24:MI:SS '));

**Querying V$LOGMNR\_CONTENTS for Redo Data of Interest**

到现在为止，我们已经分析得到了重作日志文件中的内容。动态性能视图v$logmnr\_contents包含LogMiner分析得到的所有的信息。

SELECT sql\_redo FROM v$logmnr\_contents;

如果我们仅仅想知道某个用户对于某张表的操作，可以通过下面的SQL查询得到，该查询可以得到用户LOGMINER对表EMP所作的一切工作。

SELECT OPERATION, SQL\_REDO, SQL\_UNDO

   FROM V$LOGMNR\_CONTENTS

   WHERE SEG\_OWNER = 'OE' AND SEG\_NAME = 'ORDERS' AND

   OPERATION = 'DELETE' AND USERNAME = 'RON';

|  |  |  |
| --- | --- | --- |
| **序号** | **名称** | **含义** |
| 1 | SCN | 特定数据变化的系统更改号 |
| 2 | TIMESTAM | 数据改变发生的时间 |
| 3 | COMMIT\_TIMESTAMP | 数据改变提交的时间 |
| 4 | SEG\_OWNER | 数据发生改变的段名称 |
| 5 | SEG\_NAME | 段的所有者名称 |
| 6 | SEG\_TYPE | 数据发生改变的段类型 |
| 7 | SEG\_TYPE\_NAME | 数据发生改变的段类型名称 |
| 8 | TABLE\_SPACE | 变化段的表空间 |
| 9 | ROW\_ID | 特定数据变化行的ID |
| 10 | SESSION\_INFO | 数据发生变化时用户进程信息 |
| 11 | OPERATION | 重做记录中记录的操作(如INSERT) |
| 12 | SQL\_REDO | 可以为重做记录重做指定行变化的SQL语句(正向操作) |
| 13 | SQL\_UNDO | 可以为重做记录回退或恢复指定行变化的SQL语句(反向操作) |

需要强调一点的是，视图v$logmnr\_contents中的分析结果仅在我们运行过程'dbms\_logmrn.start\_logmnr'这 个会话的生命期中存在。这是因为所有的LogMiner存储都在PGA内存中，所有其他的进程是看不到它的，同时随着进程的结束，分析结果也随之消失。

最后，使用过程DBMS\_LOGMNR.END\_LOGMNR终止日志分析事务，此时PGA内存区域被清除，分析结果也随之不再存在。

在使用LogMiner读取归档/在线日志需要按照第2章节进行设置，设置完毕后可以对归档和在线日志进行分析。特别是需要开启LogMiner日志补充 模式，如果没有开始LogMiner补充模式将无法查看DDL语句，按照测试结果看，只有开始LogMiner日志补充模式后，才能查看DDL语句，在此 之前进行DDL将无法进行查看

alter database add supplemental log data;

4 退出logmnr

SQL> execute dbms\_logmnr.end\_logmnr;