

truncate 恢复方法集合

1.1 BLOG 文档结构图

truncate 恢复方法集合
1.1 BLOG 文档结构图
1.2 前言部分
1.2.1 导读和注意事项
1.2.2 相关参考文章链接
1.2.3 本文简介
1.3 收费软件
1.4 免费软件
1.4.1 fy_recover_data 包
1.4.2 gdul 工具
1.4.2.1 gDUL 功能特点
1.4.3 dul
1.4.4 bbed 来恢复
第 2 章实验部分
2.1 实验环境介绍
2.2 实验目标
2.3 实验过程
2.3.1 fy_recover_data 包恢复 truncate 的表
2.3.2 gdul 恢复 truncate 的表
2.4 实验总结

About Me

1.2 前言部分

1.2.1 导读和注意事项

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~O(∩_∩)O~：

① truncate 操作后的恢复方法（重点）

Tips：

① 若文章代码格式有错乱，推荐使用 QQ、搜狗或 360 浏览器，也可以下载 pdf 格式的文档来查看，pdf 文

档下载地址：<http://yunpan.cn/cdEQedhCs2kFz>（提取码：ed9b）

② 本篇 BLOG 中命令的输出部分需要特别关注的地方我都用灰色背景和粉红色字体来表示，比如下边的例

子中，thread 1 的最大归档日志号为 33，thread 2 的最大归档日志号为 43 是需要特别关注的地方；而命令一般使

用黄色背景和红色字体标注；对代码或代码输出部分的注释一般采用蓝色字体表示。

```
List of Archived Logs in backup set 11
Thrd Seq      Low SCN      Low Time      Next SCN      Next Time
-----
1      32           1621589      2015-05-29 11:09:52 1625242      2015-05-29 11:15:48
1      33           1625242      2015-05-29 11:15:48 1625293      2015-05-29 11:15:58
2      42           1613951      2015-05-29 10:41:18 1625245      2015-05-29 11:15:49
2      43           1625245      2015-05-29 11:15:49 1625253      2015-05-29 11:15:53

[ZFXADB1:root]:/>ls -l
T_XDESK_APP1_vg
rootvg
[ZFXADB1:root]:/>
00:27:22 SQL> alter tablespace idxtbs read write;

====> 2097152*512/1024/1024/1024=1G
```

本文如有错误或不完善的地方请大家多多指正，ITPUB 留言或 QQ 皆可，您的批评指正是我写作的最大动力。

1.2.2 相关参考文章链接

1.2.3 本文简介

truncate 操作是比较危险的操作，不记录 redo，不能通过闪回查询来找回数据，但是只要段所占用的块没有全部被

重新占用的情况下，我们还是可以通过一些特殊的办法来找回 truncate 掉的数据，因为当 Truncate 命令发起之后，

Oracle 实际上并没有在删除底层数据块上的数据，而是要等到重用的时候才会把这一部分数据回收，于是这给了我们一个能够恢复数据库的机会。

总体而言，恢复的办法是通过一些大牛写的工具来恢复，分为收费和免费的，我们下边分别说明。实验部分我们只实验 fy_recover_data 包和 gdul 工具。

有的实验是很久之前做的，这篇文章发布太晚，因为中间学习了 DUL 和 BBED 的相关知识。

1.3 收费软件

这里简单列举一下，具体内容请到相关网站了解：

工具名称	下载地址	作者	软件
ODU	http://www.oracleodu.com/cn/	老熊	命令行操作
PRM-DUL	http://www.parnassusdata.com/	Maclean Liu	图形界面操作
AUL/mydul	http://www.dbatools.net/mydul/	d.c.b.a/楼方鑫	命令行

1.4 免费软件

1.4.1 fy_recover_data 包

作者个人信息：

WWW.HelloDBA.COM

Created By: Fuyuncat

Created Date: 08/08/2012

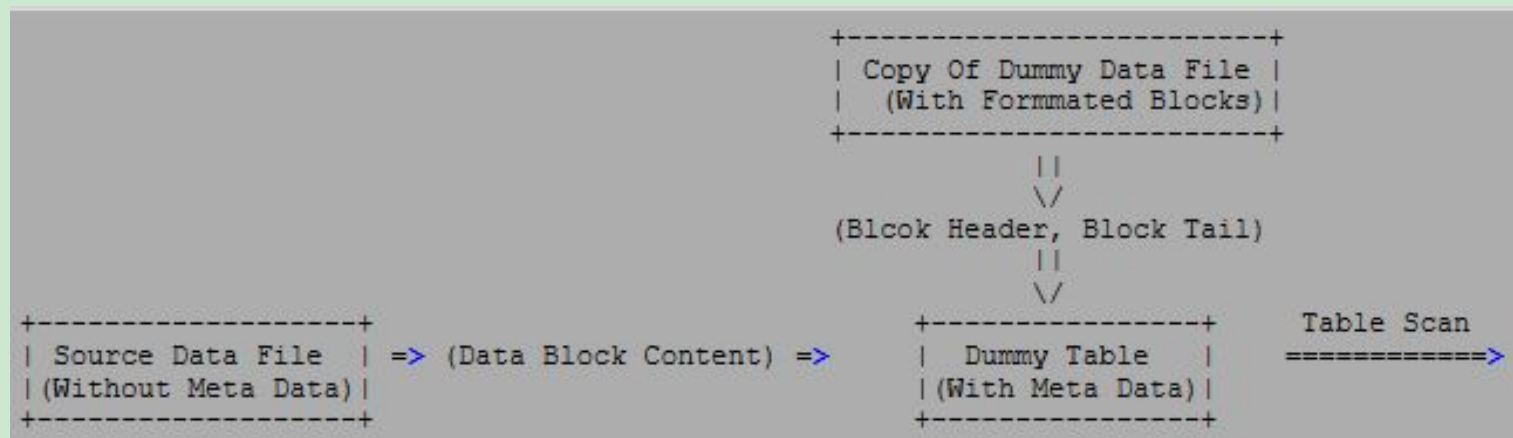
Email: Fuyuncat@gmail.com

Copyright (c), 2014, WWW.HelloDBA.COM All rights reserved.

Latest Version: http://www.HelloDBA.com/download/FY_Recover_Data.zip

该包采用纯 plsql 语句恢复被 truncate 掉的表，操作比较简单，下载可以去官网下载，或者小麦苗的云盘共享目录。

Fy_Recover_Data 是利用 Oracle 表扫描机制、数据嫁接机制恢复 TRUNCATE 或者损坏数据的工具包。由纯 PLSQL 编写，原理图如下：



包内容：



FY_Recover_Data.pck

1.4.2 gdul 工具

GDUL 是老耿开发的一款类 dul 工具，当数据库由于某种原因无法打开时，可以利用 GDUL 把表数据直接读取出来，工具下载地址参考小麦苗的 blog，老耿的信息如下：

```
*****
GDUL for ORACLE DB.
Version 4.0.0.1, build date: 2016.04.12.
Copyright (c) 2007, 2016. Andy Geng.  ALL RIGHTS RESERVED.
Email: dbtool@aliyun.com
WeChat official account: dbtool
QQ group: 235019291
*****
```

1.4.2.1 gDUL 功能特点

- ❖ 完整支持多种格式导出，包括 expdp,exp,text 格式。目前市面上的类 dul 工具只有 gDUL 支持 expdp 格式。
- ❖ 支持 ASM 文件系统，并内置 asmcmd 命令。

- ❖ 支持绝大多数列类型,支持常见的 NUMBER,CHAR, VARCHAR2, DATE ,LOB, LONG 等类型。。其中 SecureFile LOB 支持压缩,尚不支持去重和加密。
- ❖ 支持导出常规表、IOT、Cluster 表、分区表、压缩表。
- ❖ 支持 truncated 表、删除行恢复。
- ❖ 支持常规表空间和 bigfile 表空间。
- ❖ 支持主流硬件平台(HP-UX,AIX, Solaris, Linux, Windows), 各个平台仅需单一的可执行文件,方便分发。
- ❖ 重点是——永久免费使用,无需额外费用,不开源。

1.4.3 dul

DUL 是 Data Unloader 的缩写,是一个荷兰的 Oracle 工程师开发的,他的名字为 Bernard Van Duijnen。DUL 是一个 C 开发的小程序,编译后整个程序只有一个文件,大小也不过几百 KB,它工作时不需 Oracle RDBMS 以及任何的 Oracle 的程序、组件,它可以直接从一个坏了数据库的数据文件中读取数据,生成 IMP 或 SQL*Loader 可以识别的文件。

DUL 不是一个商用化的产品,Oracle 不卖、不提供也不支持它的使用。DUL 只有在 Oracle 的内部网站才可以下载到,因此也只有 Oracle 的 Supporter 才能下载到这个工具,如果与 Oracle 的 Supporter 熟悉,没准他私底下会给你一个,这个工具也因此有一些流落到民间,被一些人收入囊中,奉为珍宝。

不同的平台、不同版本的数据库都有相应的 DUL 软件,9.x 及之前 DUL 是没有 License 限制的,也就是有这个工具可以无限制的使用,不过最新的 DUL 在这方面已经改进了,kamus 说最新 DUL 拿到手只能用一个月。关于这一小点稍总结一下,获得 DUL 有以下几种途径:

✚ 如果你是 Oracle 的 Supporter ,可以在内部网站下载,地址为: <http://www.nl.oracle.com/support/dul/>

✚ 如果你有 Oracle 的 Supporter 的朋友可以向他们要一个,itpub 也几位斑竹都到 Oracle 了,如 coolyl, kamus,lunar。

✚ 一些 dul 流落到民间,可以向有这软件的朋友要一个,不过他们不一定有你需要的那个。

所以关于 DUL 我们不做过多的解释。

1.4.4 bbed 来恢复

这个比较复杂，若对 oracle 不熟悉或者 bbed 不熟悉都不推荐使用这个，具体案例参考：

<http://blog.itpub.net/26736162/viewspace-2080727/>

第 2 章 实验部分

2.1 实验环境介绍

项目	db
db 类型	单实例
db version	11.2.0.4.0
db 存储	FS
主机 IP 地址/hosts 配置	192.168.59.129
OS 版本及 kernel 版本	AIX 7.1 64 位
归档模式	Archive Mode
ORACLE_SID	oralhr

2.2 实验目标

将 truncate 掉的表数据成功找回。

2.3 实验过程

2.3.1 fy_recover_data 包恢复 truncate 的表

```
[ZFXDESKDB1:oracle]:/oracle>ORACLE_SID=oraESKDB1
[ZFXDESKDB1:oracle]:/oracle>sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Mon Mar 21 15:51:55 2016

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
Data Mining and Real Application Testing options

SYS@oraESKDB1> set time on;
15:52:10 SYS@oraESKDB1> set timing on;
15:52:10 SYS@oraESKDB1> set serveroutput on;
15:52:10 SYS@oraESKDB1> create table scott.TB_0321 as SELECT * FROM dba_objects;

Table created.

Elapsed: 00:00:00.59
15:52:18 SYS@oraESKDB1> SELECT COUNT(1) FROM scott.TB_0321;

COUNT(1)
-----
      86651

Elapsed: 00:00:00.19
15:52:24 SYS@oraESKDB1> INSERT INTO scott.TB_0321 SELECT * FROM scott.TB_0321;

86651 rows created.

Elapsed: 00:00:00.26
15:52:30 SYS@oraESKDB1> COMMIT;
Commit complete.

Elapsed: 00:00:00.01
15:52:30 SYS@oraESKDB1> INSERT INTO scott.TB_0321 SELECT * FROM scott.TB_0321;
COMMIT;

173302 rows created.

Elapsed: 00:00:00.43
15:53:02 SYS@oraESKDB1> SELECT COUNT(1) FROM scott.TB_0321;

COUNT(1)
-----
    346604

Elapsed: 00:00:00.27
16:15:18 SYS@oraESKDB1> SELECT d.BYTES/1024/1024 FROM dba_segments d WHERE d.segment_name ='TB_0321';

D.BYTES/1024/1024
-----
                40

Elapsed: 00:00:00.44
16:15:25 SYS@oraESKDB1> truncate table scott.TB_0321;

Table truncated.

Elapsed: 00:00:00.20
16:15:46 SYS@oraESKDB1> SELECT COUNT(1) FROM scott.TB_0321;

COUNT(1)
```

0

Elapsed: 00:00:00.01

====》数据已经被 truncate 掉了，下边我们来恢复

16:15:52 SYS@oraESKDB1> @/oracle/FY_Recover_Data.pck

Package created.

Elapsed: 00:00:00.06

Package body created.

Elapsed: 00:00:00.03

16:15:59 SYS@oraESKDB1> exec fy_recover_data.recover_truncated_table('scott','TB_0321');

16:16:06: Use existing Directory Name: FY_DATA_DIR

16:16:07: Recover Table: SCOTT.TB_0321\$

16:16:09: Restore Table: SCOTT.TB_0321\$\$

16:16:24: Copy file of Recover Tablespace: FY_REC_DATA_COPY.DAT1

16:16:24: begin to recover table SCOTT.TB_0321

16:16:24: Use existing Directory Name: TMP_HF_DIR

16:17:09: Recovering data in datafile +DATA/oraeskd/backup/users.351.902678817

16:17:09: Use existing Directory Name: TMP_HF_DIR

16:39:16: 4984 truncated data blocks found.

16:39:16: 346604 records recovered in backup table SCOTT.TB_0321\$\$

16:39:17: Total: 4984 truncated data blocks found.

16:39:17: Total: 346604 records recovered in backup table SCOTT.TB_0321\$\$

16:39:17: Recovery completed.

16:39:17: Data has been recovered to SCOTT.TB_0321\$\$

PL/SQL procedure successfully completed.

Elapsed: 00:23:11.59

16:39:17 SYS@oraESKDB1> SELECT COUNT(1) FROM scott.TB_0321\$;

COUNT(1)

346604

Elapsed: 00:00:01.55

16:40:51 SYS@oraESKDB1>

16:40:51 SYS@oraESKDB1> alter table scott.TB_0321 nologging;

Table altered.

Elapsed: 00:00:00.03

16:41:43 SYS@oraESKDB1> insert /*+append*/ into scott.TB_0321 select * from scott.TB_0321\$;

346604 rows created.

Elapsed: 00:00:00.86

16:41:52 SYS@oraESKDB1> commit;

Commit complete.

Elapsed: 00:00:00.01

16:41:55 SYS@oraESKDB1> alter table scott.TB_0321 logging;

Table altered.

Elapsed: 00:00:00.02

16:42:06 SYS@oraESKDB1>

16:42:06 SYS@oraESKDB1> drop tablespace FY_REC_DATA including contents and datafiles;

Tablespace dropped.


```
Elapsed: 00:00:08.00
16:42:35 SYS@oraESKDB1> drop tablespace    FY_RST_DATA    including contents and datafiles;

Tablespace dropped.

Elapsed: 00:00:07.59
16:42:44 SYS@oraESKDB1>
```

数据成功恢复。

2.3.2 gdul 恢复 truncate 的表

```
set time on;
set timing on;
set serveroutput on;
drop table scott.TB_0322_05;
create table scott.TB_0322_05    as SELECT * FROM dba_objects;

SELECT COUNT(1) FROM    scott.TB_0322_05;
INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;
COMMIT;
INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;
COMMIT;
INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;
COMMIT;
INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;
COMMIT;
SELECT COUNT(1) FROM    scott.TB_0322_05;

SELECT d.BYTES/1024/1024 FROM dba_segments d WHERE d.segment_name ='TB_0322_05';

truncate table scott.TB_0322_05;

alter system checkpoint;

col ownere format a10
col DIRECTORY_NAME format a30
col DIRECTORY_PATH format a50
select OWNER,DIRECTORY_NAME,DIRECTORY_PATH from    dba_directories;

bootstrap
desc scott.TB_0322_05
unload table    scott.TB_0322_05
scan tablespace 4
untrunc table    scott.TB_0322_05

cp SCOTT_TB_0322_05.dmp /oracle/app/oracle/admin/oralhr/dpdump/
impdp    scott/tiger directory=DATA_PUMP_DIR dumpfile=SCOTT_TB_0322_05.dmp
LOGFILE=SCOTT_TB_0322_05.log TABLES=TB_0322_05
```

```
15:41:04 SQL> set time on;
15:59:49 SQL> set timing on;
15:59:49 SQL> set serveroutput on;
15:59:49 SQL> drop table scott.TB_0322_05;
create table scott.TB_0322_05    as SELECT * FROM dba_objects;

SELECT COUNT(1) FROM    scott.TB_0322_05;
INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;
```

Table dropped.

Elapsed: 00:00:00.07

15:59:49 SQL> COMMIT;

INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;

COMMIT;

INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;

COMMIT;

INSERT INTO scott.TB_0322_05 SELECT * FROM scott.TB_0322_05;

COMMIT;

SELECT COUNT(1) FROM scott.TB_0322_05;

SELECT d.BYTES/1024/1024 FROM dba_segments d WHERE d.segment_name = 'TB_0322_05';

truncate table scott.TB_0322_05;

alter system checkpoint;

Table created.

Elapsed: 00:00:00.97

15:59:50 SQL> 15:59:50 SQL>

COUNT(1)

75707

Elapsed: 00:00:00.86

15:59:51 SQL>

75707 rows created.

Elapsed: 00:00:00.23

15:59:52 SQL>

Commit complete.

Elapsed: 00:00:00.17

15:59:52 SQL>

151414 rows created.

Elapsed: 00:00:00.50

15:59:52 SQL>

Commit complete.

Elapsed: 00:00:00.23

15:59:52 SQL>

302828 rows created.

Elapsed: 00:00:01.63

15:59:54 SQL>

Commit complete.

Elapsed: 00:00:00.22

15:59:54 SQL>

605656 rows created.

Elapsed: 00:00:06.19

16:00:00 SQL>

Commit complete.

Elapsed: 00:00:00.02

16:00:01 SQL>

COUNT(1)

1211312

Elapsed: 00:00:00.07

16:00:01 SQL> 16:00:01 SQL>

D.BYTES/1024/1024

136

```

Elapsed: 00:00:00.17
16:00:01 SQL> 16:00:01 SQL> 16:00:01 SQL>
Table truncated.

Elapsed: 00:00:01.26
16:00:02 SQL> 16:00:02 SQL>
System altered.

Elapsed: 00:00:00.15
16:00:02 SQL>
16:00:02 SQL> SELECT COUNT(1) FROM    scott.TB_0322_05;

COUNT(1)
-----
0

Elapsed: 00:00:00.00
16:02:35 SQL>

```

```
[oracle@ZFFR4CB1101:/home/oracle/gdul]$ ./gdul
```

```

*****
GDUL for ORACLE DB.
Version 3.5.0.1, build date: 2016.03.07.
Copyright (c) 2007, 2016. Andy Geng.  ALL RIGHTS RESERVED.
Email: gengyonghui@aliyun.com
QQ group: 235019291, WeChat Official Account: dbtool
*****

```

```
GDUL> bootstrap
```

```
Bootstrap finish.
```

```
GDUL> desc scott.TB_0322_05
```

```
object_id: 78302, dataobj#: 78303, cluster tab#: 0
```

```
segment header: (ts#: 4, rfile#: 4, block#: 682))
```

Seg	Column#	Column#	Name	Null?	Type
1	1	1	OWNER		VARCHAR2(30)
2	2	2	OBJECT_NAME		VARCHAR2(128)
3	3	3	SUBOBJECT_NAME		VARCHAR2(30)
4	4	4	OBJECT_ID		NUMBER
5	5	5	DATA_OBJECT_ID		NUMBER
6	6	6	OBJECT_TYPE		VARCHAR2(19)
7	7	7	CREATED		DATE
8	8	8	LAST_DDL_TIME		DATE
9	9	9	TIMESTAMP		VARCHAR2(19)
10	10	10	STATUS		VARCHAR2(7)
11	11	11	TEMPORARY		VARCHAR2(1)
12	12	12	GENERATED		VARCHAR2(1)
13	13	13	SECONDARY		VARCHAR2(1)
14	14	14	NAMESPACE		NUMBER
15	15	15	EDITION_NAME		VARCHAR2(30)

```
GDUL> unload table scott.TB_0322_05
```

```
2016-03-22 16:01:54...unloaded "SCOTT"."TB_0322_05" 0 rows
```

```
GDUL> scan tablespace 4
```

```
start scan tablespace 4...
```

```
scan tablespace completed.
```

```
GDUL> untrunc table scott.TB_0322_05
```

```
2016-03-22 16:04:29...untruncating table TB_0322_05 1211312 rows unloaded.
```

```
GDUL>
```

```
16:02:35 SQL> select * from dba_directories;
```

OWNER	DIRECTORY_NAME	DIRECTORY_PATH
SYS	SUBDIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry//2002/Sep

```

SYS          SS_OE_XMLDIR
/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry/
SYS          LOG_FILE_DIR          /u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/log/
SYS          MEDIA_DIR
/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/product_media/
SYS          XMLDIR          /u01/app/oracle/product/11.2.0/dbhome_1/rdbms/xml
SYS          DATA_FILE_DIR
/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/sales_history/
SYS          DATA_PUMP_DIR          /u01/app/oracle/product/11.2.0/dbhome_1/rdbms/log/
SYS          ORACLE_OCM_CONFIG_DIR          /u01/app/oracle/product/11.2.0/dbhome_1/ccr/state

```

8 rows selected.

Elapsed: 00:00:00.00

16:05:29 SQL>

```

[oracle@ZFFR4CB1101:/home/oracle/gdul/dump]$ impdp scott/tiger directory=DATA_PUMP_DIR
dumpfile=SCOTT_TB_0322_05.dmp LOGFILE=SCOTT_TB_0322_05.log TABLES=TB_0322_05

```

Import: Release 11.2.0.3.0 - Production on Tue Mar 22 16:16:48 2016

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

Connected to: Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production

With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,

Data Mining and Real Application Testing options

Master table "SCOTT"."SYS_IMPORT_TABLE_01" successfully loaded/unloaded

Starting "SCOTT"."SYS_IMPORT_TABLE_01": scott/***** directory=DATA_PUMP_DIR dumpfile=SCOTT_TB_0322_05.dmp

LOGFILE=SCOTT_TB_0322_05.log TABLES=TB_0322_05

Processing object type TABLE_EXPORT/TABLE/TABLE_DATA

.. imported "SCOTT"."TB_0322_05" 117.1 MB 1211312 rows

Job "SCOTT"."SYS_IMPORT_TABLE_01" successfully completed at 16:16:59

```

[oracle@ZFFR4CB1101:/home/oracle/gdul/dump]$

```

```

[oracle@ZFFR4CB1101:/home/oracle]$ sqlplus / as sysdba

```

SQL*Plus: Release 11.2.0.3.0 Production on Tue Mar 22 16:17:39 2016

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Connected to:

Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production

With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,

Data Mining and Real Application Testing options

```

SQL> SELECT COUNT(1) FROM scott.TB_0322_05;

```

```

COUNT(1)
-----

```

```

1211312

```

```

SQL>

```

数据成功恢复。

2.4 实验总结

总体而言用 fy_recover_data 包或 GDUL 工具都是非常好的，fy_recover_data 可以恢复 truncate 的数据，但不能恢复 drop 的数据，而 GDUL 工具就比较全面了，具体可以参考前边的简介或下载文档来看，小麦苗的共享云盘里也有比较全的文档。

About Me

本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用

ITPUB BLOG：<http://blog.itpub.net/26736162>

本文地址：<http://blog.itpub.net/26736162/viewspace-2082965/>

本文 pdf 版：<http://yunpan.cn/cdEQedhCs2kFz>（提取码：ed9b）

QQ：642808185 若加 QQ 请注明您所正在读的文章标题

于 2016-03-10 10:00~ 2016-04-15 19:00 在中行完成

【版权所有，文章允许转载，但须以链接方式注明源地址，否则追究法律责任】