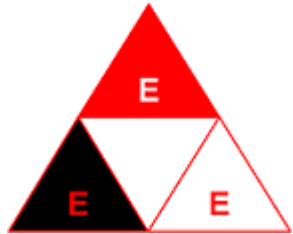


The Recovery Area: Why it is recommended



presented by

Uwe Hesse



Senior Principal Instructor at **Oracle** University

Blogging on uhesse.com

Twitter Handle @UweHesse

Agenda

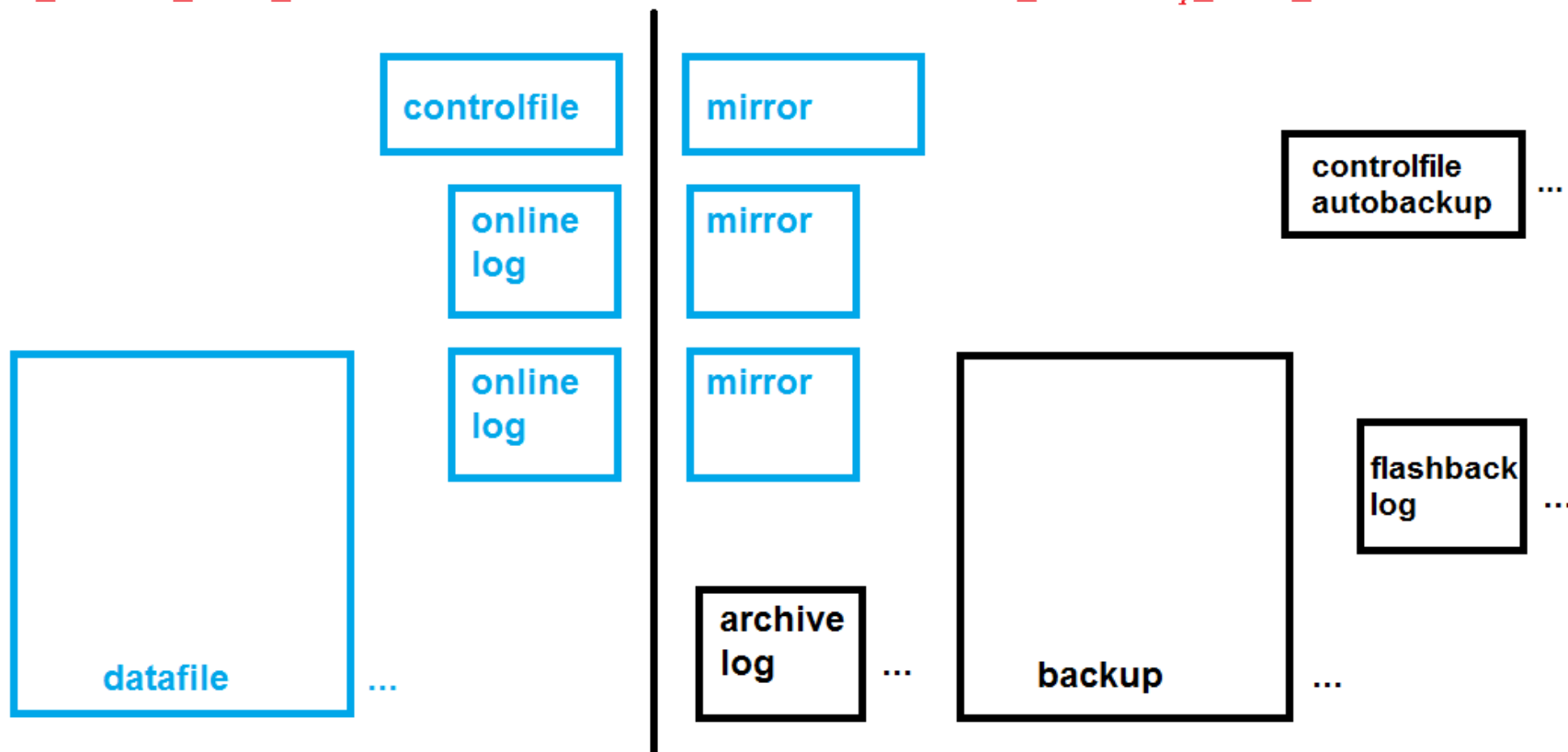
The Recovery Area helps:

- During database creation
- Implementing a robust backup strategy easily
 - That includes handling of archive logs
- Managing flashback logs
- Upon recovery

Idea of the Recovery Area: It contains everything for successful recovery

Database Area
db_create_file_dest

Recovery Area
db_recovery_file_dest



Database Creation: Initialization parameters

- Using **Database Area** together with **Recovery Area** makes **create database** easy:

```
[oracle@uhesse1 ~]$ cat initprima.ora

compatible=11.2.0.2
db_block_size=8192
db_name='prima'
db_create_file_dest='/u01/app/oracle/oradata'
db_recovery_file_dest='/u02/fra'
db_recovery_file_dest_size=2g
memory_target=640m
undo_management=auto
undo_tablespace=undotbs1
db_domain=''
diagnostic_dest='/u01/app/oracle/'
```

Database Creation: create database

- This creates the database with mirrored controlfiles and mirrored logfiles.
- **CONTROL_FILES** is maintained automatically

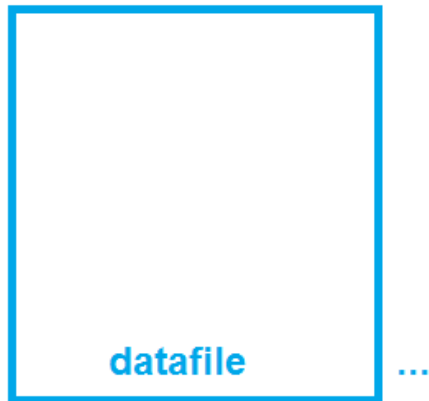
```
[oracle@uhesse ~]$ sqlplus / as sysdba

SQL> create spfile from
pfile='/home/oracle/skripte/initprima.ora';
SQL> startup nomount
SQL> create database undo tablespace undotbs1
      default tablespace users
      default temporary tablespace temp
      character set al32utf8;
```

State after Database Creation

Database Area

`db_create_file_dest`



Recovery Area

`db_recovery_file_dest`



Mirrors got created automatically

Checking the **mirrors**: Controlfiles

```
SQL> select name from v$controlfile;
```

```
NAME
```

```
-----  
/u01/app/oracle/oradata/PRIMA/controlfile/o1_mf_b1hpg83v_.ctl  
/u02/fra/PRIMA/controlfile/o1_mf_b1hpg869_.ctl
```

```
SQL> select value from v$spparameter where name='control_files';
```

```
VALUE
```

```
-----  
/u01/app/oracle/oradata/PRIMA/controlfile/o1_mf_b1hpg83v_.ctl  
/u02/fra/PRIMA/controlfile/o1_mf_b1hpg869_.ctl
```

Checking the **mirrors**: Logfiles

```
SQL> select group#,sequence#,bytes/1024/1024 as mb,members from v$log;
```

GROUP#	SEQUENCE#	MB	MEMBERS
1	1	100	2
2	0	100	2

```
SQL> select group#,member from v$logfile;
```

```
GROUP# MEMBER
```

```
-----  
1 /u01/app/oracle/oradata/PRIMA/onlinelog/o1_mf_1_b1hpg885_.log  
1 /u02/fra/PRIMA/onlinelog/o1_mf_1_b1hpg8bh_.log  
2 /u01/app/oracle/oradata/PRIMA/onlinelog/o1_mf_2_b1hpg8dr_.log  
2 /u02/fra/PRIMA/onlinelog/o1_mf_2_b1hpg8n8_.log
```


Finalize Database Creation

```
SQL> @?/rdbms/admin/catalog
```

```
SQL> @?/rdbms/admin/catproc
```

Do you want to see that for the next approx. 15 minutes?

```
...  
Synonym created.  
  
Grant succeeded.  
  
View created.  
...
```

Turn on ARCHIVELOG Mode

```
SQL> select log_mode from v$database;
```

```
LOG_MODE
```

```
-----  
NOARCHIVELOG
```

```
SQL> shutdown immediate
```

```
SQL> startup mount
```

```
SQL> alter database archivelog;
```

```
SQL> select log_mode from v$database;
```

```
LOG_MODE
```

```
-----  
ARCHIVELOG
```

Check generation of archive logs

Subfolders created automatically

```
SQL> select name from v$archived_log;
```

```
no rows selected
```

```
SQL> alter system switch logfile;
```

```
System altered.
```

```
SQL> select name from v$archived_log;
```

```
NAME
```

```
-----  
/u02/fra/PRIMA/archivelog/2014_09_16/o1_mf_1_7_b1hvqdkg_.arc
```

Very useful: V\$RECOVERY_AREA_USAGE

```
SQL> select file_type,percent_space_used as "%USED",  
           number_of_files as "#" from v$recovery_area_usage;
```

FILE_TYPE	%USED	#
CONTROL FILE	.36	1
REDO LOG	9.77	2
ARCHIVED LOG	2.09	1
BACKUP PIECE	0	0
IMAGE COPY	0	0
FLASHBACK LOG	0	0
FOREIGN ARCHIVED LOG	0	0

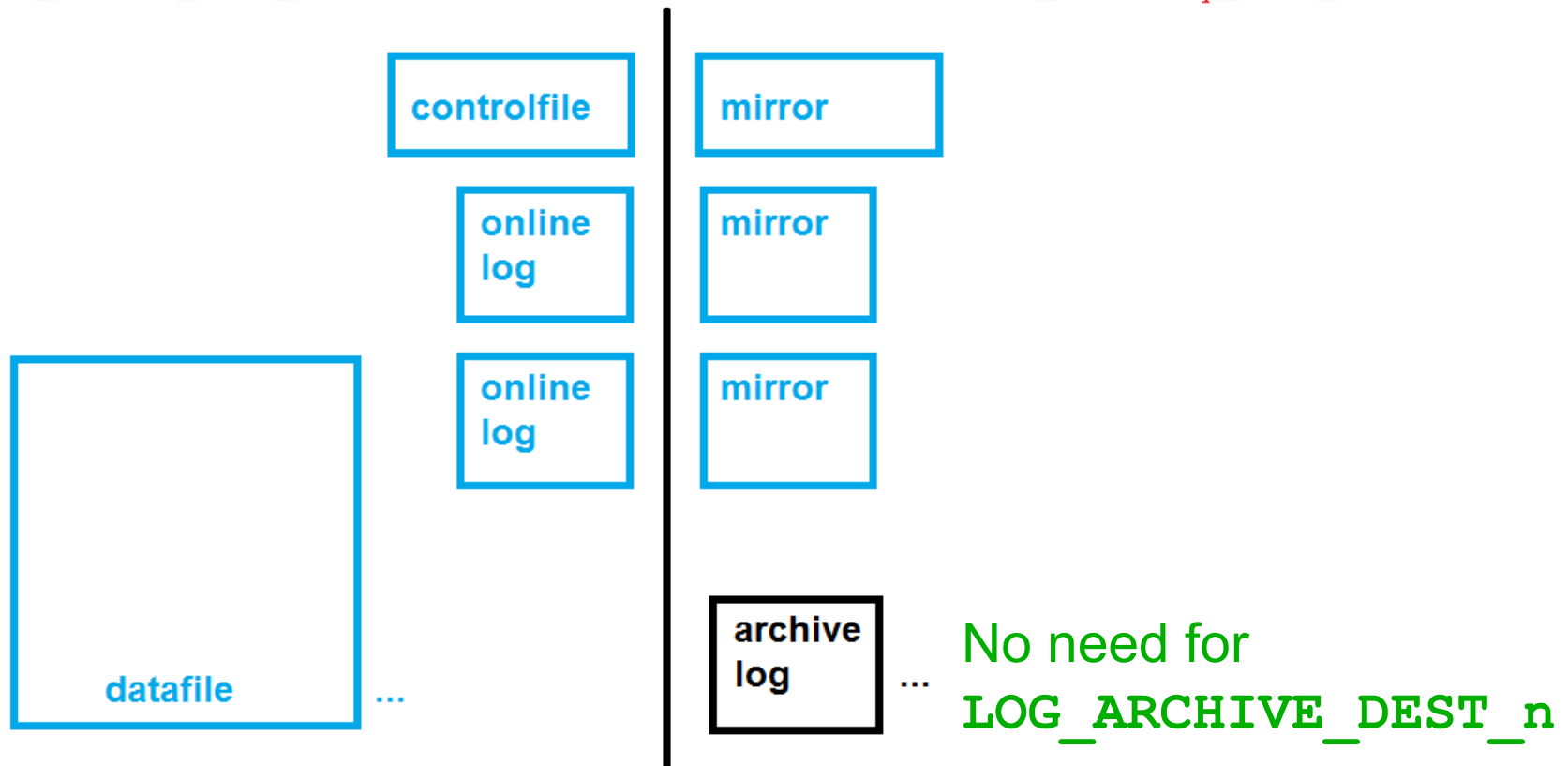
State after turning on ARCHIVELOG mode

Database Area

db_create_file_dest

Recovery Area

db_recovery_file_dest



Weird behavior when LOG_ARCHIVE_DEST_n is set additionally

```
SQL> alter system set log_archive_dest_2='location=/home/oracle/archive';  
SQL> select name from v$archived_log;
```

NAME

/u02/fra/PRIMA/archivelog/2014_09_17/o1_mf_1_7_b11b8z5b_.arc

```
SQL> alter system switch logfile;
```

```
SQL> select name from v$archived_log;
```

NAME

/u02/fra/PRIMA/archivelog/2014_09_17/o1_mf_1_7_b11b8z5b_.arc
/home/oracle/archive/1_8_856715038.dbf

```
SQL> alter system set  
log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST';
```

Backups go automatically into the Recovery Area: Autobackups and Backupsets

```
[oracle@uhessel ~]$ rman target /  
RMAN> configure controlfile autobackup on;  
RMAN> backup database;  
RMAN> list backup of database;
```

List of Backup Sets

=====

BS Key	Type	LV	Size	Device	Type	Elapsed Time	Completion Time
1	Full		227.27M	DISK		00:00:01	2014-09-16 10:29:25
	BP Key: 1		Status: AVAILABLE		Compressed: NO	Tag:	
	TAG20140916T102924						
	Piece Name:	/u02/fra/PRIMA/backupset/2014_09_16/ o1_mf_nnndf_TAG20140916T102924_b1hx74g6_.bkp					

Backups go automatically into the Recovery Area: Image Copies

```
RMAN> backup as copy database;  
RMAN> list copy of database;
```

List of Datafile Copies

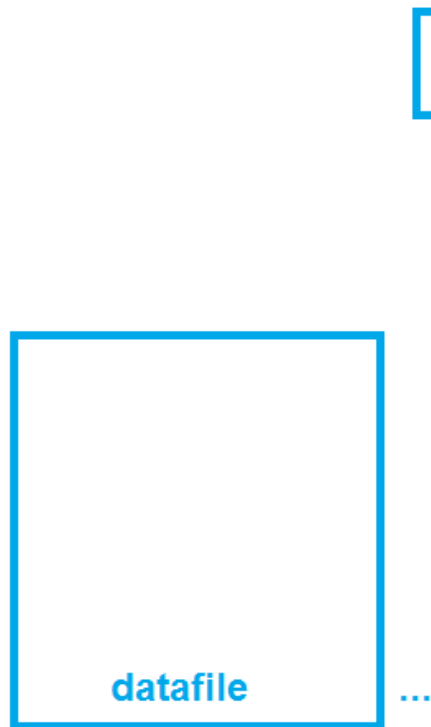
=====

Key	File S	Completion Time	Ckp SCN	Ckp Time
1	1 A	2014-09-16 10:43:47	223612	2014-09-16 10:43:46
	Name: /u02/fra/PRIMA/datafile/o1_mf_system_b1hy2254_.dbf			
3	2 A	2014-09-16 10:43:48	223615	2014-09-16 10:43:48
	Name: /u02/fra/PRIMA/datafile/o1_mf_sysaux_b1hy24bz_.dbf			
2	3 A	2014-09-16 10:43:48	223613	2014-09-16 10:43:47
	Name: /u02/fra/PRIMA/datafile/o1_mf_undotbs1_b1hy2381_.dbf			
4	4 A	2014-09-16 10:43:49	223616	2014-09-16 10:43:49
	Name: /u02/fra/PRIMA/datafile/o1_mf_users_b1hy25f1_.dbf			

State after Backups have been taken

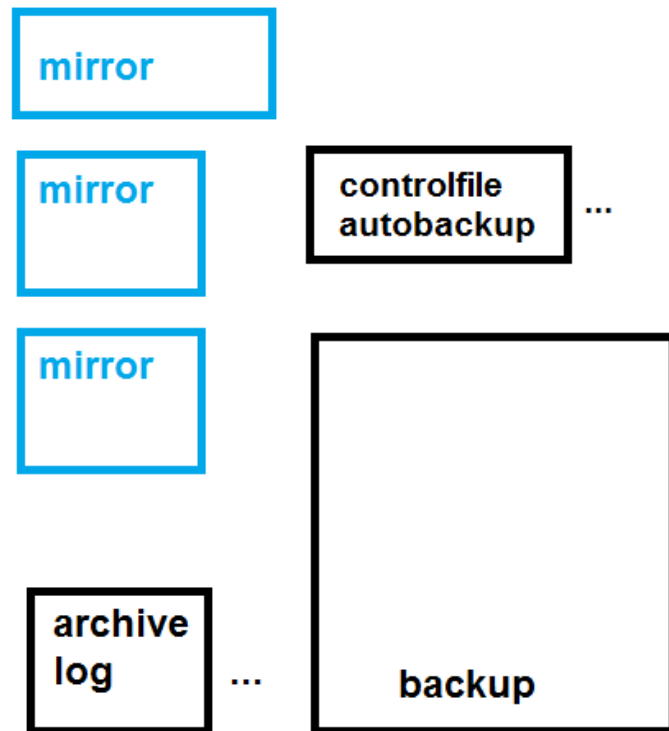
Database Area

db_create_file_dest



Recovery Area

db_recovery_file_dest



No need
for
FORMAT
clause

Treatment of obsolete files in the Recovery Area

```
RMAN> show retention policy;
```

```
RMAN configuration parameters for database with db_unique_name PRIMA are:  
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
```

Obsolete backups & archive logs get deleted automatically
– but only if space gets scarce. Let's try to see that live!

```
SQL> select file_type,percent_space_used as "%USED",  
percent_space_reclaimable as "%OBSO",number_of_files as "#"  
from v$recovery_area_usage;
```

FILE_TYPE	%USED	%OBSO	#
CONTROL FILE	.36	0	1
REDO LOG	9.77	0	2
ARCHIVED LOG	2.09	2.09	1
BACKUP PIECE	11.83	11.47	3
IMAGE COPY	28.81	0	4
FLASHBACK LOG	0	0	0

Flashback Logs must be in *Fast Recovery Area*

```
SQL> select open_mode, flashback_on from v$database;
```

OPEN_MODE	FLASHBACK_ON
-----	-----
READ WRITE	NO

```
SQL> alter database flashback on;
```

Database altered.

```
SQL> show parameter flashback
```

NAME	TYPE
-----	-----
VALUE	
-----	-----
db_flashback_retention_target	integer
1440	

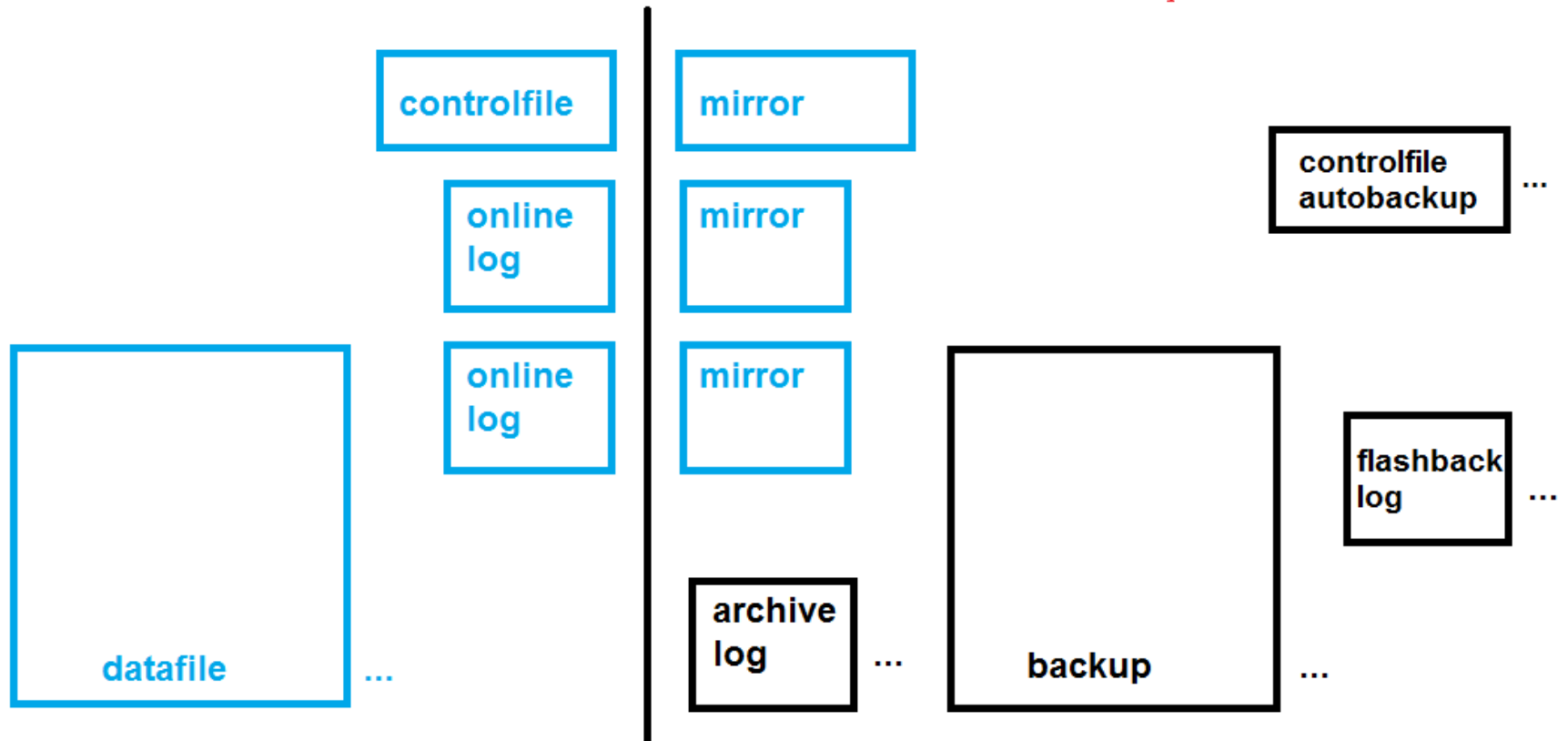
State after turning on Flashback Logging

Database Area

db_create_file_dest

Recovery Area

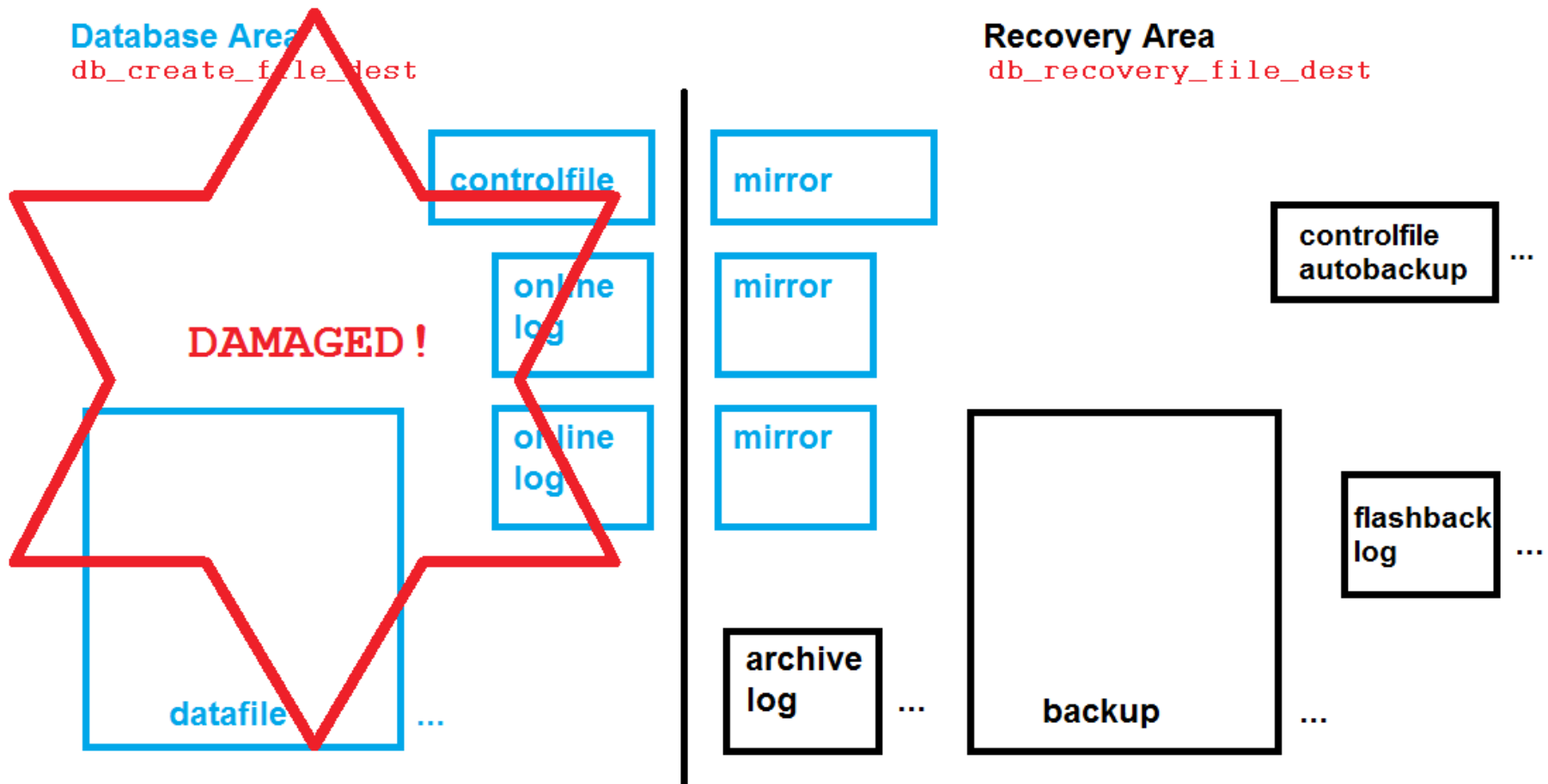
db_recovery_file_dest



Meaning of DB_RECOVERY_FILE_DEST_SIZE

- Logical limit that this database is allowed to consume for
 - Mirrors, Backups, Archive logs, Flashback logs
- *Will* be consumed almost entirely
- Obsolete files get deleted automatically *on space pressure*
- RMAN retention policy determines what is obsolete
- Goal is taking only backups regularly and let the Recovery Area take care of itself
- Flashback retention target is just a wish
- Flashback logs may be overwritten unless a *guaranteed* restore point requires them to be kept

What if the Database Area is gone? Recovery Area to the rescue!



Database Area gone: 1. restore controlfile

Database Area

db_create_file_dest

controlfile

mirror

mirror

mirror

archive
log ...

Recovery Area

db_recovery_file_dest

controlfile
autobackup ...

flashback
log ...

backup ...

```
RMAN> restore controlfile from
'/u02/fra/PRIMA/controlfile/o1
_mf_9zvthz0k_.ctl';
alter database mount;
```

Database Area gone: 2. restore database or switch database to copy, then recover

Database Area

db_create_file_dest

controlfile

mirror

mirror

mirror

archive
log ...

Recovery Area

db_recovery_file_dest

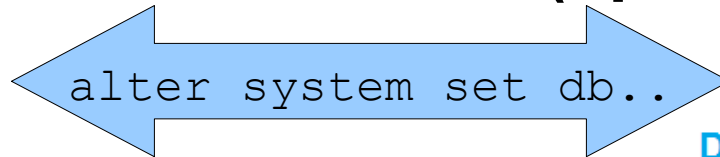
controlfile
autobackup ...

flashback
log ...

backup ...

```
RMAN> switch database to copy;  
RMAN> recover database;  
RMAN> alter database open;
```


Database Area gone: 3. switch back to former Database Area (optionally)



Recovery Area

db_recovery_file_dest

Database Area

db_create_file_dest

```
SQL> alter database
clear logfile group
1; c/1/2 r

RMAN> backup as copy
database;
RMAN> shutdown
immediate;
RMAN> startup mount;
RMAN> switch
database to copy;
```

controlfile

mirror

online log

mirror

online log

mirror

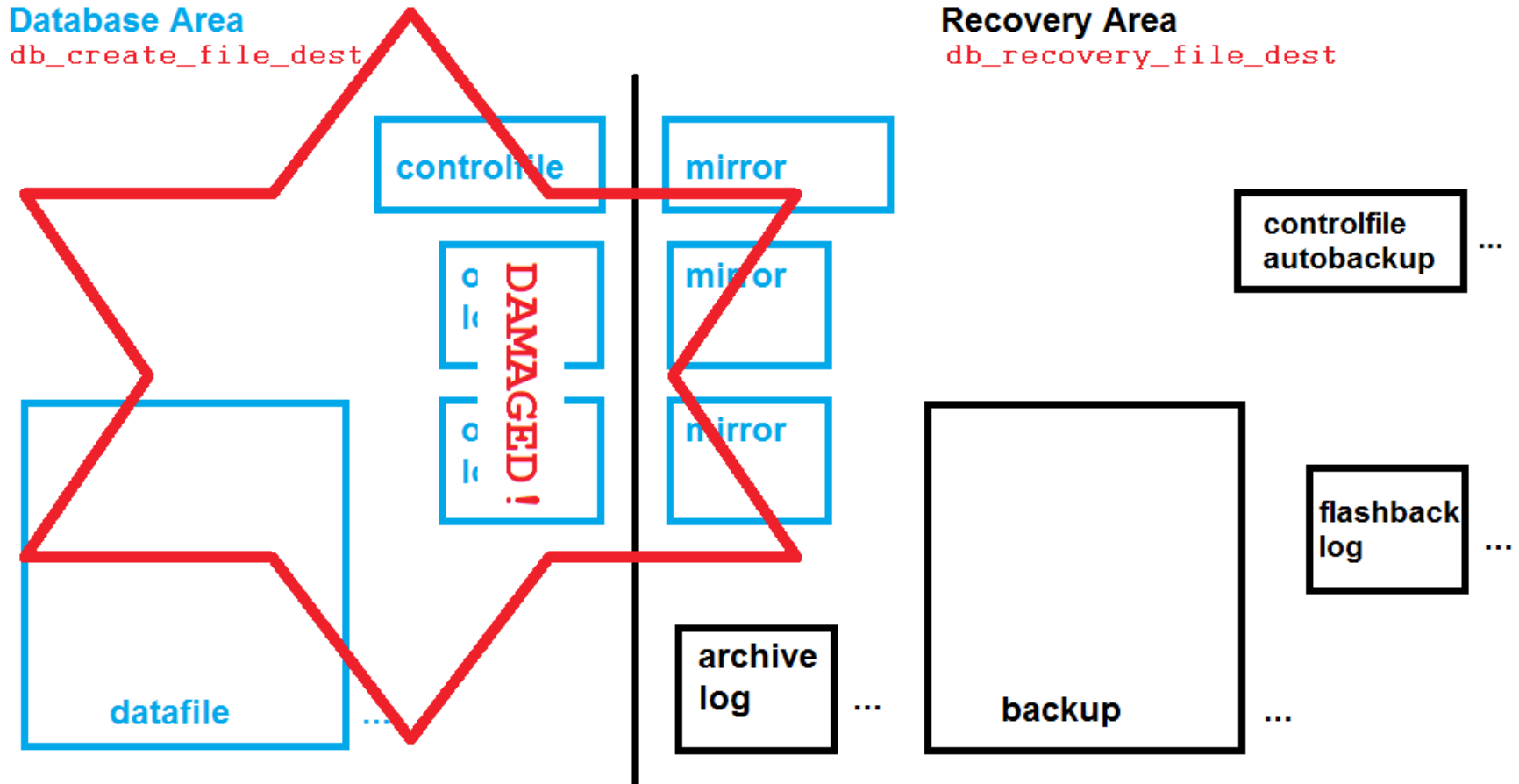
archive log ...

datafile ...

controlfile
autobackup ...

flashback
log ...

Disaster Recovery: No mirrors of controlfile and logfiles available



Disaster Recovery made easier by using the Recovery Area

No need for `set DBID`

Archive logs are automatically cataloged

```
RMAN> restore controlfile from autobackup;  
RMAN> alter database mount;  
RMAN> restore database;  
RMAN> recover database;
```

Take missing sequence# (e.g. 16) from above and then

```
RMAN> run{set until sequence 16 thread 1;  
        recover database;  
        alter database open resetlogs;}
```

Data loss cannot be avoided because onlinelogs have been completely (!) damaged

Thank you for your attention!