



RECM

Version 1.0

Recovery Manager for PostgreSQL

Installation and User's guide
(Revision 2021-12)

PREFACE

Purpose

This document describes how to install, configure, and use the PostgreSQL Recovery Manager tool.

Audience

This document is intended for system administrators, database administrators (DBAs) and developers who want to backup and restore easily PostgreSQL database.

Conventions used in this document

This document uses the following conventions for special notices:

A note presents information that is important,

Note:

Typographical conventions

Bold	Used for reserved keywords
Monospace blue	Used for command examples
Monospace Gray	Comments within examples (System Output, errors,)
[...]	Optional values
{...}	Alternate
	Separator that indicate alternate choice
...	Output message omitted voluntarily

Table of content

PREFACE.....	2
Purpose.....	2
Audience.....	2
Conventions used in this document.....	2
REVISION HISTORY.....	6
INTRODUCTION.....	7
INSTALLATION.....	8
Binaries.....	8
Requirements.....	8
Environment variables.....	8
PRODUCT OVERVIEW.....	9
Description.....	9
Supported configuration.....	10
CONFIGURATION.....	10
COMMANDS.....	11
COMMAND LINE ARGUMENTS.....	12
ENVIRONMENT VARIABLES.....	12
Commands by alphabetic order.....	13
BACKUP CONFIG.....	14
Options and qualifier:.....	14
Example:.....	14
BACKUP FULL.....	15
Options and qualifiers.....	15
Example:.....	15
BACKUP META.....	16
Options and qualifiers.....	16
Example:.....	16
BACKUP WAL.....	17
Options and qualifiers.....	17
Example:.....	17
CLRSCR.....	18
Options and qualifiers.....	18
CREATE DEPOSIT.....	19
Options and qualifiers.....	19
Description.....	19
CONNECT CLUSTER.....	20
Options and qualifiers.....	20
Requirement.....	20
Example:.....	20
CONNECT DEPOSIT.....	21
Options and qualifiers.....	21
Description.....	21
Examples:.....	21
CREATE RESTORE POINT.....	22
Options and qualifiers.....	22
Description.....	22
DELETE BACKUP.....	23
Options and qualifiers.....	23
Description.....	23

Example:.....	23
DELETE CLUSTER.....	24
Options and qualifiers.....	24
Description.....	24
Example:.....	24
DELETE RESTORE POINT.....	25
Options and qualifiers.....	25
Description.....	25
Example:.....	25
DISCONNECT CLUSTER.....	26
Options and qualifiers.....	26
DISCONNECT DEPOSIT.....	27
Options and qualifiers.....	27
DUPLICATE FULL.....	28
Options and qualifiers.....	28
Description.....	28
Example:.....	29
EXIT.....	30
HISTORY.....	31
LIST BACKUP.....	32
Options and qualifiers.....	32
Description.....	32
Examples:.....	33
LIST CLUSTERS.....	34
Options and qualifiers.....	34
Description.....	34
Example:.....	34
LIST RESTORE POINT.....	35
Options and qualifiers.....	35
Description.....	35
Example:.....	35
LIST WAL.....	36
Options and qualifiers.....	36
Description.....	36
MODIFY CLUSTER.....	37
Options and qualifiers.....	37
Description.....	37
MODIFY BACKUP.....	39
Options and qualifiers.....	39
Description.....	39
QUIT.....	40
Options and qualifiers.....	40
REGISTER CLUSTER.....	41
Options and qualifiers.....	41
Description.....	41
REGISTER FILES.....	42
Options and qualifiers.....	42
Description.....	42
RESTORE CONFIG.....	43
Options and qualifiers.....	43

Description.....	43
Full example for common usage.....	44
Create a deposit an register our first postgresQL engine.....	45

REVISION HISTORY

The following table presents the revision history of this document

Release	Date	Description
1	September 2021	Initial release.

INTRODUCTION

RECM is a tool dedicated perform PostgreSQL database backups, restores and duplicates.

Features :

Backups

- Perform FULL backup
- Perform WAL backup
- Perform METADATA backup
- Perform Configuration files backups
- Delete WAL files when they are backuped

Restore

- Perform FULL restore
- Perform Partial restore (restore at database,schema or table level)
- Perform Point in time recovery
 - By a date
 - By a restore point
 - By a LSN
- Perform DUPLICATES
- Restore a configuration file

INSTALLATION

RECM is available for Unix/Linux platform only.

Binaries

To install, visit our 'download page' from '<https://pgrecm.org/download>'.

Example:

```
#> yum install recm
```

Requirements

RECM require a PostgreSQL database to run. The database is a repository that can be used by all PostgreSQL databases version 12 and above.

RECM need to have access to a network share to save/restore backups.

RECM use PostgreSQL LIBPQ library, and must be used locally on the server you want to manage. You must use a local session to use RECM.

List of packages required :

Package Name	Description
PostgreSQL libpq	C application programmer's interface to PostgreSQL
libzip	ZIP file management (Version 1.5 minimum)

Environment variables

The following environment variables can be used, but are optional :

RECM_CONFIG	Change default placement of the 'recm.conf' file. By default, the file is '<HOME>/.recm/recm.conf'
RECM_META_DIR	By default, META generated files are stored in directory '<PGDATA>/recm_meta'. Setting this variable will save generated files within the defined directory.

PRODUCT OVERVIEW

Description

The RECM tool allow you to create hot backups, and allow you to perform restore.
This tools present some interesting feature that can help you choose RECM.

Feature	Advantage
Remove indexes from backup	Remove index from backup. You may create backup twice as smaller that other tool. On some backups, indexes may represent half of the database size. Restriction: Removed indexes backup cannot be use to create a standby database.
Reindex at restore time	At restore time, reindex command is executed to recreate omitted indexes at backup time.
Partial restore	RECM allow you to perform partial restore from any FULL backup. It is very easy to restore only one table, one schema or one database.
Restore	Restore never start on a NON empty PGDATA. It is impossible to perform a restore on directory that is not empty.

Remark:

A backup full without indexes may not be useable for creating a standby instance.

Supported configuration

RECM can be used with PostgreSQL version 12 and later.

CONFIGURATION

RECM need a PostgreSQL database as a repository called 'deposit'.
A deposit a database where RECM store all informations for managing retention of backups. The deposit allow you to perform 'partial restore' and FULL backup without indexes.

COMMANDS

The command is composed into three different kinds of keywords.
You have

- The verb (the command itself)
- Options (parameter without values)
- Qualifiers (parameter with a value)
- The string delimiter is the double quote.

Each command can be ended by ';' character. Arguments and Qualifiers of command must start with a slash '/' character.

Options

An option is a keyword that does not have any value (a switch in Unix world). The best example is the /verbose option. To activate the verbose option, you need to add the 'verbose' option in the command.

Qualifiers

A Qualifier is an option that needs a value. A value is provided by the equal sign.

Example:

```
BACKUP FULL /NOINDEX /VERBOSE /TAG="MY_TAG";
```

keyword	Description
Backup full	Invoke backup FULL command
/noindex	Option : Remove indexes from backup
/verbose	Option : Enable verbose mode
/tag="MY_TAG"	Qualifier : Change default TAG name by a provided value

COMMAND LINE ARGUMENTS

RECM accept the following options and parameters you can use on the command line.

Short name	Long name	Mandatory	Description
-c “...”	--command=”...”	No	Execute a command, than exit
-v	--verbose	No	Verbose mode
-t	--target=”S”	No	Use a predefined cluster
-d	--deposit=”S”	No	Use a predefined Deposit
-V	--version	No	Display RECM version only
-h	--help	No	Display a short help usage.

ENVIRONMENT VARIABLES

Variable Name	Mandatory	Description
RECM_PGBIN	No	Force the tool to use specific binary version
PGDATA	No	PostgreSQL data directory
RECM_CONFIG	No	Change default placement of the 'recm.conf' file. By default, the file is '<HOME>/.recm/recm.conf'
RECM_METADIR	No	By default, META generated files are stored in directory '<PGDATA>/recm_meta'. Setting this variable will save generated files within the defined directory.
RECM_RESTORE_COMMAND	No	You may customize the 'restore_command' when you perform a 'duplicate' or a 'restore'

Commands by alphabetic order

BACKUP CONFIG

Perform a backup of all the configuration files. Configuration files are already part of any FULL backup. But, this backup of configuration files have it's own retention

Options and qualifier:

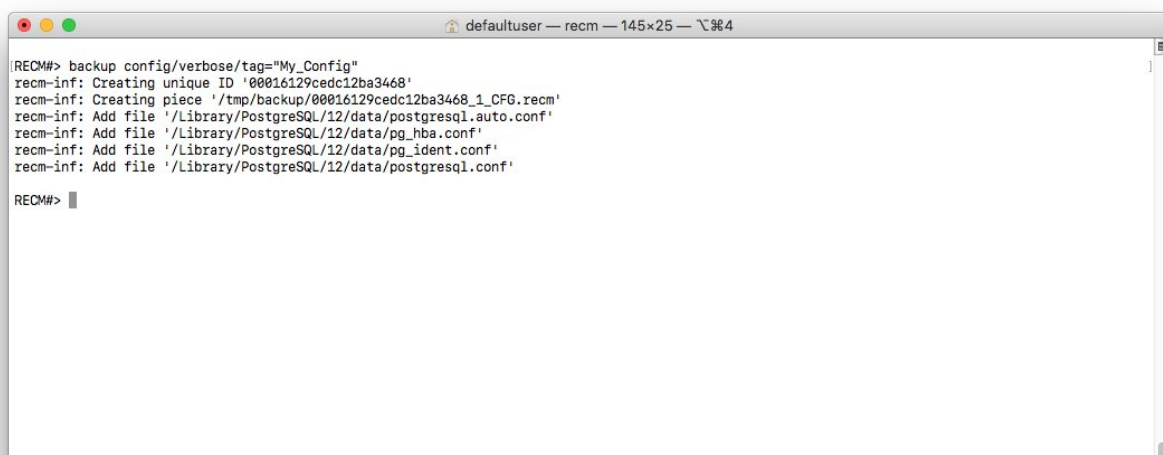
Option/Qualifier	Mandatory	Description
/verbose	no	Change log details
/tag=<string>	no	Change default TAG of the backup

Perform a backup of all the configuration files. Configuration backups are never used at restore time. The restore config must be initiated voluntarily. A configuration backup contain the following list of files by default :

Parameter	Default File Name	Description
-	PostgreSQL.conf	Default configuration file
-	PostgreSQL.auto.conf	Default configuration file
hba_file	pg_hba.conf	Hba file
ident_file	pg_ident.conf	Identity file
ssl_ca_file	(no default)	SSL certificates authority
ssl_dh_params_fil	(no default)	SSL certificates parameter filep

Optionaly, you can add other specific files to the backup by defining the 'addfiles' to configuration (See 'set configuration' for more details)

Example:



```
defaultuser — recm — 145x25 — ㄟ ㄣ 4
[RECM#> backup config/verbose/tag="My_Config"
recm-inf: Creating unique ID '00016129cedc12ba3468'
recm-inf: Creating piece '/tmp/backup/00016129cedc12ba3468_1_CFG.recm'
recm-inf: Add file '/Library/PostgreSQL/12/data/postgresql.auto.conf'
recm-inf: Add file '/Library/PostgreSQL/12/data/pg_hba.conf'
recm-inf: Add file '/Library/PostgreSQL/12/data/pg_ident.conf'
recm-inf: Add file '/Library/PostgreSQL/12/data/postgresql.conf'
[RECM#> ]
```

BACKUP FULL

Perform a FULL online backup of the whole PGDATA directory. The command use PostgreSQL functions 'pg_start_backup()' and 'pg_stop_backup()'.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/cleanup	No	Invoke 'delete backup/obsolete'. Usually used with '/verify' option.
/directory=<string>	no	Set a different location for the backup
/noindex	no	Remove all user indexes from the backup. At restore time, the indexes will be recreated by a 'reindex'
/verbose	no	Change log details
/verify	no	Invoke 'verify backup' command after the backup process.
/tag=<string>	no	Change default TAG of the backup
/wal	No	Invoke a 'backup wal' command after the backup. The backup ID will be different, but the TAG will remain the same.

Example:

```
defaultuser — recm — 145x25 — ㄟ 364
[RECM#> backup full /tag="my_tag" /noindex /wal;
recm-inf: Backup UID : 0001612b766718af4278
recm-inf: Total pieces ..... : 13
recm-inf: Total files backedup..... : 2407
recm-inf: Total Backup size ..... : 147156175 (140 MB)
recm-inf: Total bytes skipped (Indexes) . : 3179813 (3 MB)
recm-inf: Total files skipped (Indexes) . : 63
NOTICE: all required WAL segments have been archived
*****
[RECM#> ]
```

BACKUP META

Create a backup set of database dump.

This operation invoke the 'pg_dump' utility to create sql file. This action never export data. That's why the files can be stored in the PGDATA folder, subfolder 'recm_meta' by default.

By default, the dump files are generated for all existing databases.

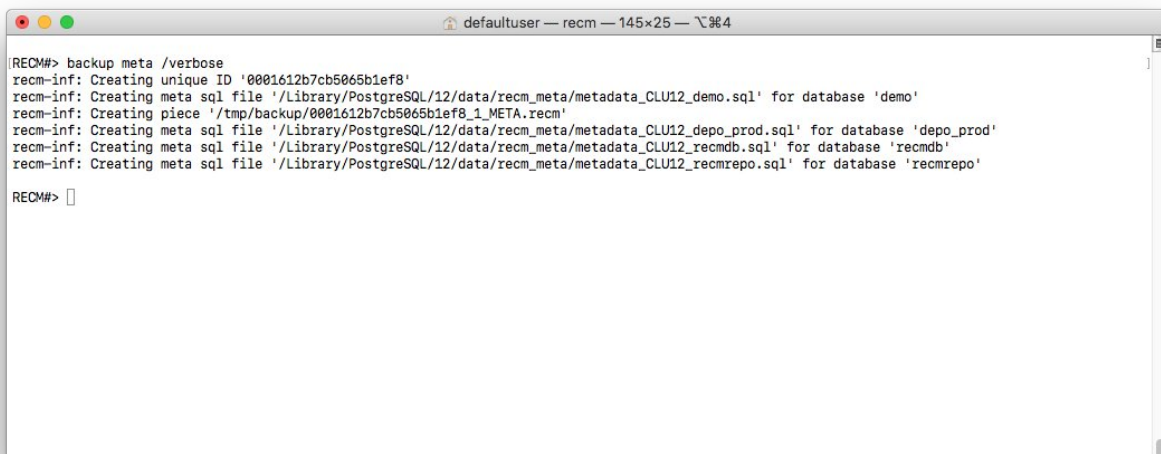
Optionally, you can define environment variable 'RECM_METADIR' to place generated files in a different folder. If you define it, you have to specify the full path directory.

You may have to configure '.pgpass' file for pg_dump working properly.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/db=<string>	no	Select databases to select or omit. Example: /db="+DB1,+DB2" Will backup only DB1 and DB2 database. /db="-DB1" Will backup ALL databases except DB1.
/verbose	no	Change log details
/tag=<string>	no	Change default TAG of the backup

Example:



```
defaultuser — recm — 145x25 — ㄟ ㄏ 4
[RECM#> backup meta /verbose
recm-inf: Creating unique ID '0001612b7cb5065b1ef8'
recm-inf: Creating meta sql file '/Library/PostgreSQL/12/data/recm_meta/metadata_CLU12_demo.sql' for database 'demo'
recm-inf: Creating piece '/tmp/backup/0001612b7cb5065b1ef8_1_META.recm'
recm-inf: Creating meta sql file '/Library/PostgreSQL/12/data/recm_meta/metadata_CLU12_depo_prod.sql' for database 'depo_prod'
recm-inf: Creating meta sql file '/Library/PostgreSQL/12/data/recm_meta/metadata_CLU12_recmdb.sql' for database 'recmdb'
recm-inf: Creating meta sql file '/Library/PostgreSQL/12/data/recm_meta/metadata_CLU12_recmrepo.sql' for database 'recmrepo'
[RECM#> ]
```

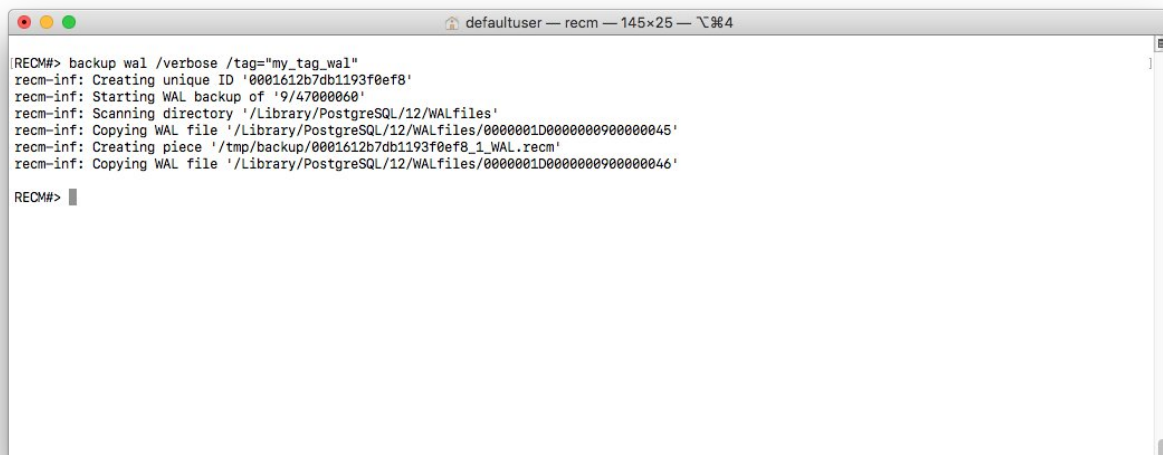

BACKUP WAL

Perform a backup of all WAL files, and optionally remove WAL files from directory once backedup.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/directory=<string>	no	Set a different location for the current WAL backup
/switch	no	Invoke a switch wal before to start the backup. By default,a checkpoint is generated.
/verbose	no	Change log details
/nochkpt	No	Disable checkpoint when using '/switch'
/tag=<string>	no	Change default TAG of the backup
/delete	no	Force deletion of WAL files after backup. This option override the configuration.
/nodelete	no	Skip WAL deletion. This option override the configuration. (See 'sshow cluster' for more details)

Example:



```
defaultuser — recm — 145x25 — ㄟ 364

[RECM#> backup wal /verbose /tag="my_tag_wal"
recm-inf: Creating unique ID '0001612b7db1193f0ef8'
recm-inf: Starting WAL backup of '9/47000060'
recm-inf: Scanning directory '/Library/PostgreSQL/12/WALfiles'
recm-inf: Copying WAL file '/Library/PostgreSQL/12/WALfiles/0000001D0000000900000045'
recm-inf: Creating piece '/tmp/backup/0001612b7db1193f0ef8_1_WAL.recm'
recm-inf: Copying WAL file '/Library/PostgreSQL/12/WALfiles/0000001D0000000900000046'

[RECM#> ]
```

CLRSCR

Clear screen.

Options and qualifiers

None.

CREATE DEPOSIT

RECM require a deposit. A PostgreSQL database is used for this purpose. The deposit maintains backup retention, restore point and history for all your registered PostgreSQL databases.

A deposit is mandatory.

The 'create deposit' command creates objects within the specified database.

Be careful, the database content will be erased. It is recommended to use a dedicated database for this purpose. The command will create the database.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/usr=<string>	yes	User that will be created.
/pwd=<string>	yes	User password
/db=<string>	yes	Database to create(or reuse)
/force	no	Force deletion of database to have a fresh installation.
/verbose	no	Enable more details.

Description

The best way to create a deposit is to use a dedicated database on a PostgreSQL engine.

Requirements :

- Choose a PostgreSQL database greater or equal to version 12.
- Use a database name that helps you to remember its usage.
- Don't use a database for another usage
- Need around 10MB per registered PostgreSQL engine, depending on the retention

CONNECT CLUSTER

Connect to a PostgreSQL cluster. If you are already connected, the previous connection is automatically close.

(See also command 'connect target')

Options and qualifiers

Option/Qualifier	Mandatory	Description
/host=<string>	Yes	Hostname or host IP to connect to.
/port=<string>	Yes	PostgreSQL listen port number
/usr=<string>	No	User to connect to (must be SUPERUSER)
/pwd=<string>	No	Password associated
/db=<string>	No	Deposit Database name.

Requirement

- The configuration parameter 'cluster_name' must be set.

Example:

CONNECT DEPOSIT

This command will connect to DEPOSIT. The previous connection is implicitly close, because you cannot connect to more than one DEPOSIT at a time.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/name=<string>	No	Name of the DEPOSIT. This is the name given at creation time.
/host=<string>	No	Hostname or host IP to connect to.
/port=<string>	No	PostgreSQL listen port number
/usr=<string>	No	User to connect to (must be SUPERUSER)
/pwd=<string>	No	Password associated
/db=<string>	No	Default database. By default, the 'postgres' database is used

Description:

The first time you connect to the deposit, you need to use qualifiers '/host','/port','/usr','/pwd' and '/db'.

At next connection, you will be able to use the short way, by using the '/name' qualifier only.

Examples:

```
recm — recm — 145x11 — ㄟ#4
{CluName}> connect deposit /host=192.168.1.62/port=5555/usr=recmadm/pwd=recmadm/db=recm_db
recm-inf: Connected to deposit 'REP012'
{CluName}> █
```

```
recm — recm — 145x11 — ㄟ#4
{CluName}> connect deposit/name=BADREPO
recm-err(104): Deposit name unknown. Cannot connect to 'BADREPO'
{CluName}> connect deposit/name=REP012
recm-inf: Connected to deposit 'REP012'
{CluName}> █
```

CREATE RESTORE POINT

PostgreSQL allow you to create restore point, via the 'pg_create_restore_point' function. RECM call the function and keep trace in the deposit the necessary information for doing a restore, or a partial restore.

The PERMANENT restore point option guaranty to keep the required backups, independently of the retention setting.

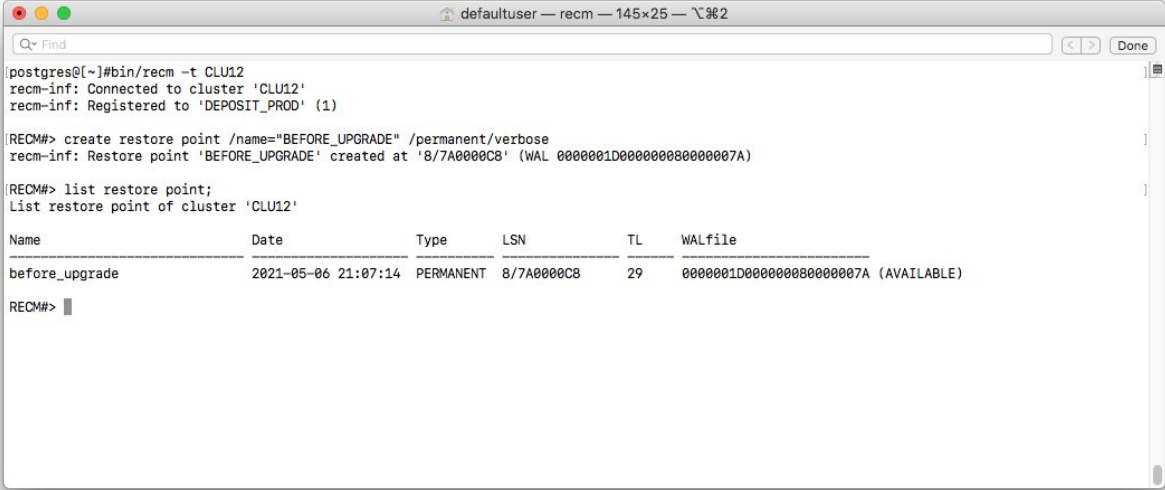
Options and qualifiers

Option/Qualifier	Mandatory	Description
/name=<string>	Yes	Name of the restore point.
/temporary	No	By default,a restore point is flagged as 'temporary'. That means, when the associated backup will be expired, the restore point will be expired also.
/permanent	No	When adding '/permanent' option, the associated backup will be kept while the restore point is not explicitly removed.
/verbose	No	Enable more details.

Description

Some useful information are kept in deposit, like the LSN, the date and the WAL file.

Example:



```
postgres@[-]#bin/recm -t CLU12
recm-inf: Connected to cluster 'CLU12'
recm-inf: Registered to 'DEPOSIT_PROD' (1)

[RECM#> create restore point /name="BEFORE_UPGRADE" /permanent/verbose
recm-inf: Restore point 'BEFORE_UPGRADE' created at '8/7A0000C8' (WAL 0000001D000000080000007A)

[RECM#> list restore point;
List restore point of cluster 'CLU12'

Name                Date                Type      LSN          TL      WALfile
-----
before_upgrade      2021-05-06 21:07:14  PERMANENT  8/7A0000C8   29      0000001D000000080000007A (AVAILABLE)

[RECM#>
```

DELETE BACKUP

This command delete backups. Usually, we will delete 'INCOMPLETE' or 'OBSOLETE' backups. But in some case, we can delete other backups.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/incomplete	No	Delete 'incomplete' backups.
/obsolete	No	Delete 'obsolete' backups.
/failed	No	Delete 'failed' backups.
/uid=<UID> /uid=<UID1>,<UID2>	No	Delete one, or more backups, specified by the backup ID
/noremove	No	Delete backup definition in the DEPOSIT, but keep physical files.
/before=<date>	No	You can reduce the range of the deletion, before a date
/after=<date>	No	You can reduce the range of the deletion, after a date

Description

Without one of the following options : '/incomplete','/obsolete','/failed', the delete backup operation will not delete anything.

Remarks:

- You cannot delete backups of other cluster.
- The command does not ask for confirmation to proceed.
- The only way to delete 'active' backups is to use the qualifier '/uid=<uid>'.

Example:

```
recm — recm — 145x28 — ™4
CLU12> list backup
List backups of cluster 'CLU12' (CID=1)
UID                Date                Type    Status    TL      Size  Tag
-----
000161d1604f392d67c8 2022-01-02 09:20:32 FULL    OBSOLETE  35      2 GB  toto
000161d16094102f5f20 2022-01-02 09:21:40 WAL     OBSOLETE  35      48 MB  toto
000161d1628919627190 2022-01-02 09:30:01 WAL     OBSOLETE  35      80 MB  WAL_cron1
000161d16e6e0a0c0198 2022-01-02 10:20:46 FULL    AVAILABLE 35      2 GB  FULL_daily
000161d16ea135e2a1c8 2022-01-02 10:21:37 WAL     AVAILABLE 35      64 MB  FULL_daily
000161d1709a1295eb58 2022-01-02 10:30:02 WAL     AVAILABLE 35      16 MB  WAL_cron1
000161d177762cf1d048 2022-01-02 10:59:19 FULL    AVAILABLE 35      2 GB  20220102_CLU12_FULL
000161d177a502e62448 2022-01-02 11:00:05 WAL     FAILED    35      64 MB  WAL_cron1
000161d177a51c2e7e28 2022-01-02 11:00:05 WAL     AVAILABLE 35      16 MB  20220102_CLU12_FULL

CLU12> delete backup/obsolete

CLU12> list backup
List backups of cluster 'CLU12' (CID=1)
UID                Date                Type    Status    TL      Size  Tag
-----
000161d16e6e0a0c0198 2022-01-02 10:20:46 FULL    AVAILABLE 35      2 GB  FULL_daily
000161d16ea135e2a1c8 2022-01-02 10:21:37 WAL     AVAILABLE 35      64 MB  FULL_daily
000161d1709a1295eb58 2022-01-02 10:30:02 WAL     AVAILABLE 35      16 MB  WAL_cron1
000161d177762cf1d048 2022-01-02 10:59:19 FULL    AVAILABLE 35      2 GB  20220102_CLU12_FULL
000161d177a502e62448 2022-01-02 11:00:05 WAL     FAILED    35      64 MB  WAL_cron1
000161d177a51c2e7e28 2022-01-02 11:00:05 WAL     AVAILABLE 35      16 MB  20220102_CLU12_FULL
CLU12>
```

DELETE CLUSTER

This command delete a cluster from the deposit. You cannot delete the cluster that is currently connected.

This command just delete the cluster entry in the repository. It does not affect the cluster itself.

Options and qualifiers

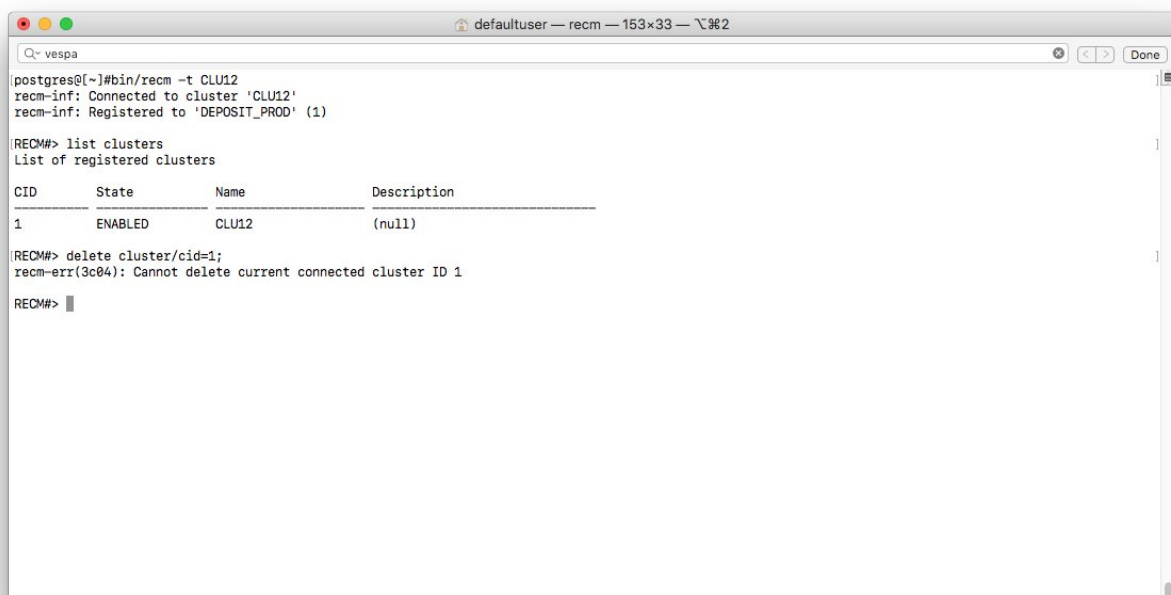
Option/Qualifier	Mandatory	Description
/cid=<ID>	No	ID of the cluster to delete. The id can be found via the command 'list cluster'.
/noremove	No	Delete backup definition in the DEPOSIT, but keep physical backup files.
/force	No	Deleting a cluster require the cluster to be disabled first. The force option allow you to remove it without taking care of the status of the cluster.
/verbose	No	

Description

The deletion of the cluster does not affect the potential remaining backup files.

The deletion is only logical (Just removed from the deposit). No alteration is done on the existing backups and the PostgreSQL cluster itself.

Example:



```
postgres@[~]#bin/recm -t CLU12
recm-inf: Connected to cluster 'CLU12'
recm-inf: Registered to 'DEPOSIT_PROD' (1)

RECM#> list clusters
List of registered clusters

CID      State      Name      Description
-----
1        ENABLED    CLU12     (null)

RECM#> delete cluster/cid=1;
recm-err(3c04): Cannot delete current connected cluster ID 1

RECM#> █
```


DELETE RESTORE POINT

This command delete a restore point from the deposit. This does not do anything on the target instance.

Options and qualifiers

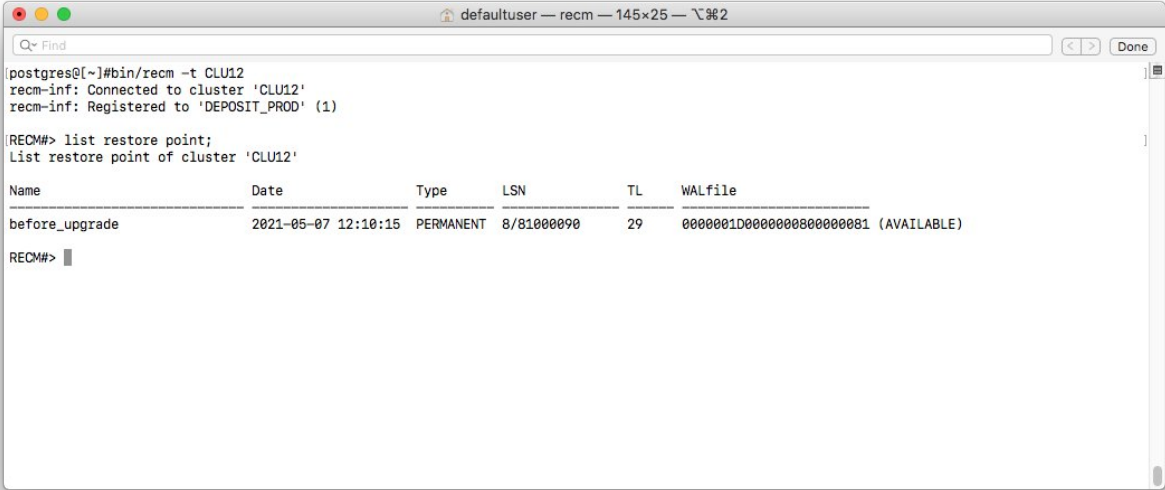
Option/Qualifier	Mandatory	Description
/name=<name>	Yes	Name of the restore point to remove. You can have the full list of restore point by invoking the command 'list restore point'.
/verbose	No	

Description

The deletion of the restore point is done only within the deposit. It is not necessary to perform anything on the PostgreSQL database.

If the restore point was a 'Permanent' one, you may have to run 'verify backup' and 'delete backup/obsolete' commands, to cleanup old backups.

Example:



```
defaultuser — recm — 145x25 — ￼2
postgres@[~]#bin/recm -t CLU12
recm-inf: Connected to cluster 'CLU12'
recm-inf: Registered to 'DEPOSIT_PROD' (1)

[RECM#> list restore point;
List restore point of cluster 'CLU12'

Name                Date                Type      LSN          TL      WALfile
-----
before_upgrade      2021-05-07 12:10:15 PERMANENT  8/81000090   29      0000001D0000000800000081 (AVAILABLE)

[RECM#> ]
```

DISCONNECT CLUSTER

This command close the current connected cluster.

Options and qualifiers

None.

DISCONNECT DEPOSIT

This command close the current connected deposit.

Options and qualifiers

None.

DUPLICATE FULL

This command allow you to restore a whole PostgreSQL engine onto a different server/location.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/name="<NAME>"	Yes	Name of the new cluster to restore/create
/directory="<PATH>"	Yes	New PGDATA diectory. If the directory exist, it must be empt.
/port=<NUMBER>	Yes	Port number to use for the new cluster
/until="<DATE>"	No	Allow you to perform a Point in time recorery (PTR).
/rp="<RPNAME>"	No	Restore until a restore point name.
/lsn="<LSN>"	No	Restore until a given LSN
/uid=<UID>	No	UID of the full backup to use for restore.
/cid="<CID>"	No	Cluster source to use (Default is currently connected).
/verbose	No	
/norecover	No	Perform restore and stop before to launch the recovery process. This allow you to perform the recovery by yourself. Skip recovery imply no reindex if backup has been done with '/noindex' option.
/standby	No	Change settings at restore time to make the restored instance a 'Standby'

Description

After the restore of the all the files, the configuration file 'PostgreSQL.conf' and 'pg_hba.conf' are overwritten by a more simple configuration. This allow you to have free access to the restored database. If you plan to keep the new cluster, you may have to reconfigure those files to be close to your environment configuration.

The date format is defined by the configuration item '/date_format'. The default is 'YYYY-MM-DD-HH'. (See 'set config' command for more details)

Example:

```
  recm — recm — 145x25 — `⌘`4
```

```
RECM#> duplicate full /directory="/tmp/tmpdata"/port=6666  
recm-inf: Restoring backup from UID '000161cc23de2ec61348' (TL 33, Version 12, WAL '13/9300028')  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
  
Create link '/tmp/tmpdata/pg_tblspc/131219' with '/tmp/tmpdata/TBS0_6666'(rc=0 : 2)  
#####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
100.00% #####  
  
recm-inf: Recovery in progress (LSN ('13/9500000'))  
recm-inf: Recovery ended at WAL '00000022000001300000094', LSN '13/9500000'  
recm-inf: Database successfully restored.  
recm-inf: Connect using Host:port: localhost:6666  
  
RECM#>
```

EXIT

Simply exit from the RECM tool. (Same as 'quit' command)

HISTORY

Display previous command executed. The number of command kept depend on the configuration parameter '/history'. The default is 10.

LIST BACKUP

This command display a list of all created backups. By default, the backup of the current connected cluster is listed. By adding a qualifier/option, you may filter the backups to display.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/cfg	No	(filter)List configuration backups.
/nocfg	No	(filter) Exclude configuration backups
/full	No	(filter)List FULL backups.
/nofull	No	(filter) Exclude FULL backups
/meta	No	(filter)List METADATA backups.
/nometa	No	(filter) Exclude METADATA backups.
/wal	No	(filter)List WAL backups.
/nowal	No	(filter) Exclude WAL backups.
/available	No	(filter) List only AVAILABLE backups
/obsolete	No	(filter) List only OBSOLETE backups
/incomplete	No	(filter) List only INCOMPLETE backups
/tag='<STRING>'	No	(filter) List backups where tag contain '<STRING>'
/before='<DATE>'	No	(filter)List all backups prior the date.
/after='<DATE>'	No	(filter)List all backups after the date.
/source='<STRING>'	No	(filter) list backups from an other cluster by it's name
/cid=<ID>	No	(filter)List backups from an other cluster by it's ID.

Description

You can use multiple options to filter on.

The date format is defined by the configuration item '/date_format'. The default is 'YYYY-MM-DD-HH'. (See 'set config' command for more details)

The order is always the backup date.

Examples:

Display WAL backups that are obsolete

```
defaultuser — recm — 145x25 — 1364

[RECM#> list backup/wal/obsolete
List backups of cluster 'CLU12' [ Retention mode 'COUNT' value 3 ]

ID                Date                Type    Status    TL      Size  Tag
-----
000160ef361318055678 14-07-2021 21:08:03 WAL     OBSOLETE  29      32 MB 20210714_CLU12_FULL
000160ef37b4280360b0 14-07-2021 21:15:00 WAL     OBSOLETE  29      16 MB WAL_cron
000160f06a8317960660 15-07-2021 19:04:04 WAL     OBSOLETE  29      80 MB 20210715_CLU12_FULL
000160f06b0f2b9885c0 15-07-2021 19:06:23 WAL     OBSOLETE  29      16 MB 20210715_CLU12_WAL
000160f06d141b7e21b8 15-07-2021 19:15:00 WAL     OBSOLETE  29      16 MB WAL_cron
00016106a86508595828 01-08-2021 15:57:59 WAL     OBSOLETE  29      80 MB my_full
00016106ac6437cce1b0 01-08-2021 16:15:01 WAL     OBSOLETE  29      16 MB WAL_cron

[RECM#> ]
```

Search for a backup by it's TAG

```
defaultuser — recm — 145x25 — 1364

[RECM#> list backup/tag="META"
List backups of cluster 'CLU12'

ID                Date                Type    Status    TL      Size  Tag
-----
0001612b7cb5065b1ef8 29-08-2021 14:25:26 META     AVAILABLE  29      26 KB 20210829_CLU12_META

[RECM#> ]
```

LIST CLUSTERS

This command display the registered clusters into the current deposit.

Options and qualifiers

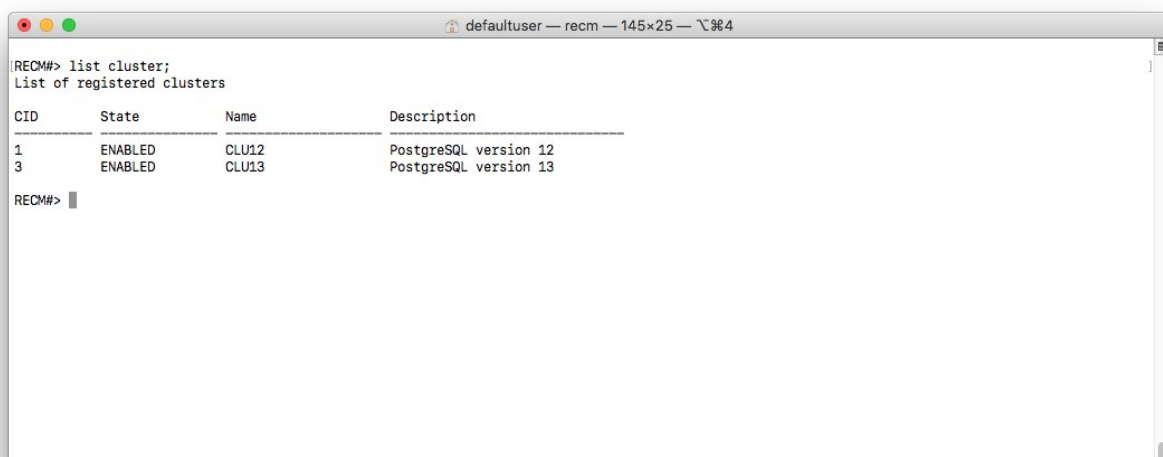
Option/Qualifier	Mandatory	Description
/verbose	No	Display more details

Description

This view can be useful to manage clusters.

See 'modify cluster' for more details about cluster management.

Example:



```
defaultuser — recm — 145x25 — 100%
RECM#> list cluster;
List of registered clusters
CID      State      Name      Description
-----
1        ENABLED    CLU12     PostgreSQL version 12
3        ENABLED    CLU13     PostgreSQL version 13
RECM#>
```

LIST RESTORE POINT

This command display a list of defined restore point.

By default, the restore point of the current connected cluster is listed. By adding a qualifier, you may filter the restore point to display.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/before='<DATE>'	No	(filter)List all restore point prior the date.
/after='<DATE>'	No	(filter)List all restore point after the date.
/source='<STRING>'	No	(filter) list restore point from an other cluster by it's name
/cid=<ID>	No	(filter)List restore point from an other cluster by it's ID.

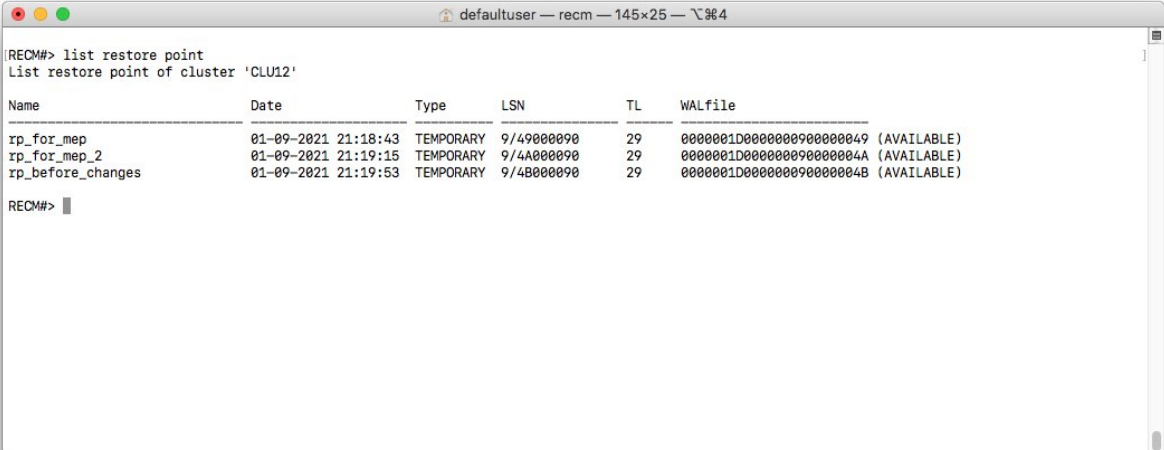
Description

You can use multiple options to filter the list.

The date format is defined by the configuration item '/date_format'. The default is 'YYYY-MM-DD-HH'. (See 'set config' command for more details)

The order is always the creation date.

Example:



```
defaultuser — recm — 145x25 — 100%
[RECM#> list restore point
List restore point of cluster 'CLU12'

Name                Date                Type      LSN          TL      WALfile
-----
rp_for_mep           01-09-2021 21:18:43  TEMPORARY 9/49000090   29      0000001D0000000900000049 (AVAILABLE)
rp_for_mep_2         01-09-2021 21:19:15  TEMPORARY 9/4A000090   29      0000001D000000090000004A (AVAILABLE)
rp_before_changes    01-09-2021 21:19:53  TEMPORARY 9/4B000090   29      0000001D000000090000004B (AVAILABLE)

[RECM#> ]
```

LIST WAL

This command display a list of WAL files. The sequence should never contain gap. This list help you to check you have all the WAL backedup.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/available	No	(Filter) List only available WAL files
/incomplete	No	(Filter) List only incomplete WAL files
/obsolete	No	(Filter) List only obsolete WAL files
/before='<DATE>'	No	(filter)List all restore point prior the date.
/after='<DATE>'	No	(filter)List all restore point after the date.
/source='<STRING>'	No	(filter) list restore point of cluster by it's name
/cid=<ID>	No	(filter)List restore point of cluster by it's ID.

Description

You can use multiple options to filter the list.

The date format is defined by the configuration item '/date_format'. The default is 'YYYY-MM-DD-HH'. (See 'set config' command for more details)

Example:

```

10] - 0001609303043740f948 OBSOLETE 0000001D0000000000000072 2021-05-05 22:41:39
11] - 0001609303043740f948 OBSOLETE 0000001D0000000000000071.00000028.backup 2021-05-05 22:41:39
12] - 000160930450347a4bb0 OBSOLETE 0000001D0000000000000073 2021-05-05 22:45:39
13] - 000160930450347a4bb0 OBSOLETE 0000001D0000000000000074 2021-05-05 22:46:39
14] - 0001609307510292bcb8 OBSOLETE 0000001D0000000000000075 2021-05-05 22:47:12
15] - 0001609307510292bcb8 OBSOLETE 0000001D0000000000000076 2021-05-05 22:48:12
16] - 0001609307510292bcb8 OBSOLETE 0000001D0000000000000077 2021-05-05 22:51:12
17] - 00016094403410d382a8 AVAILABLE 0000001D0000000000000078 2021-05-06 21:05:28
18] - 00016094403410d382a8 AVAILABLE 0000001D0000000000000079 2021-05-06 21:06:29
19] - 00016094403410d382a8 AVAILABLE 0000001D000000000000007A 2021-05-06 21:07:29
20] - 00016094403410d382a8 AVAILABLE 0000001D000000000000007B 2021-05-06 21:11:30
21] - 00016094c7a61d604b50 AVAILABLE 0000001D000000000000007C 2021-05-07 06:52:18
22] - 00016094c7a61d604b50 AVAILABLE 0000001D000000000000007D.00000028.backup 2021-05-07 06:52:53
23] - 00016094c7a61d604b50 AVAILABLE 0000001D000000000000007D 2021-05-07 06:52:53
24] - 00016094c8152ff582f8 AVAILABLE 0000001D000000000000007E 2021-05-07 06:54:13
25] - 00016094c8152ff582f8 AVAILABLE 0000001D000000000000007F 2021-05-07 06:54:44
26] - 00016094c8152ff582f8 AVAILABLE 0000001D000000000000007F.00000028.backup 2021-05-07 06:54:44
27] - 00016094c9501d5a3c88 AVAILABLE 0000001D0000000000000080 2021-05-07 06:59:44
28] - 0001609513241a80b6b8 AVAILABLE 0000001D0000000000000081 2021-05-07 12:10:48
--?] *** GAP *** between '0000001D0000000000000081' and '0000001D0000000000000083'
29] - 0001609510540376c3b8 AVAILABLE 0000001D0000000000000083 2021-05-07 12:49:53
30] - 0001609510540376c3b8 AVAILABLE 0000001D0000000000000084 2021-05-07 12:49:54
31] - 0001609510540376c3b8 AVAILABLE 0000001D0000000000000083.00000028.backup 2021-05-07 12:49:54
32] - 000160951db038725910 AVAILABLE 0000001D0000000000000085 2021-05-07 12:54:54
33] - 0001609903291dff9890 AVAILABLE 0000001D0000000000000086.00000028.backup 2021-05-10 11:55:52
34] - 0001609903291dff9890 AVAILABLE 0000001D0000000000000086 2021-05-10 11:55:52
35] - 0001609907a5091ff3e8 AVAILABLE 0000001D0000000000000087 2021-05-10 12:00:53
36] - 00016099123023914538 AVAILABLE 0000001D0000000000000088 2021-05-10 12:47:59
37] - 00016099123023914538 AVAILABLE 0000001D0000000000000089 2021-05-10 12:51:59
38] - 0001609aebfd36263a78 AVAILABLE 0000001D000000000000008A 2021-05-11 22:41:24
39] - 0001609aeccc181f3778 AVAILABLE 0000001D000000000000008B 2021-05-11 22:42:24
RECM#> exit

```

MODIFY CLUSTER

This command allow you to modify a cluster entry in the DEPOSIT.

By default the current connected cluster is changed.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/backupdir="<PATH>"	No	Modify backup location
/blksize=<SIZE>	No	Modify BlockSize
/compression=<-1..9>	No	Modify compression level (-1 to disable it)
/concurrently=yes no	No	Change the way to rebuild indexes
/db=<NAME>	No	Change default database name used at connection time (Default is 'postgres')
/date_format=<fmt>	No	Change date format
/delwal=<MIN>	No	In minutes, specify how long a WAL file stay on disk before deletion
/description='<STRING>'	No	Change the cluster description
/enable	No	Enable the cluster activity
/disable	No	Suspend the cluster activity (All backups will be skipped)
/full="<RETENTION>"	No	Modify retention setting for FULL backups
/ip=<IP>	No	Set IP address to use
/maxfiles=<COUNT>	No	Set Maximum number of files in a backup set
/maxsize=<size>	No	Set Maximum size of a backup set
/meta="<RETENTION>"	No	Modify retention setting for META backups
/name="<STRING>"	No	Modify the cluster, by identifying it by it's name
/pwd="<STRING>"	No	Change password
/port=<port>	No	Change port
/repopts="<STRING>"	No	Change options for replication (standby)
/reppwd="<STRING>"	No	Change password of replication user(standby)
/repusr="<STRING>"	No	Change user of replication (standby)
/waldir="<PATH>"	No	Change WAL backup directory (same as 'archive_command' setting)
/cid=<ID>	No	Modify the cluster, by identifying it by it's ID

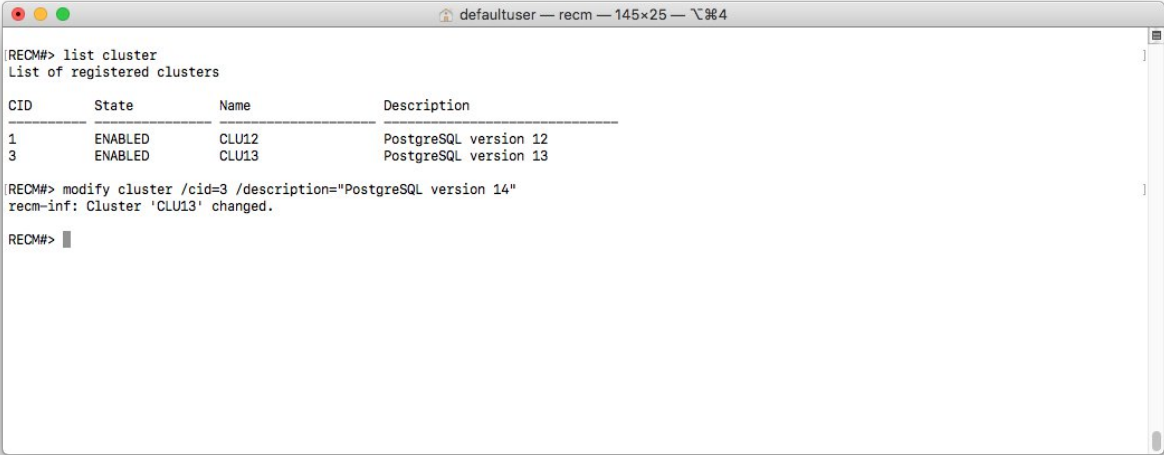
Description

This command allow you to :

- Suspend all backups for a specified cluster.
- Modify the description

- Modify some parameters

Example:



```
defaultuser — recm — 145x25 — 100%
```

```
[RECM#> list cluster  
List of registered clusters
```

CID	State	Name	Description
1	ENABLED	CLU12	PostgreSQL version 12
3	ENABLED	CLU13	PostgreSQL version 13

```
[RECM#> modify cluster /cid=3 /description="PostgreSQL version 14"  
recm-inf: Cluster 'CLU13' changed.  
[RECM#> ]
```

MODIFY BACKUP

This command allow you to change some properties of an existing backup.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/available	No	Change status of a backup to 'Available'
/incomplete	No	Change status of a backup to 'incomplete'
/obsolete	No	Change status of a backup to 'obsolete'
/cid=<CID>	No	Allow you to modify a backup of an other server
/uid=<STRING>	Yes	UID of the backup.
/tag=<STRING>	no	Modify backup TAG

Description

This command allow you to change the Tag, and the state of a backup.

You can also change backup properties of an other cluster, by using the cluster ID (CID)

Example:

```
defaultuser — recm — 145x25 — ㄟ%4
[RECM#> list backup/cid=3
List backups of cluster 'CLU13'

ID                Date                Type  Status    TL      Size  Tag
-----
000360ad4a7a26e1196  25-05-2021 22:52:06  FULL  AVAILABLE  1       54 MB  20210525_CLU13_FULL
000360ad4a7e037413bc  25-05-2021 22:52:10  WAL   AVAILABLE  1       32 MB  20210525_CLU13_FULL
000360ad682d354ce045  26-05-2021 12:58:11  FULL  AVAILABLE  1       54 MB  to_nass
000360ad682f089aa80f  26-05-2021 12:58:13  WAL   AVAILABLE  1       32 MB  NAS_ROOM_A

[RECM#> modify backup/cid=3/tag="NAS_ROOM_B" /id=000360ad682f089aa80f
recm-inf: Backup '000360ad682f089aa80f' changed.

[RECM#>
```

QUIT

This command terminate the program. This is a synonym of 'exit'.

Options and qualifiers

none

REGISTER CLUSTER

This command register the cluster in the current DEPOSIT.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/force	No	Register again the cluster.
/cid=<CID>	No	Force a registration, using a specific cluster ID.

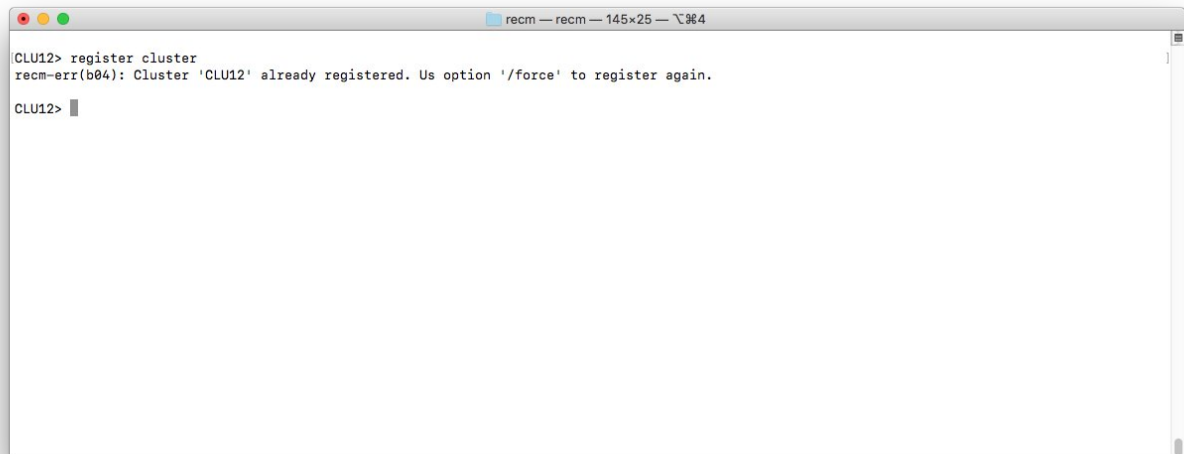
Description

This command register the current connected cluster to the currently connected DEPOSIT. The '/force' option perform the 'registration' without loosing the content of the DEPOSIT (See 'register files' for recovering backup files).

If you want to force a cleanup, you have to use 'delete cluster' command, and register it again.

Once the register id done, the connection to the DEPOSIT is automatically associated to the cluster connection.

Example:



```
recm — recm — 145x25 — 1004
CLU12> register cluster
recm-err(b04): Cluster 'CLU12' already registered. Us option '/force' to register again.
CLU12> 
```

REGISTER FILES

This command register backup files from a file system in the current DEPOSIT.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/cfg	No	Register only configuration backups files
/wal	No	Register only WAL backup files
/full	No	Register only FULL and WAL backups files
/meta	No	Register only METADATA backup files
/directory=<path>	No	Register backup files from a given directory

Description

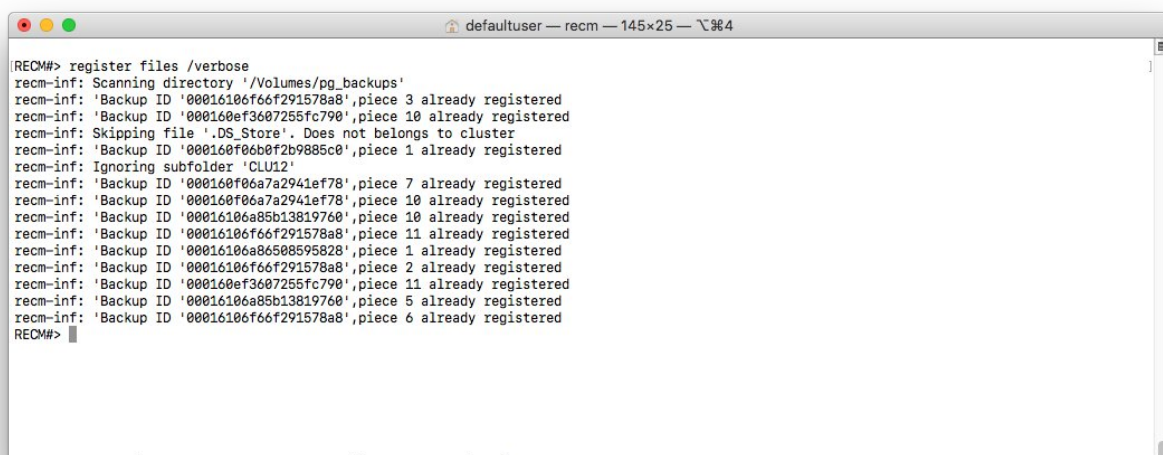
This command register backup files from a file system in the current DEPOSIT. This command can be used when you create a new deposit, or when you need to move backup files to different location.

The process does not take subdirectories into account.

The default directory used is the '/backpudir' parameter defined in cluster (show cluster/backupdir)

Registering files does not retrieve 'indexes' informations. That means, backup done with '/noindex' will loose index informations. The backup is still usable, but you will have to recreate all the missing indexes by yourself.

Example:



```
RECM#> register files /verbose
recm-inf: Scanning directory '/Volumes/pg_backups'
recm-inf: 'Backup ID '00016106f66f291578a8',piece 3 already registered
recm-inf: 'Backup ID '000160ef3607255fc790',piece 10 already registered
recm-inf: Skipping file '.DS_Store'. Does not belongs to cluster
recm-inf: 'Backup ID '000160f06b0f2b9885c0',piece 1 already registered
recm-inf: Ignoring subfolder 'CLU12'
recm-inf: 'Backup ID '000160f06a7a2941ef78',piece 7 already registered
recm-inf: 'Backup ID '000160f06a7a2941ef78',piece 10 already registered
recm-inf: 'Backup ID '00016106a85b13819760',piece 10 already registered
recm-inf: 'Backup ID '00016106f66f291578a8',piece 11 already registered
recm-inf: 'Backup ID '00016106a86508595828',piece 1 already registered
recm-inf: 'Backup ID '00016106f66f291578a8',piece 2 already registered
recm-inf: 'Backup ID '000160ef3607255fc790',piece 11 already registered
recm-inf: 'Backup ID '00016106a85b13819760',piece 5 already registered
recm-inf: 'Backup ID '00016106f66f291578a8',piece 6 already registered
RECM#>
```

RESTORE CONFIG

This command restore a configuration file.

Options and qualifiers

Option/Qualifier	Mandatory	Description
/verbose	No	Display more details about the operation
/overwrite	No	When restoring, overwrite exiting file
/file=<String>	No	Using syntax like 'regexpr', search for file to restore
/uid=<string>	No	Backup UID to use for restore
/before=<date>	No	Filter backups by the older files than the date specified
/after=<date>	No	Filter backups by the newer files than the date specified
/directory=<path>	No	Restore files on a different location than original one.
/tag=<string>	No	Filter backups on a keyword from the tag

Description

This command restore one or more configuration files from a backup.

The process can filter on

- The TAG
- The date
- The file name

By defaultn the files restored are save on their original places. The '/overwrite' option allow you to overwrite the restored files.

Example:

```
000161d17da9180b70f8 2022-01-02 11:25:45 CFG AVAILABLE 35 7 KB 20220102_CLU12_CFG

CLU12> show backup/uid=000161d17da9180b70f8/content
Backup UID '000161d17da9180b70f8' Cluster 'CLU12' (CID=1)
Backup type...: CFG Version.....: 12 Status:AVAILABLE
Size....: 8159 Bytes (0 MB) in 1 piece(s)
Started:2022-01-02 11:25:45 Elapsed Time:0s
Ended...:0000-00-00 00:00:00
Options: compressed(C)

  id      Files      Size Piece
-----
1]      4          7 KB (3 KB) /Volumes/pg_backups/000161d17da9180b70f8_1_CFG.recn [51.27 %]
Content:
  mdate      Size File name
-----
1] 2021-12-29 17:48:22      977 ./postgresql.auto.conf
2] 2021-12-29 17:48:22     4921 ./pg_hba.conf
3] 2021-12-29 17:48:22     1636 ./pg_ident.conf
4] 2021-12-29 17:48:22     625 ./postgresql.conf
-----
4          7 KB (3 KB) [51.27%]
CLU12> restore config/uid=000161d17da9180b70f8/file=pg_hba /dir="/tmp/retore"
```

Full example for common usage

Create a deposit and register our first PostgreSQL engine

In this example, we assume that you already have a PostgreSQL engine to use as DEPOSIT.

1 - Connect to the PostgreSQL engine

```
#> recm
RECM#> connect cluster /host=localhost /port=5432 /usr=postgres /pwd=postgres /db=postgres
recm-inf: Connected to cluster 'CLU12'
RECM#> create deposit /usr=depoadm /pwd=depoadm/db=recmdb
.....
recm-inf: Deposit 'CLU12' created.
RECM#>
RECM#> create deposit /user=depoadm /password=depoadm/database=recmdb /verbose
recm-err(704): Deposit 'recmdb' already exist. Use '/force' to recreate it.
RECM#>
```

The deposit name correspond to the configuration parameter 'cluster_name'. In this example, the deposit database is 'recmdb', and the object owner is 'depoadm'.

2 – Register the first PostgreSQL engine to the deposit.

```
#> recm
RECM#> connect cluster /host=localhost /port=5432 /usr=postgres /pwd=postgres /db=postgres
recm-inf: Connected to cluster 'CLU12'
RECM#> create deposit /usr=depoadm /pwd=depoadm/db=recmdb
.....
recm-inf: Deposit 'CLU12' created.
RECM#>
RECM#> create deposit /user=depoadm /password=depoadm/database=recmdb /verbose
recm-err(704): Deposit 'recmdb' already exist. Use '/force' to recreate it.
RECM#>
```

Create our first PostgreSQL backup

We can see, in this example, that we have missing WAL files (GAP)

```
#> recm -t CLU12
recm-inf: Connected to cluster 'CLU12'
recm-inf: Registered to 'REPO12' (1)
RECM#> backup full/wal/noindex/verify/cleanup/tag="My_First_Backup"
recm-inf: Backup UID : 000161d17fd02bdc2640
recm-inf: Total pieces ..... : 13
recm-inf: Total files backedup.. : 2574
recm-inf: Total Backup size .... : 2742350498 (2 GB)
recm-inf: Total bytes skipped... : 5528560 (5 MB)
recm-inf: Total files skipped... : 42
NOTICE: all required WAL segments have been archived
recm-inf: Backup UID : 000161d17ff9259dfd58
recm-inf: Total pieces ..... : 1
recm-inf: Total files backedup.. : 4
recm-inf: Total files deleted... : 0      (Files aged 10 minute(s) or more)
recm-inf: Total Backup size .... : 50331988 (48 MB)
recm-inf: Verifying backups of cluster 'CLU12'
recm-inf: Verifying WAL sequence
recm-wrn(d204): GAP between '00000023000000140000003D' and '000000230000001400000042'.
(AVAILABLE backup). Verify and Launch a FULL backup if necessary.
recm-inf: Backup piece(s) state changed : 9 AVAILABLE, 3 OBSOLETE, 1 INCOMPLETE.
```