**PostgreSQL**

**version 13**

**-**

**Backup, Recovery**

**e Alta Disponibilità**

Distribuito tramite il sito www.manualioracle.it

Copyright © 2023 Assi Loris

Qualsiasi abuso sarà perseguito e punito secondo i termini di legge.

Version: 3.2

Alcuni termini usati sono trademarks registrati dei rispettivi proprietari.

**Sommario**

Introduzione al Manuale 4

Cap. 1 - Introduzione al Backup e Recovery 5

1.1 – Overview 5

1.2 – Metodi per eseguire il Backup e il Restore 5

Cap. 2 - Backup e Recovery logico 7

2.1 – Introduzione al Dump logico 7

2.2 – *pg\_dump, pg\_dumpall* e *pg\_restore* 7

2.3 – Esempi di Dump e Restore di Database 9

2.4 – Esempi di Dump e Restore di singole Tabelle 10

2.5 – Dump di grandi database 11

2.6 – Upgrade di un Cluster Postgres 12

2.7 – Migrazione di un singolo database 13

2.8 – WAL creation e Confronto fra pg\_dump/pg\_restore e CREATE DATABASE ... TEMPLATE 14

2.9 – Tempistiche e Spazi su Disco 15

Cap. 3 - Backup e Recovery fisico 17

3.1 – File System Level Backup 17

3.2 – Configurazione senza Continuous Archiving 18

3.3 – Configurazione con Continuous Archiving 19

3.4 – Settare il WAL archiving 20

3.5 – Numero di WAL file nella directory pg\_wal 22

3.6 – Esempio di Continuous archiving 23

3.7 – Fare un Base Backup 25

3.8 – Esempi di Backup configuration 27

Cap. 4 - Barman 29

4.1 – Backup fisico di Postgres tramite tool “esterni” 29

4.2 – Il tool Barman 29

4.3 – Barman: Design e Architettura 30

4.4 – Barman: Scenari Tipici per il Backup 31

4.5 – Barman: Installazione e Configurazione 33

4.6 – Comandi Barman 34

Cap. 5 - pgBackRest 36

5.1 – Introduzione al tool pgBackRest 36

5.2 – Principali feature di pgBackRest 37

5.3 – Installazione di pgBackRest su macchina redHat 38

5.4 – Installazione di pgBackRest su macchina Debian 40

5.5 – pgbackrest.conf e Stanza 44

5.6 – Primi comandi di pgBackRest 46

5.7 – Tipi di backup pgbackrest 47

5.8 – Come interpretare le info relative ai backup di una stanza 48

5.9 – Esempio di configurazione pgbackrest (singolo server) con lancio backup 49

5.10 – Esempio di configurazione pgbackrest (2 server) con lancio e schedulazione backup 50

Cap. 6 - pgBackRest: Restore e Rimozioni 53

6.1 – Restore in un nuovo database server 53

6.2 – Restore issues e Link utili 55

6.3 – Restore ad uno specifico orario (Point In Time Recovery, PITR) 56

6.4 – Restore nello stesso database server 57

6.5 – Rimozione di una stanza esistente 59

6.6 – Disinstallazione di pgBackRest 60

Cap. 7 - Standby database 61

7.1 – Alta Disponibilità e Standby database 61

7.2 – Standby db: Alta Disponibilità e Load Balancing 64

7.3 – Standby db: Configurazioni esistenti 65

7.4 – Hot Standby 66

7.5 – Failover e Switchover 67

7.6 – Esempio di Switchover 68

7.7 – Replication slot 71

7.8 – Cascade Replication 71

Cap. 8 - Architettura HA con Streaming Replication 73

8.1 – Come configurare la Streaming Replication 73

8.2 – Streaming Replication: controlli tramite processi e query 76

8.3 – Streaming Replication: controlli con pg\_stat\_replication 79

8.4 – Streaming Replication + Backup Server con pgBackRest 80

8.5 – Streaming Replication: pgBackRest Backup details 85

8.6 – pgBackRest Backup di un singolo database cluster in Streaming Replication 87

8.7 – Standby db: i WAL nella Streaming replication 90

8.8 – Setup del passwordless SSH 93

8.9 – Patroni 96

8.10 – Patronictl 97

Cap. 9 - Logical Replication e Miscellaneous 99

9.1 – Introduzione alla Logical Replication 99

9.2 – La Publication 100

9.3 – La Subscription 101

9.4 – Esempio di configurazione della Logical Replication 102

Cap. 10 - Miscellaneous 104

10.1 – Introduzione ed installazione di check\_pgbackrest 104

10.2 – Comandi check\_pgbackrest 104

10.3 – Viste di Catalogo Postgres 106

## Introduzione al Manuale

 **Contenuto**

Argomenti trattati in questo manuale:

- backup e recovery logico (*pg\_dump, pg\_restore*)

- backup e recovery fisico, continuous archiving, ecc

- tool barman e pgbackrest

- standby database, hot standby, streaming replication, ecc

- passwordless SSH

- logical replication

Ci riferiamo a Postgres versione 13 su Linux (soprattutto Debian e Red Hat) ma quasi tutti gli argomenti e comandi qui scritti sono validi anche per le altre Linux distribution.

 **Audience**

Il presente manuale è rivolto ai tecnici informatici, agli studenti e a chiunque voglia capire che cosa sia un database e voglia conoscere le principali caratteristiche di Postgres.

 **Particolarità**

Il presente manuale è in italiano ma molti termini tecnici in esso contenuti sono in lingua inglese.

Abbiamo fatto tale scelta perché spesso a lavoro tali termini vengono detti in inglese e perché riteniamo più utile che i lettori del manuale conoscano tali termini nella lingua usata nella documentazione ufficiale.

Molti argomenti sono schematizzati al fine di fornire una comprensione ed una memorizzazione superiore.

Alcune abbreviazioni usate:

 !!! : Attenzione

db : database

 HA: High Availability: alta disponibilità

 restorare : fare il restore

 O.S. : *Operating System* (Sistema Operativo)

 *<data\_directory>* : valore del parametro Postgres chiamato *data\_directory*

 **Principali Versioni**

- 10.05.2023 : version 1.0

- 14.05.2023 : version 1.1 rilascio al pubblico

- 10.11.2023 : version 2.0 rimosso i capitoli relativi al *Tuning* e approfondito gli altri argomenti

- 03.05.2024 : version 2.6 aggiunto vari approfondimenti (soprattutto riguardanti il *restore*)

**Disclaimer**

Non si fornisce alcuna garanzia relativamente al fatto che il presente documento sia privo di errori. Non si assume nessuna responsabilità sugli eventuali errori o danni derivanti dall’uso delle informazioni qui contenute.

## Introduzione al Backup e Recovery

1.1 – Overview

Il backup e recovery è uno degli aspetti più importanti dell’amministrazione del database.

Se un db crasha e non c’è modo di ripristinarlo, il risultato potrebbe essere devastante per il business e potrebbe causare:

* perdita di dati
* insoddisfazione da parte dei clienti
* perdita del fatturato aziendale

Il backup e recovery si riferisce alle strategie e procedure utili per:

* la protezione del db contro la perdita di dati
* la ricostruzione dei dati nel caso di perdita dati

**Backups**

Un backup è una copia dei dati. Grazie al backup puoi:

* ricostruire i dati persi
* creare un *db clone* del db originale

I tipi di backup possono essere classificati in due categorie principali:

* **Backup Fisico**:

Consiste nella copia dei *data file* del database, inclusi i file delle tabelle, dei log e delle configurazioni. Questo tipo di backup è generalmente più veloce e si utilizza per ripristinare l'intero database in caso di disastri o guasti hardware.

È la scelta ideale per il ripristino rapido, soprattutto per database di grandi dimensioni.

* **Backup Logico**:

..................

..................

..................

## Backup e Recovery logico

### 2.1 – Introduzione al Dump logico

Il Dump logico è un backup che permette di generare degli script che possono essere rilanciati per creare un database come era quello *source* al momento del *dump*.

Per fare questo, Postgres usa l'utility chiamata *pg\_dump*.

Come sintetizzato nella figura sottostante, esistono 2 tipi di *dump* e *restore*:

* Se lanci il comando *pg\_dump*, di default verrà creatoun *SQL script* che può essere caricato nel *db target* tramite *psql.*

Tale *SQL script* viene anche chiamato *SQL Dump* (normalmente viene letto “*Siquel Damp”*).

* Se lanci il comando *pg\_dump* con l’opzione *-Fc* produrrai un *dump binario* (chiamato anche *custom dump format script)* che può essere caricato nel *db target* solo tramite l’utility *pg\_restore.*

Normalmente tale *custom dump file* avrà dimensioni inferiori rispetto al *SQL script*, soprattutto se stai backuppando tanti dati.



Questo comando normalmente viene lanciato dall'O.S. user *postgres* e questa è la sua sintassi base:

..................

..................

..................

## Backup e Recovery fisico

Come qualsiasi programma che contiene dati delicati, Postgres andrebbe backuppato ad intervalli regolari (ad esempio ogni notte).

Un backup regolare garantisce che i dati possano essere ripristinati rapidamente in caso di guasto o perdita accidentale.

I principali metodi di backup fisico di Postgres sono:

*- File system level backup*

*- Base backup + Continuous archiving*

*-* Tool specializzati come*Barman* o *pgBackRest*

I primi 2 metodi li descriveremo in questo capitolo, gli altri li vedremo nei capitoli successivi.

### 3.1 – File System Level Backup

Un modo per fare il backup fisico del *cluster database* è quello di copiare direttamente i file che Postgres usa per contenere i file del db.

Tale metodo è consigliato solo per database non di produzione. Per database di produzione il metodo che suggeriamo è tramite tool specializzati come *Barman* e *pgBackRest*.

Tali file sono contenuti nella directory indicata dal parametro *data\_directory*: da *psql* lancia*:*

 show data\_directory;

 data\_directory

 ---------------------------

 /var/lib/postgresql/13/main

Tale parametro è configurato dentro il file dei parametri *postgresql.conf.*

..................

..................

..................

## Barman

### 4.1 – Backup fisico di Postgres tramite tool “esterni”

Come visto nei capitoli precedenti esistono 2 tipi di backup:

* Backup Logico

Il *pg\_dump* permette un backup logico del database e dei suoi oggetti e permette un semplice trasferimento dei dati da un database ad un altro, anche fra 2 diverse release Postgres.

* Backup Fisico

Permette di fare un *point in time recovery.* Ossia permette di ricostruire un database ad un momento diverso da quello in cui è stato lanciato il *base backup*.

I backup fisici utilizzano anche le informazioni contenute nei *WAL file*.

Questo tipo di backup tipicamente lo fai tramite il comando *pg\_basebackup* o tramite tool esterni che ti permettono facilmente di gestire il backup e recovery di Postgres.

Alcuni dei tool di terze parti più usati per il backup e recovery fisico di un database Postgres sono:

* *Barman*

Prima versione “stabile” è uscita nel Luglio 2012.

* *pgBackRest*

Prima versione “stabile” è uscita nell’Aprile 2016.

Praticamente entrambi sono il corrispettivo Postgres del tool RMAN della Oracle.

..................

..................

..................