

Postgres Pro Enterprise 13

Резервное копирование — 2



Авторские права

© Postgres Professional, 2023 год.

Авторы: Алексей Береснев, Илья Баштанов, Павел Толмачев

Использование материалов курса

Некоммерческое использование материалов курса (презентации, демонстрации) разрешается без ограничений. Коммерческое использование возможно только с письменного разрешения компании Postgres Professional. Запрещается внесение изменений в материалы курса.

Обратная связь

Отзывы, замечания и предложения направляйте по адресу:

edu@postgrespro.ru

Отказ от ответственности

Компания Postgres Professional не несет никакой ответственности за любые повреждения и убытки, включая потерю дохода, нанесенные прямым или косвенным, специальным или случайным использованием материалов курса. Компания Postgres Professional не предоставляет каких-либо гарантий на материалы курса. Материалы курса предоставляются на основе принципа «как есть» и компания Postgres Professional не обязана предоставлять сопровождение, поддержку, обновления, расширения и изменения.

Удаленная работа

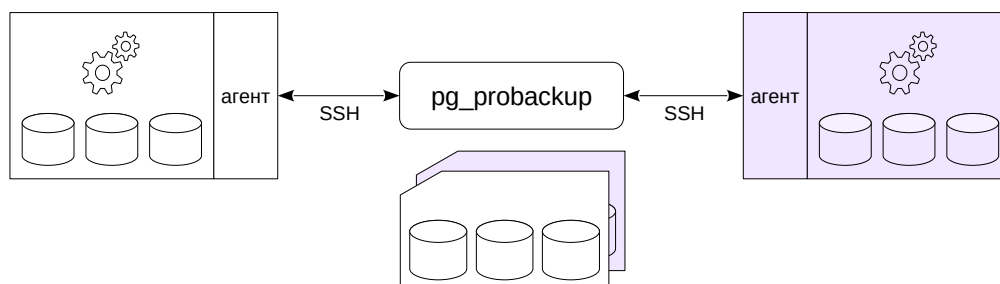
Архивирование WAL

Страничное копирование и восстановление

Частичное восстановление

Восстановление страниц

Восстановление на момент времени



Копируемые экземпляры и каталог копий располагаются на разных серверах

В удаленных системах запускаются агенты

Агенты читают файлы и передают по протоколу SSH

3

Утилита `pg_probackup` обычно используется в удаленном режиме, основанном на протоколе SSH (Secure Shell). Копируемые экземпляры могут находиться на разных серверах. Операции копирования, восстановления и архивирования WAL выполняются удаленно по сети.

На сервере с каталогом копий и на серверах с копируемыми экземплярами должен быть установлен `pg_probackup` одной и той же версии.

В удаленном режиме `pg_probackup` на сервере с каталогом копий подключается по SSH к утилите на сервере Postgres, последняя запускает один или несколько процессов-агентов, которые взаимодействуют с экземпляром Postgres и обмениваются данными с основной утилитой. Количество агентов определяется ключом `-j`.

Для пользователя ОС, выполняющего резервное копирование, нужно настроить доступ по ssh к целевому серверу с аутентификацией без пароля (по ключу).

Параметры SSH-соединения команды `backup`:

- `--remote-host` — адрес целевого сервера Postgres;
- `--remote-user` — владелец экземпляра Postgres;
- `--remote-port` — порт на целевом сервере.

<https://postgrespro.ru/docs/enterprise/13/app-pgprobackup#PBK-CONFIGURING-THE-REMOTE-MODE>

Удаленная работа

Определим роль backup для резервного копирования и восстановления.

```
student$ psql
```

```
=> CREATE ROLE backup LOGIN REPLICATION PASSWORD 'b@ckUp';
```

CREATE ROLE

Создадим файл `.pgpass`, чтобы не вводить пароль вручную.

```
student$ cat >~/pgpass <<EOF
localhost:5432:backup:backup:b@ckUp
localhost:5432:replication:backup:b@ckUp
EOF
```

```
student$ chmod 600 ~/.pgpass
```

Создадим базу данных, к которой будет подключаться роль bascur.

```
=> CREATE DATABASE backup OWNER backup;
```

CREATE DATABASE

Привилегии для роли bascur.

=> \c backup

You are now connected to database "backup" as user "student".

```
> BEGIN;
GRANT USAGE ON SCHEMA pg_catalog TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.current_setting(text) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.set_config(text, text, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_is_in_recovery() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_start_backup(text, boolean, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_stop_backup(boolean, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_create_restore_point(text) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_switch_wal() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_last_wal_replay_lsn() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_current() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_current_snapshot() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_snapshot_xmax(txid_snapshot) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_control_checkpoint() TO backup;
COMMIT;
```

[illegible]

Мы будем запускать утилиту локально от имени пользователя ОС student, а агент на удаленном сервере будет обращаться к файлам кластера от имени postgres. Поэтому пользователи student и postgres должны обмениваться публичными ключами, сгенерированными без парольной фразы. Это уже сделано в виртуальной машине курса.

Подготовим локальный каталог копий, владелец файлов — student.

```
student$ sudo mkdir /var/probackup
```

```
student$ sudo chown student: /var/probackup
```

```
student$ pg_probackup init -B /var/probackup
```

```
INFO: Backup catalog '/var/probackup' successfully initialized
```

Добавим в локальный каталог копий экземпляр БД, работающий на удаленном сервере.

```
student$ pg_probackup add-instance -B /var/probackup -D /var/lib/pgpro/ent-13 --instance ent-13 --remote-host=localhost --remote-user=postgres
```

INFO: Instance 'ent-13' successfully initialized

Чтобы сократить командную строку, сохраним параметры удаленного доступа в конфигурации.

```
student$ pg_probackup set-config -B /var/probackup --instance ent-13 -d backup -U backup --remote-host=localhost --remote-user=postgres
```

Вот что записалось в конфигурацию:

```
students$ pg_probackup show-config -B /var/probackup --instance ent-13
```

```

pgBackup instance information
pgdata = /var/lib/pgpro/ent-13
system-identifier = 7323121610635147817
xlog-seg-size = 16777216
# Connection parameters
pgdatabase = backup
pghost = localhost
pguser = backup
# Archive parameters
archive-timeout = 5min
# Logging parameters
log-level-console = INFO
log-level-file = OFF
log-format-console = PLAIN
log-format-file = PLAIN
log-filename = pg_probackup.log
log-rotation-size = 8TB
log-rotation-age = 8d
# Retention parameters
retention-redundancy = 0
retention-window = 0
wal-depth = 0
# Compression parameters
compress-algorithm = none
compress-level = 1
# Remote access parameters
remote-proto = ssh
remote-host = localhost
remote-user = postgres

```

Выполним полное резервное копирование с удаленного сервера, используя потоковую доставку записей WAL:

```
students$ pg_probackup backup -B /var/probackup --instance ent-13 -b FULL --stream
```

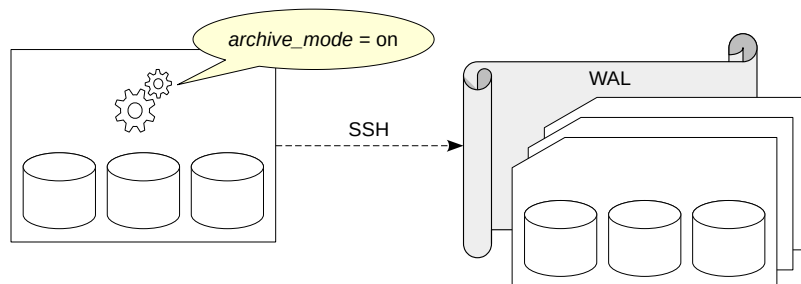
```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAG6, backup mode: FULL, wal mode: STREAM, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
```

INFO: wait for pg_start_backup()
INFO: PGDATA size: 334MB
INFO: Current Start LSN: 0/10000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 1s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: stop_stream_lsn 0/11000000 currentpos 0/11000000
INFO: backup->stop_lsn 0/1000C168
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 4s
INFO: Validating backup S7DAG6
INFO: Backup S7DAG6 data files are valid
INFO: Backup S7DAG6 resident size: 350MB
INFO: Backup S7DAG6 completed

В каталоге резервных копий появилась запись:

student\$ pg_probackup show -B /var/probackup --instance ent-13

Instance	Version	ID	Recovery Time		Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAG6	2024-01-16	21:38:32.830407+03	FULL	STREAM	1/0	6s	334MB	16MB	1,00	0/10000028	0/1000C168	OK



Архив WAL сохраняется в каталог копий

`archive_command = 'pg_probackup archive-push ...'`

- `--remote-host` — адрес каталога копий
- `--remote-user` — владелец каталога резервных копий
- `--remote-port` — порт SSH на сервере резервного копирования

5

Непрерывная (файловая) архивация сегментов WAL применяется чаще всего, так как она требуется для инкрементального копирования в режиме PAGE и обеспечивает возможность восстановления на момент в прошлом — PITR (Point In Time Recovery). При создании резервной копии `pg_probackup` по умолчанию предполагает, что на копируемом экземпляре настроена архивация WAL. Утилита `pg_probackup` может использоваться для архивации, для этого параметр `archive_command` должен содержать команду `pg_probackup archive-push`.

При работе через SSH нужно задать параметры удаленного доступа.

Утилиту можно использовать и для архивации при локальной работе, при этом она будет запускаться от имени владельца кластера данных, обычно `postgres`. В таком случае каталог копий также должен принадлежать `postgres` (это особенность реализации текущей версии `pg_probackup`).

Подробности настройки описаны в документации:

<https://postgrespro.ru/docs/enterprise/13/app-pgprobackup#PBK-SETTING-UP-CONTINUOUS-WAL-ARCHIVING>

Архивация WAL

Подготовим экземпляр к работе с архивацией WAL.

```
=> ALTER SYSTEM SET archive_mode = on;

ALTER SYSTEM

=> ALTER SYSTEM SET archive_command = 'pg_probackup archive-push -B /var/probackup --instance=ent-13 --wal-file-path=%p --wal-file-name=%f --remote-host=localhost --remote-user=student

ALTER SYSTEM
```

Экземпляр необходимо перезагрузить.

```
student$ sudo systemctl restart postgrespro-ent-13.service

student$ psql
```

Убедимся, что файловая архивация работает и сегменты WAL попадают в каталог копий.

```
student$ tree --noreport /var/probackup/wal

/var/probackup/wal
└─ ent-13

=> SELECT pg_switch_wal();

pg_switch_wal
-----
0/110000E8
(1 row)
```

```
=> CHECKPOINT;
```

CHECKPOINT

```
student$ tree --noreport /var/probackup/wal

/var/probackup/wal
└─ ent-13
   └─ 000000010000000000000000000011
```

Информацию об архиве WAL получим средствами pg_probackup:

```
student$ pg_probackup show -B /var/probackup --archive

INFO: checking WAL file name "000000010000000000000000000011"
```

ARCHIVE INSTANCE 'ent-13'										
TLI	Parent TLI	Switchpoint	Min Segno	Max Segno	N segments	Size	Zratio	N backups	Status	
1	0	0/0	00000001000000000000000011	00000001000000000000000011	1	16MB	1,00	1	OK	

Выполним полное резервное копирование. Поскольку настроена файловая архивация, копировать WAL не нужно, ключ --stream не указываем.

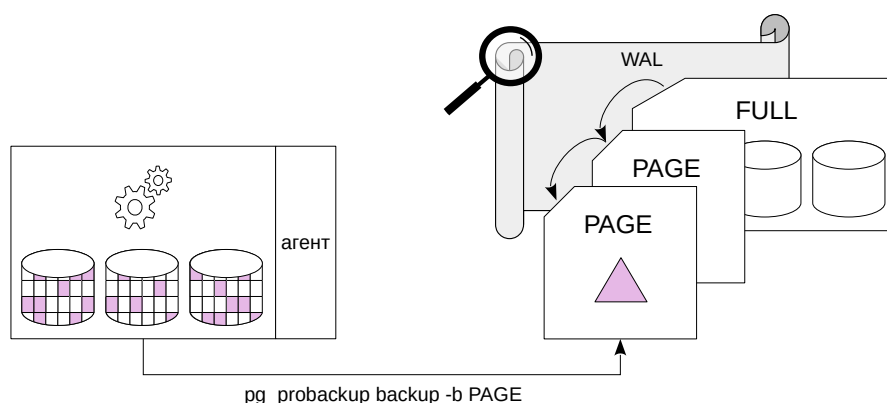
```
student$ pg_probackup backup -B /var/probackup --instance ent-13 -b FULL

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAGE, backup mode: FULL, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Wait for WAL segment /var/probackup/wal/ent-13/00000001000000000000000013 to be archived
INFO: PGDATA size: 334MB
INFO: Current Start LSN: 0/13000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 2s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/140000D0 in archived WAL segment /var/probackup/wal/ent-13/00000001000000000000000014
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 4s
INFO: Validating backup S7DAGE
INFO: Backup S7DAGE data files are valid
INFO: Backup S7DAGE resident size: 334MB
INFO: Backup S7DAGE completed
```

Проверим каталог резервных копий:

```
student$ pg_probackup show -B /var/probackup --instance ent-13
```

Instance	Version	ID	Recovery Time		Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAGE	2024-01-16	21:38:40.396964+03	FULL	ARCHIVE	1/0	6s	334MB	16MB	1,00	0/13000028	0/140000D0	OK
ent-13	13	S7DAGE6	2024-01-16	21:38:32.830407+03	FULL	STREAM	1/0	6s	334MB	16MB	1,00	0/10000028	0/1000C168	OK



Информация об измененных страницах — из архива WAL

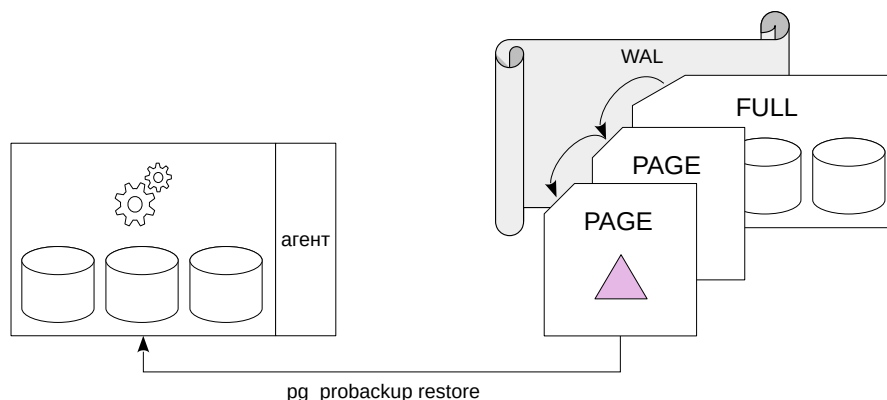
В архиве должны быть все сегменты WAL, записанные после предыдущей копии

7

При наличии архива WAL для создания инкрементальных копий можно использовать еще один режим инкрементального резервного копирования `pg_probackup` — страничное копирование (PAGE).

При копировании PAGE `pg_probackup` сканирует все записи WAL в архиве, начиная с момента создания предыдущей полной или инкрементальной копии, и определяет по ним список изменившихся страниц. При этом необходимо, чтобы в архиве присутствовали все файлы WAL, сгенерированные после предыдущей копии.

Режим PAGE особенно эффективен, когда в базах данных происходит немного изменений. Если же размер журнальных файлов сравним с общим размером файлов базы данных, ускорение будет незначительным, хотя размер копии все равно будет меньше.



--remote-host — сервер с копируемым экземпляром
--remote-user — владелец экземпляра (postgres)
--remote-port — порт SSH на сервере с копируемым экземпляром
--archive-host — сервер резервного копирования
--archive-user — владелец каталога резервных копий

8

Для восстановления при удаленной работе утилита `pg_probackup` запускается на сервере резервного копирования и взаимодействует с агентом через SSH.

При удаленном восстановлении команда `pg_probackup restore` использует ключи:

- --remote-host — сервер с копируемым экземпляром;
- --remote-user — владелец экземпляра (postgres);
- --remote-port — порт SSH на сервере с копируемым экземпляром;
- --archive-host — сервер резервного копирования;
- --archive-user — владелец каталога резервных копий.

Обратите внимание, что ключ `--remote_port` задает порт для SSH-соединения (по умолчанию 22), а не порт для соединения с экземпляром Postgres (по умолчанию 5432).

Режим PAGE

Архивацию WAL мы уже настроили. Создадим таблицу и добавим в нее строку.

```
=> CREATE TABLE IF NOT EXISTS t1( msg text );

CREATE TABLE

=> INSERT INTO t1 VALUES ('Проверим режим PAGE.');
```

Выполним инкрементальное резервное копирование в режиме PAGE.

```
student$ pg_probackup backup -B /var/probackup --instance ent-13 -b PAGE

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAGK, backup mode: PAGE, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAGE
INFO: Wait for WAL segment /var/probackup/wal/ent-13/00000001000000000000000016 to be archived
INFO: PGDATA size: 334MB
INFO: Current Start LSN: 0/16000028, TLI: 1
INFO: Parent Start LSN: 0/13000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 0
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/170000D0 in archived WAL segment /var/probackup/wal/ent-13/0000000100000000000000017
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAGK
INFO: Backup S7DAGK data files are valid
INFO: Backup S7DAGK resident size: 460kB
INFO: Backup S7DAGK completed
```

Проверим каталог резервных копий:

```
student$ pg_probackup show -B /var/probackup --instance ent-13

=====
Instance Version ID Recovery Time Mode WAL Mode TLI Time Data WAL Zratio Start LSN Stop LSN Status
=====
ent-13 13 S7DAGK 2024-01-16 21:38:45.728813+03 PAGE ARCHIVE 1/1 2s 460kB 16MB 1,00 0/16000028 0/170000D0 OK
ent-13 13 S7DAGE 2024-01-16 21:38:40.396964+03 FULL ARCHIVE 1/0 6s 334MB 16MB 1,00 0/13000028 0/140000D0 OK
ent-13 13 S7DAG6 2024-01-16 21:38:32.830407+03 FULL STREAM 1/0 6s 334MB 16MB 1,00 0/10000028 0/1000C168 OK
=====
```

Попробуем восстановить кластер из этого архива.

```
student$ sudo systemctl stop postgrespro-ent-13.service
```

Удаляем содержимое PGDATA.

```
postgres$ rm -rf /var/lib/pgpro/ent-13/*
```

Восстанавливаем. К команде добавим ключи для автоматического формирования restore_command.

```
student$ pg_probackup restore -B /var/probackup --instance ent-13 -i S7DAGK --archive-host=localhost --archive-user=student

INFO: Validating parents for backup S7DAGK
INFO: Validating backup S7DAGE
INFO: Backup S7DAGE data files are valid
INFO: Validating backup S7DAGK
INFO: Backup S7DAGK data files are valid
INFO: Backup S7DAGK WAL segments are valid
INFO: Backup S7DAGK is valid.
INFO: Restoring the database from backup at 2024-01-16 21:38:44+03
INFO: Start restoring backup files. PGDATA size: 334MB
INFO: Backup files are restored. Transferred bytes: 334MB, time elapsed: 2s
INFO: Restore incremental ratio (less is better): 100% (334MB/334MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 3s
INFO: Restore of backup S7DAGK completed.
```

Стартуем Postgres Pro Enterprise.

```
student$ sudo systemctl start postgrespro-ent-13.service
```

```
student$ psql
```

```
=> SELECT * FROM t1;

      msg
-----
Проверим режим PAGE.
(1 row)
```

Проверим пишущую транзакцию.

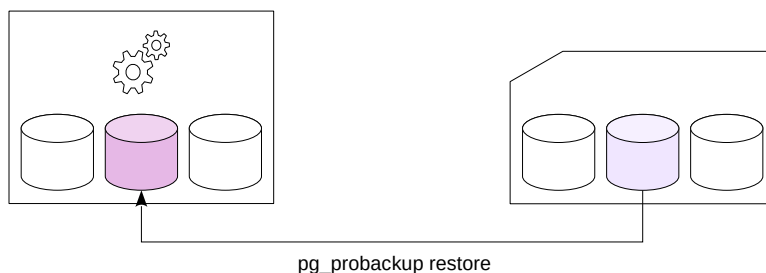
```
=> INSERT INTO t1 VALUES ('Проверим пишущую транзакцию.');
```

```
INSERT 0 1

=> SELECT * FROM t1;

      msg
-----
Проверим режим PAGE.
Проверим пишущую транзакцию.
(2 rows)
```

После удаленного восстановления данные не потеряны, экземпляр работоспособен.



Восстановление только требуемых баз данных кластера

--db-include — база данных для восстановления

--db-exclude — база данных, которую не надо восстанавливать

Экспериментальная возможность, требует осторожности

10

Частичное восстановление применяется в случаях, когда не требуется восстанавливать все базы данных кластера. Эта возможность экспериментальная, ее не следует использовать в промышленной эксплуатации.

Ключом --db-include можно задать имя базы данных для восстановления. Базы template0 и template1 восстанавливаются всегда. Можно указать несколько ключей, по одному для каждой восстанавливаемой базы данных.

Используя ключи --db-exclude, можно попросить утилиту восстановить все базы, кроме указанных.

При частичном восстановлении потребуется столько же свободного места на диске, сколько и для полного восстановления.

Роль, от имени которой утилита pg_probackup подключается к кластеру баз данных, должна иметь право на чтение таблицы pg_catalog.pg_database.

Частичное восстановление

Для частичного восстановления роль backup должна иметь право на чтение pg_catalog.pg_database в базе данных backup.

=> \c backup

You are now connected to database "backup" as user "student".

=> GRANT SELECT ON TABLE pg_catalog.pg_database TO backup;

GRANT

Для демонстрации частичного восстановления создадим базу данных и таблицу.

=> CREATE DATABASE fatedb;

CREATE DATABASE

=> \c fatedb

You are now connected to database "fatedb" as user "student".

=> CREATE TABLE prof_fate (button text);

CREATE TABLE

=> INSERT INTO prof_fate VALUES ('Push the Button, Max!');

INSERT 0 1

Выполним инкрементальное копирование, созданная база попадет в резервную копию.

student\$ pg_probackup backup -B /var/probackup --instance ent-13 -b PAGE

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAGV, backup mode: PAGE, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
WARNING: Valid full backup on current timeline 2 is not found, trying to look up on previous timelines
INFO: Parent backup: S7DAGK
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000020000000000000019 to be archived
INFO: PGDATA size: 342MB
INFO: Current Start LSN: 0/19000028, TLI: 2
INFO: Parent Start LSN: 0/16000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 0
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/1A000888 in archived WAL segment /var/probackup/wal/ent-13/00000002000000000000001A
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Backup S7DAGV resident size: 27MB
INFO: Backup S7DAGV completed

Проверим каталог резервных копий:

student\$ pg_probackup show -B /var/probackup --instance ent-13

Instance	Version	ID	Recovery Time		Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAGV	2024-01-16	21:38:56.792804+03	PAGE	ARCHIVE	2/1	2s	27MB	16MB	1,00	0/19000028	0/1A000888	OK
ent-13	13	S7DAGK	2024-01-16	21:38:45.728813+03	PAGE	ARCHIVE	1/1	2s	460kB	16MB	1,00	0/16000028	0/170000D0	OK
ent-13	13	S7DAGE	2024-01-16	21:38:40.396964+03	FULL	ARCHIVE	1/0	6s	334MB	16MB	1,00	0/13000028	0/140000D0	OK
ent-13	13	S7DAG6	2024-01-16	21:38:32.830407+03	FULL	STREAM	1/0	6s	334MB	16MB	1,00	0/10000028	0/1000C168	OK

Команда merge объединяет полную копию со всеми дочерними инкрементальными:

student\$ pg_probackup merge -B /var/probackup --instance ent-13 -i S7DAGV

INFO: Merge started
INFO: Merging backup S7DAGV with parent chain
INFO: Validate parent chain for backup S7DAGV
INFO: Validating backup S7DAGE
INFO: Backup S7DAGE data files are valid
INFO: Validating backup S7DAGK
INFO: Backup S7DAGK data files are valid
INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Start merging backup files
INFO: Backup files are successfully merged, time elapsed: 4s
INFO: Delete: S7DAGK 2024-01-16 21:38:45+03
INFO: Delete: S7DAGV 2024-01-16 21:38:56+03
INFO: Rename merged full backup S7DAGE to S7DAGV
INFO: Validating backup S7DAGE
INFO: Backup S7DAGE data files are valid
INFO: Merge of backup S7DAGV completed

Останавливаем экземпляр.

student\$ sudo systemctl stop postgrespro-ent-13.service

Удаляем содержимое PGDATA.

postgres\$ rm -rf /var/lib/pgpro/ent-13/*

Сначала восстановим кластер целиком.

student\$ pg_probackup restore -B /var/probackup --instance ent-13 -i S7DAGV --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-user=student

INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Backup S7DAGV WAL segments are valid
INFO: Backup S7DAGV is valid.
INFO: Restoring the database from backup at 2024-01-16 21:38:55+03
INFO: Start restoring backup files. PGDATA size: 334MB
INFO: Backup files are restored. Transferred bytes: 342MB, time elapsed: 2s
INFO: Restore incremental ratio (less is better): 102% (342MB/334MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 3s
INFO: Restore of backup S7DAGV completed.

Срапчем.

student\$ sudo systemctl start postgrespro-ent-13.service

student\$ psql

Восстановились все базы данных:

=> \l

Name	Owner	Encoding	List of databases		Access privileges
			Collate	Ctype	
backup	backup	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
demo	student	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
fatedb	student	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
postgres	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
student	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
template0	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	=c/postgres +
template1	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	postgres=Ctc/postgres
					=c/postgres +
postgres=Ctc/postgres					
(7 rows)					

Таблица в базе fatedb тоже восстановлена:

```
=> \c fatedb
```

You are now connected to database "fatedb" as user "student".

```
=> SELECT * FROM prof_fate;
```

```
      button
-----
Push the Button, Max!
(1 row)
```

Теперь восстановим все базы данных кластера кроме fatedb.

```
student$ sudo systemctl stop postgrespro-ent-13.service
```

Снова удаляем содержимое PGDATA.

```
postgres$ rm -rf /var/lib/pgpro/ent-13/*
```

Добавим к командной строке ключ --db-exclude, чтобы база fatedb не восстанавливалась.

```
student$ pg_probackup restore -B /var/probackup --instance ent-13 -i S7DAGV --db-exclude=fatedb --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-user=s
```

```
INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Backup S7DAGV WAL segments are valid
INFO: Backup S7DAGV is valid.
INFO: Restoring the database from backup at 2024-01-16 21:38:55+03
INFO: Start restoring backup files. PGDATA size: 334MB
INFO: Backup files are restored. Transferred bytes: 334MB, time elapsed: 2s
INFO: Restore incremental ratio (less is better): 100% (334MB/334MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 3s
INFO: Restore of backup S7DAGV completed.
```

Стартуем.

```
student$ sudo systemctl start postgrespro-ent-13.service
```

```
student$ psql
```

База данных fatedb есть в списке:

```
=> \l
```

Name	Owner	Encoding	List of databases		Access privileges
			Collate	Ctype	
backup	backup	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
demo	student	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
fatedb	student	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
postgres	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
student	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
template0	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	=c/postgres +
template1	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	postgres=Ctc/postgres
					=c/postgres +
postgres=Ctc/postgres					
(7 rows)					

Однако подключиться к ней нельзя:

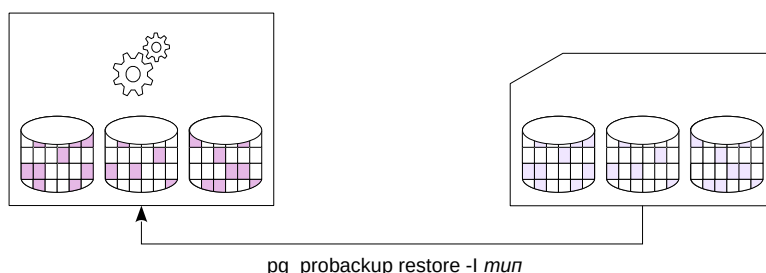
```
student$ psql fatedb
```

```
psql: error: FATAL:  "base/24731" is not a valid data directory
DETAIL:  File "base/24731/PG_VERSION" does not contain valid data.
HINT:  You might need to initdb.
```

Подкаталог PGDATA этой базы пуст.

```
postgres$ du -ch /var/lib/pgpro/ent-13/base/24731 | tail -n 1
```

```
4,0K    total
```



Инкрементальное восстановление страниц — оптимизация
Восстанавливаются только некорректные или измененные
страницы

CHECKSUM — восстановление страниц с некорректными
контрольными суммами

LSN — по точке расхождения каталога с данными от резервной копии

12

Инкрементальное восстановление работает при непустом каталоге данных экземпляра, а восстанавливаются только некорректные или измененные страницы. Инкрементальное восстановление включается параметром `-l` команды `pg_probackup restore`, после которого указывают требуемый тип восстановления. Рекомендуется инкрементальное восстановление CHECKSUM, при котором читаются все файлы в целевом каталоге, проверяются заголовки и контрольные суммы страниц. Восстанавливаются только некорректные страницы и те, в которых контрольная сумма и LSN отличаются от значений в странице в копии.

При типе восстановления LSN из файла `pg_control` восстанавливаемого экземпляра извлекаются значения REDO LSN и REDO TLI, позволяющие определить точку в истории (точку сдвига), в которой состояние каталога данных сдвинулось с цепочки резервных копий. Если эта точка достижима, то читаются все файлы данных в каталоге данных, проверяются заголовки и контрольные суммы страниц, затем восстанавливаются только страницы с неверной контрольной суммой или с LSN, превышающим позицию точки сдвига. При этом требуется, чтобы контрольные суммы были включены, а также необходима синхронность файла `pg_control` с состоянием каталога данных. При повреждении файла `pg_control` этот режим использовать нельзя.

Тип NONE отключает инкрементальную оптимизацию.

Возможный сценарий использования: ускоренное восстановление бывшего ведущего сервера после переключения на реплику.

Инкрементальное восстановление

```
student$ sudo systemctl stop postgrespro-ent-13.service
```

Инкрементальное восстановление включается параметром -I с указанием режима.

Выполним инкрементальное восстановление в режиме CHECKSUM:

```
student$ pg_probackup restore -B /var/probackup --instance ent-13 -i S7DAGV -I CHECKSUM --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-user=student
```

```
INFO: Running incremental restore into nonempty directory: "/var/lib/pgpro/ent-13"
INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Backup S7DAGV WAL segments are valid
INFO: Backup S7DAGV is valid.
INFO: Restoring the database from backup at 2024-01-16 21:38:55+03
INFO: Extracting the content of destination directory for incremental restore
INFO: Destination directory content extracted, time elapsed: 0
INFO: Removing redundant files in destination directory
INFO: Redundant files are removed, time elapsed: 0
INFO: Start restoring backup files. PGDATA size: 334MB
INFO: Backup files are restored. Transferred bytes: 8337kB, time elapsed: 1s
INFO: Restore incremental ratio (less is better): 2% (8337kB/334MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 3s
INFO: Restore of backup S7DAGV completed.
```

```
student$ sudo systemctl start postgrespro-ent-13.service
```

```
student$ psql
```

База данных fatedb должна быть восстановлена, и запрос к таблице в этой базе должен выполняться успешно.

```
=> \c fatedb
```

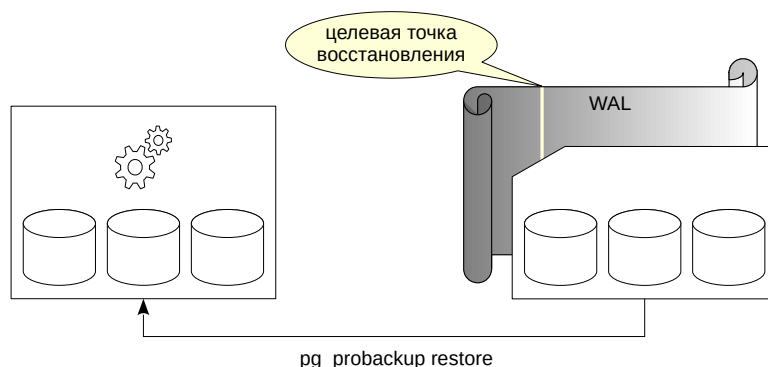
You are now connected to database "fatedb" as user "student".

```
=> SELECT * FROM prof_fate;
```

```
      button
-----
 Push the Button, Max!
(1 row)
```

```
=> \c student
```

You are now connected to database "student" as user "student".



При наличии архива WAL можно восстановить систему на момент времени — point-in-time recovery

Требуется архив WAL от момента создания резервной копии

14

Если до выполнения резервного копирования уже был настроен архив WAL, то с помощью `pg_probackup` можно выполнить восстановление на момент времени в прошлом (Point-in-Time Recovery, PITR). Резервные копии могут как включать журнальные файлы (автономные копии), так и не включать, однако обязательно требуется наличие архива WAL как минимум с момента создания резервной копии.

При восстановлении на момент времени не обязательно указывать идентификатор копии. Утилита `pg_probackup` автоматически выбирает резервную копию, ближайшую к заданной целевой точке, и начинает восстановление.

Указать точку восстановления можно ключами:

- `--recovery-target-time` — на заданное время в формате timestamp;
- `--recovery-target-xid` — до заданной транзакции;
- `--recovery-target-lsn` — до заданного LSN;
- `--recovery-target-name` — до заранее созданной именованной точки;
- `--recovery-target="latest"` — до последнего согласованного состояния;
- `--recovery-target='immediate'` — до самого раннего согласованного состояния.

Восстановление на момент в прошлом

Пока в каталоге лишь две полные копии.

```
student$ pg_probackup show -B /var/probackup --instance ent-13
```

Instance	Version	ID	Recovery Time		Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAGV	2024-01-16	21:38:56.792804+03	FULL	ARCHIVE	2/0	4s	342MB	16MB	1,00	0/19000028	0/1A000888	OK
ent-13	13	S7DAG6	2024-01-16	21:38:32.830407+03	FULL	STREAM	1/0	6s	334MB	16MB	1,00	0/10000028	0/1000C168	OK

Добавим еще одну запись к таблице и выполним инкрементальное копирование.

```
=> SELECT * FROM t1;
```

msg

```

Проверим режим PAGE.
Проверим пишущую транзакцию.
(2 rows)

```

```
=> INSERT INTO t1 VALUES( 'Запись попадет в инкрементальную копию.' );
```

```
INSERT 0 1
```

```
student$ pg_probackup backup -B /var/probackup --instance ent-13 -b PAGE
```

```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAHT, backup mode: PAGE, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
WARNING: Valid full backup on current timeline 5 is not found, trying to look up on previous timelines
INFO: Parent backup: S7DAGV
INFO: Wait for WAL segment /var/probackup/wal/ent-13/00000005000000000000001C to be archived
INFO: PGDATA size: 342MB
INFO: Current Start LSN: 0/1C000028, TLI: 5
INFO: Parent Start LSN: 0/19000028, TLI: 2
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 1s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/1D000088 in archived WAL segment /var/probackup/wal/ent-13/00000005000000000000001D
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 1s
INFO: Validating backup S7DAHT
INFO: Backup S7DAHT data files are valid
INFO: Backup S7DAHT resident size: 21MB
INFO: Backup S7DAHT completed
```

Проверим каталог резервных копий:

```
students$ pg_probackup show -B /var/probackup --instance ent-13
```

Instance	Version	ID	Recovery Time			Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAHT	2024-01-16	21:39:30.302291+03	PAGE	ARCHIVE	5/2	2s	21MB	16MB	1,00	0/1C000028	0/1D000888	OK	
ent-13	13	S7DAGV	2024-01-16	21:38:56.792804+03	FULL	ARCHIVE	2/0	4s	342MB	16MB	1,00	0/19000028	0/1A000888	OK	
ent-13	13	S7DAG6	2024-01-16	21:38:32.830407+03	FULL	STREAM	1/0	6s	334MB	16MB	1,00	0/10000028	0/1000C168	OK	

Используя команду `validate`, можно проверить возможность восстановления до заданной целевой точки.

Идентификатор копии указывать не требуется. Проверим, можно ли восстановиться до заданной позиции WAL:

```
student$ pg_probackup validate -B /var/probackup --instance ent-13 --recovery-target-lsn=0/1A000888
```

```
INFO: Validating backup S7DAGV
INFO: Backup S7DAGV data files are valid
INFO: Backup validation completed successfully on time 2024-01-16 21:38:56.792804+03, xid 749 and LSN 0/1A000888
INFO: Backup S7DAGV is valid.
INFO: Validate of backup S7DAGV completed.
```

Выполним восстановление до LSN 0/1A000888.

```
students$ sudo systemctl stop postgrespro-ent-13.service
```

Удаляем содержимое PGDATA и запускаем команду restore:

```
postgres$ rm -rf /var/lib/pgpro/ent-13/*
```

```
student$ pg_probackup restore -B /var/probackup --instance ent-13 --recovery-target-lsn=0/1A000888 --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-use
```

```
INFO: Validating backup $7DAGV
INFO: Backup $7DAGV data files are valid
INFO: Backup validation completed successfully on time 2024-01-16 21:38:56.792804+03, xid 749 and LSN 0/1A000888
INFO: Backup $7DAGV is valid.
INFO: Restoring the database from backup at 2024-01-16 21:38:55+03
INFO: Start restoring backup files. PGDATA size: 334MB
INFO: Backup files are restored. Transferred bytes: 342MB, time elapsed: 2s
INFO: Restore incremental ratio (less is better): 102% (342MB/334MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 3s
INFO: Restore of backup $7DAGV completed.
```

Стартуем.

```
students$ sudo systemctl start postgrespro-ent-13.service
```

```
student$ psql
```

```
=> SELECT * FROM t1;
```

msq

```

Проверим режим PAGE.
Проверим пишущую транзакцию.
(2 rows)

```

Восстановление приостановлено после достижения целевой точки:

```
=> SELECT pg_is_wal_replay_paused();
```

```
pg is wal replay paused
```

$$\begin{array}{c} \text{--} \text{--} \\ \text{+} \end{array}$$

(1 row)

Завершим восстановление.

```
=> SELECT pg_wal_replay_resume();
```

```
pg_wal_replay_resume
```

```
(1 row)
```

```
=> SELECT pg_is_in_recovery();
```

```
pg_is_in_recovery
```

```
f
```

```
(1 row)
```

Данные восстановлены на момент, соответствующий LSN 0/1A000888.

Выполним полное копирование с архивированием WAL, так как текущий экземпляр восстановлен до точки времени в прошлом.

```
student$ pg_probackup backup -B /var/probackup --instance ent-13 -b FULL
```

```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAI5, backup mode: FULL, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Wait for WAL segment /var/probackup/wal/ent-13/0000000600000000000000001C to be archived
INFO: PGDATA size: 342MB
INFO: Current Start LSN: 0/1C000028, TLI: 6
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 2s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/1D0000D0 in archived WAL segment /var/probackup/wal/ent-13/0000000600000000000000001D
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 3s
INFO: Validating backup S7DAI5
INFO: Backup S7DAI5 data files are valid
INFO: Backup S7DAI5 resident size: 342MB
INFO: Backup S7DAI5 completed
```

При удаленной работе каталог резервных копий находится на выделенном сервере

Полное резервирование и три режима инкрементального: DELTA, PAGE и PTRACK

Резервное копирование и восстановление может использовать потоковую доставку и файловый архив WAL

1. Подготовьте экземпляр к удаленной работе с архивированием WAL.
2. Подключите и настройте PTRACK.
3. Сравните время, затрачиваемое на создание полной и инкрементальных копий в режимах DELTA, PAGE и PTRACK.
4. Добавьте сжатие и проведите такие же измерения времени создания копий.
5. Восстановите данные из последней сжатой инкрементальной копии.

1. Подготовьте и инициализируйте каталог, зарегистрируйте роли и базу данных так же, как в демонстрации. Не забудьте указать параметры `--remote-host=localhost` `--remote-user=postgres` утилиты `pg_probackup`, а также включить режим архивирования WAL в экземпляре СУБД.
2. Настройте параметр СУБД `shared_preload_libraries = 'ptrack'` и перезагрузите экземпляр. Задайте размер `ptrack.map_size = '1MB'`, как в демонстрации. Подключите расширение `ptrack`.
3. Создайте полную и инкрементальные копии в трех режимах, не забывая добавлять перед `pg_probackup` команду `time` для вывода затраченного времени. Между командами резервного копирования регистрируйте любые новые учетные записи и создавайте принадлежащие этим пользователям базы данных для генерации данных, которые будут попадать в инкрементальные копии. Сравните полученные результаты с помощью `pg_probackup show`.
4. Прodelайте такой же эксперимент, добавив `--compress-algorithm=zlib`.
5. Восстановитесь из последней сжатой резервной копии, не забывая параметры `pg_probackup --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-user=student`.

1. Подготовка каталога резервных копий и регистрация экземпляра. Режим WAL

Создаем и подготавливаем каталог копий.

```
student$ sudo mkdir /var/probackup
```

```
student$ sudo chown student: /var/probackup
```

Инициализируем каталог.

```
student$ pg_probackup init -B /var/probackup
```

```
INFO: Backup catalog '/var/probackup' successfully initialized
```

Регистрируем экземпляр.

```
student$ pg_probackup add-instance -B /var/probackup -D /var/lib/pgpro/ent-13 --instance ent-13 --remote-host=localhost --remote-user=postgres
```

```
INFO: Instance 'ent-13' successfully initialized
```

```
student$ psql
```

Создадим роль backup и базу данных, к которой будет подключаться эта роль.

```
=> CREATE ROLE backup LOGIN REPLICATION PASSWORD 'b@ckUp';
```

CREATE ROLE

```
student$ echo "localhost:5432:backup:backup:b@ckUp" > ~/.pgpass && chmod 600 ~/.pgpass && cat ~/.pgpass
```

```
localhost:5432:backup:backup:b@ckUp
```

```
=> CREATE DATABASE backup OWNER backup;
```

CREATE DATABASE

Предоставляем права.

```
=> \c backup
```

```
You are now connected to database "backup" as user "student".
```

```
=>
```

```
BEGIN;
GRANT USAGE ON SCHEMA pg_catalog TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.current_setting(text) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.set_config(text, text, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_is_in_recovery() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_start_backup(text, boolean, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_stop_backup(boolean, boolean) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_create_restore_point(text) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_switch_wal() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_last_wal_replay_lsn() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_current() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_current_snapshot() TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.txid_snapshot_xmax(txid_snapshot) TO backup;
GRANT EXECUTE ON FUNCTION pg_catalog.pg_control_checkpoint() TO backup;
COMMIT;
```

```
BEGIN
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
GRANT
```

```
COMMIT
```

```
=> \c student
```

```
You are now connected to database "student" as user "student".
```

Запоминаем в настройках конфигурации атрибуты подключения.

```
student$ pg_probackup set-config -B /var/probackup --instance ent-13 -d backup -U backup --remote-host=localhost --remote-user=postgres
```

Подготовимся к работе в режиме архивирования WAL.

```
=> ALTER SYSTEM SET archive_mode = on;
```

ALTER SYSTEM

```
=> ALTER SYSTEM SET archive_command = '/opt/pgpro/ent-13/bin/pg_probackup archive-push -B /var/probackup --instance=ent-13 --wal-file-path=%p --wal-file-name=%f --remote-host=localhost
```

ALTER SYSTEM

Выполним перезагрузку.

```
student$ sudo systemctl restart postgrespro-ent-13.service
```

```
student$ psql
```

Проверим работоспособность архивирования WAL.

```
=> SELECT pg_switch_wal();
```

```
pg_switch_wal
```

```
-----
```

```
0/F175468
```

```
(1 row)
```

```
=> CHECKPOINT;
```

CHECKPOINT

```
student$ pg_probackup show -B /var/probackup --archive
```

```
INFO: checking WAL file name "000000010000000000000000F"
```

```
ARCHIVE INSTANCE 'ent-13'
```

TLI	Parent TLI	Switchpoint	Min Segno	Max Segno	N segments	Size	Zratio	N backups	Status
1	0	0/0	000000010000000000000000F	000000010000000000000000F	1	16MB	1,00	0	OK

2. Подключение и настройка PTRACK

```
=> ALTER SYSTEM SET shared_preload_libraries = 'ptrack';
```

ALTER SYSTEM

```
=> ALTER SYSTEM SET client_min_messages TO error;
```

ALTER SYSTEM

Требуется перезагрузка экземпляра.

```
student$ sudo systemctl restart postgrespro-ent-13.service
```

```
student$ psql
```

Подключим расширение и зададим значение для параметра `ptrack.map_size`.

```
=> \c backup
```

You are now connected to database "backup" as user "student".

```
=> CREATE EXTENSION IF NOT EXISTS ptrack;
```

```
CREATE EXTENSION
```

Достаточное значение для `ptrack.map_size` здесь — 1 Мбайт.

```
=> ALTER SYSTEM SET ptrack.map_size = '1MB';
```

```
ALTER SYSTEM
```

Снова требуется перезагрузка экземпляра.

```
student$ sudo systemctl restart postgrespro-ent-13.service
```

```
student$ psql
```

Проверим работоспособность PTRACK с помощью вызова функции `ptrack_version()`.

```
=> \c backup
```

You are now connected to database "backup" as user "student".

```
=> SELECT ptrack_version();
```

```
 ptrack_version
-----
2.5
(1 row)
```

3. Сравнение полных и инкрементальных копий

Полная копия без сжатия.

```
student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b FULL
```

```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAWF, backup mode: FULL, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000010000000000000013 to be archived
INFO: PGDATA size: 335MB
INFO: Current Start LSN: 0/13000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 2s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/14007CD0 in archived WAL segment /var/probackup/wal/ent-13/000000010000000000000014
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 4s
INFO: Validating backup S7DAWF
INFO: Backup S7DAWF data files are valid
INFO: Backup S7DAWF resident size: 335MB
INFO: Backup S7DAWF completed
```

```
real    0m6,854s
user    0m0,537s
sys     0m0,520s
```

Небольшая активность...

```
student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user1
```

```
student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user1 user1
```

Разностная копия.

```
student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b DELTA
```

```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAWM, backup mode: DELTA, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAWF
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000010000000000000016 to be archived
INFO: PGDATA size: 343MB
INFO: Current Start LSN: 0/16000028, TLI: 1
INFO: Parent Start LSN: 0/13000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 0
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/17000920 in archived WAL segment /var/probackup/wal/ent-13/000000010000000000000017
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAWM
INFO: Backup S7DAWM data files are valid
INFO: Backup S7DAWM resident size: 9471kB
INFO: Backup S7DAWM completed
```

```
real    0m2,209s
user    0m0,134s
sys     0m0,124s
```

Небольшая активность...

```
student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user2
```

```
student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user2 user2
```

Инкрементальная копия измененных страниц по записям в WAL.

```
student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b PAGE
```

```
INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAWP, backup mode: PAGE, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAWM
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000010000000000000019 to be archived
INFO: PGDATA size: 351MB
INFO: Current Start LSN: 0/19000028, TLI: 1
INFO: Parent Start LSN: 0/16000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 1s
INFO: wait for pg_stop_backup()
INFO: pg_stop_backup() successfully executed
INFO: Wait for LSN 0/1A000A38 in archived WAL segment /var/probackup/wal/ent-13/00000001000000000000001A
```

```
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAWP
INFO: Backup S7DAWP data files are valid
INFO: Backup S7DAWP resident size: 9428kB
INFO: Backup S7DAWP completed

real    0m1,799s
user    0m0,082s
sys     0m0,047s

Небольшая активность...

student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user3

student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user3 user3

Инкрементальная копия измененных страниц по карте изменений PTRACK.

student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b PTRACK

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAWR, backup mode: PTRACK, wal mode: ARCHIVE, remote: true, compress-algorithm: none, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAWP
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000010000000000000001C to be archived
INFO: PGDATA size: 359MB
INFO: Current Start LSN: 0/1C000028, TLI: 1
INFO: Parent Start LSN: 0/19000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 0
INFO: wait for pg_stop_backup()
INFO: pg_stop backup() successfully executed
INFO: Wait for LSN 0/1D004AC8 in archived WAL segment /var/probackup/wal/ent-13/000000010000000000000001D
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAWR
INFO: Backup S7DAWR data files are valid
INFO: Backup S7DAWR resident size: 9432kB
INFO: Backup S7DAWR completed

real    0m1,654s
user    0m0,078s
sys     0m0,049s
```

Сравним полученные результаты.

```
student$ pg_probackup show -B /var/probackup --instance ent-13

=====
Instance Version ID Recovery Time Mode WAL Mode TLI Time Data WAL Zratio Start LSN Stop LSN Status
=====
ent-13 13 S7DAWR 2024-01-16 21:48:28.336875+03 PTRACK ARCHIVE 1/1 1s 9432kB 16MB 1,00 0/1C000028 0/1D004AC8 OK
ent-13 13 S7DAWP 2024-01-16 21:48:26.030399+03 PAGE ARCHIVE 1/1 1s 9428kB 16MB 1,00 0/19000028 0/1A000A38 OK
ent-13 13 S7DAWM 2024-01-16 21:48:23.700357+03 DELTA ARCHIVE 1/1 2s 9471kB 16MB 1,00 0/16000028 0/17000920 OK
ent-13 13 S7DAWF 2024-01-16 21:48:17.244315+03 FULL ARCHIVE 1/0 6s 335MB 16MB 1,00 0/13000028 0/14007CD0 OK
=====
```

4. Сравнение полных и инкрементальных копий со сжатием

Добавим сжатие в настройки по умолчанию.

```
student$ pg_probackup set-config -B /var/probackup --instance ent-13 --compress-algorithm=zlib

Полная копия со сжатием.

student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b FULL

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAWT, backup mode: FULL, wal mode: ARCHIVE, remote: true, compress-algorithm: zlib, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Wait for WAL segment /var/probackup/wal/ent-13/000000010000000000000001F to be archived
INFO: PGDATA size: 359MB
INFO: Current Start LSN: 0/1F000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 5s
INFO: wait for pg_stop_backup()
INFO: pg_stop backup() successfully executed
INFO: Wait for LSN 0/200000120 in archived WAL segment /var/probackup/wal/ent-13/0000000100000000000000020
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 5s
INFO: Validating backup S7DAWT
INFO: Backup S7DAWT data files are valid
INFO: Backup S7DAWT resident size: 106MB
INFO: Backup S7DAWT completed

real    0m12,664s
user    0m1,296s
sys     0m0,828s

Небольшая активность...

student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user4

student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user4 user4

Разностная копия со сжатием.

student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b DELTA

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAX6, backup mode: DELTA, wal mode: ARCHIVE, remote: true, compress-algorithm: zlib, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAWT
INFO: Wait for WAL segment /var/probackup/wal/ent-13/0000000100000000000000022 to be archived
INFO: PGDATA size: 367MB
INFO: Current Start LSN: 0/22000028, TLI: 1
INFO: Parent Start LSN: 0/1F000028, TLI: 1
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 1s
INFO: wait for pg_stop_backup()
INFO: pg_stop backup() successfully executed
INFO: Wait for LSN 0/230000C60 in archived WAL segment /var/probackup/wal/ent-13/0000000100000000000000023
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAX6
INFO: Backup S7DAX6 data files are valid
INFO: Backup S7DAX6 resident size: 3643kB
INFO: Backup S7DAX6 completed

real    0m2,551s
user    0m0,228s
sys     0m0,165s
```

```
Небольшая активность...

student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user5

student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user5 user5

Инкрементальная копия измененных страниц по записям в WAL со сжатием.

student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b PAGE

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAX9, backup mode: PAGE, wal mode: ARCHIVE, remote: true, compress-algorithm: zlib, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAX6
INFO: Wait for WAL segment /var/probackup/wal/ent-13/0000000100000000000000025 to be archived
INFO: PGDATA size: 375MB
INFO: Current Start LSN: 0/25000028, TLI: 1
INFO: Parent Start LSN: 0/22000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 1s
INFO: wait for pg_stop_backup()
INFO: pg_stop backup() successfully executed
INFO: Wait for LSN 0/260000D0 in archived WAL segment /var/probackup/wal/ent-13/0000000100000000000000026
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 1s
INFO: Validating backup S7DAX9
INFO: Backup S7DAX9 data files are valid
INFO: Backup S7DAX9 resident size: 3773kB
INFO: Backup S7DAX9 completed

real    0m2,182s
user    0m0,150s
sys     0m0,071s
```

```
Небольшая активность...

student$ /opt/pgpro/ent-13/bin/createuser -p 5432 user6

student$ /opt/pgpro/ent-13/bin/createdb -p 5432 -O user6 user6

Инкрементальная копия измененных страниц по карте изменений PTRACK со сжатием.

student$ time pg_probackup backup -B /var/probackup --instance ent-13 -b PTRACK

INFO: Backup start, pg_probackup version: 2.6.7, instance: ent-13, backup ID: S7DAXC, backup mode: PTRACK, wal mode: ARCHIVE, remote: true, compress-algorithm: zlib, compress-level: 1
INFO: This PostgreSQL instance was initialized with data block checksums. Data block corruption will be detected
INFO: Database backup start
INFO: wait for pg_start_backup()
INFO: Parent backup: S7DAX9
INFO: Wait for WAL segment /var/probackup/wal/ent-13/0000000100000000000000028 to be archived
INFO: PGDATA size: 383MB
INFO: Current Start LSN: 0/28000028, TLI: 1
INFO: Parent Start LSN: 0/25000028, TLI: 1
INFO: Extracting pagemap of changed blocks
INFO: Pagemap successfully extracted, time elapsed: 0 sec
INFO: Start transferring data files
INFO: Data files are transferred, time elapsed: 0
INFO: wait for pg_stop_backup()
INFO: pg_stop backup() successfully executed
INFO: Wait for LSN 0/290000E8 in archived WAL segment /var/probackup/wal/ent-13/0000000100000000000000029
INFO: Getting the Recovery Time from WAL
INFO: Syncing backup files to disk
INFO: Backup files are synced, time elapsed: 0
INFO: Validating backup S7DAXC
INFO: Backup S7DAXC data files are valid
INFO: Backup S7DAXC resident size: 3652kB
INFO: Backup S7DAXC completed

real    0m2,072s
user    0m0,121s
sys     0m0,099s
```

Сравним полученные результаты.

```
student$ pg_probackup show -B /var/probackup --instance ent-13
```

Instance	Version	ID	Recovery Time	Mode	WAL Mode	TLI	Time	Data	WAL	Zratio	Start LSN	Stop LSN	Status
ent-13	13	S7DAXC	2024-01-16 21:48:48.907140+03	PTRACK	ARCHIVE	1/1	1s	3652kB	16MB	2,58	0/28000028	0/290000E8	OK
ent-13	13	S7DAX9	2024-01-16 21:48:46.316919+03	PAGE	ARCHIVE	1/1	2s	3773kB	16MB	2,60	0/25000028	0/260000D0	OK
ent-13	13	S7DAX6	2024-01-16 21:48:43.655359+03	DELTA	ARCHIVE	1/1	2s	3643kB	16MB	2,59	0/22000028	0/230000C6	OK
ent-13	13	S7DAWT	2024-01-16 21:48:34.698999+03	FULL	ARCHIVE	1/0	11s	106MB	16MB	3,39	0/1F000028	0/20000120	OK
ent-13	13	S7DAWR	2024-01-16 21:48:28.336875+03	PTRACK	ARCHIVE	1/1	1s	9432kB	16MB	1,00	0/1C000028	0/1D004AC8	OK
ent-13	13	S7DAWP	2024-01-16 21:48:26.030399+03	PAGE	ARCHIVE	1/1	1s	9428kB	16MB	1,00	0/19000028	0/1A000A38	OK
ent-13	13	S7DAWM	2024-01-16 21:48:23.700357+03	DELTA	ARCHIVE	1/1	2s	9471kB	16MB	1,00	0/16000028	0/17000920	OK
ent-13	13	S7DAWF	2024-01-16 21:48:17.244315+03	FULL	ARCHIVE	1/0	6s	335MB	16MB	1,00	0/13000028	0/14007CD0	OK

- Полные копии занимают значительно больше места, чем инкрементальные.
- Сжатые копии значительно меньше по размеру, чем несжатые.
- Использование сжатия лишь незначительно замедлило выполнение резервного копирования.

5. Восстановление из сжатой резервной копии

```
Останавливаем экземпляр.

student$ sudo systemctl stop postgrespro-ent-13.service

Удаляем содержимое PGDATA.

postgres$ rm -rf /var/lib/pgpro/ent-13/*

student$ pg_probackup restore -B /var/probackup --instance ent-13 --remote-user=postgres --remote-host=localhost --archive-host=localhost --archive-user=student

INFO: Validating parents for backup S7DAXC
INFO: Validating backup S7DAWT
INFO: Backup S7DAWT data files are valid
INFO: Validating backup S7DAX6
INFO: Backup S7DAX6 data files are valid
INFO: Validating backup S7DAX9
INFO: Backup S7DAX9 data files are valid
INFO: Validating backup S7DAXC
INFO: Backup S7DAXC data files are valid
INFO: Backup S7DAXC WAL segments are valid
INFO: Backup S7DAXC is valid.
INFO: Restoring the database from backup at 2024-01-16 21:48:48+03
INFO: Start restoring backup files. PGDATA size: 383MB
INFO: Backup files are restored. Transferred bytes: 383MB, time elapsed: 3s
INFO: Restore incremental ratio (less is better): 100% (383MB/383MB)
INFO: Syncing restored files to disk
INFO: Restored backup files are synced, time elapsed: 6s
INFO: Restore of backup S7DAXC completed.

Стартуем.

student$ sudo systemctl start postgrespro-ent-13.service

student$ psql
```


=> \du

Role name	List of roles		Member of
	Attributes		
backup	Replication		{ }
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS		{ }
student	Superuser		{ }
user1			{ }
user2			{ }
user3			{ }
user4			{ }
user5			{ }
user6			{ }

=> \l

Name	Owner	Encoding	List of databases		Access privileges
			Collate	Ctype	
backup	backup	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
demo	student	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
postgres	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
student	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
template0	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	=c/postgres + postgres=Ct/postgres
template1	postgres	UTF8	en_US.UTF-8@icu	en_US.UTF-8	=c/postgres + postgres=Ct/postgres
user1	user1	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
user2	user2	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
user3	user3	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
user4	user4	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
user5	user5	UTF8	en_US.UTF-8@icu	en_US.UTF-8	
user6	user6	UTF8	en_US.UTF-8@icu	en_US.UTF-8	

(12 rows)

Данные восстановлены.