PostgresPro

PGCONF.RUSSIA 2024

Новые возможности pg_probackup 3

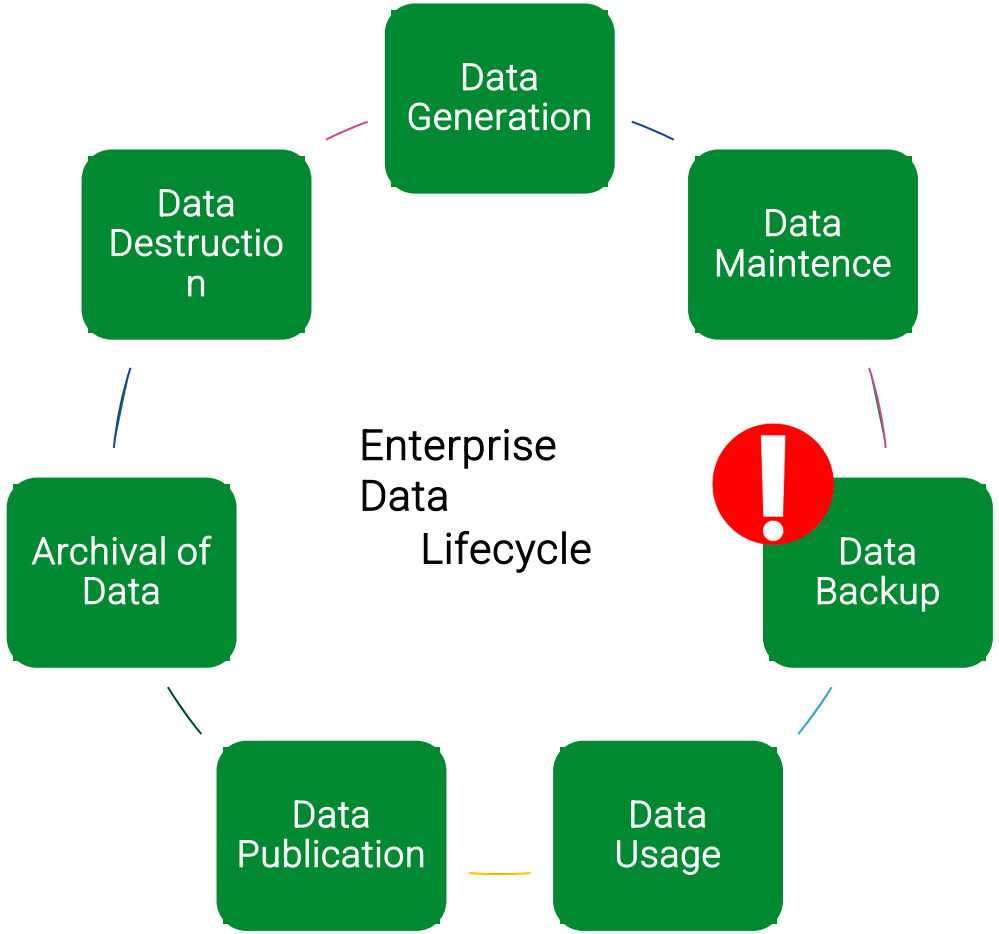
Алексей Дарвин
Менеджер продукта

Дарья Лепихова
Разработчик

PostgresPro

Postgres Professional: цели и задачи

- Накопление уникальных технологических компетенций
- Разработка СУБД мирового уровня
- Активно участвуем в PostgreSQL Community
  - 2 место по участию в PostgreSQL Community
- Развитие собственных решений:
  - Postgres Pro Standart;
  - Postgres Pro Enterprise;
  - Postgres Pro Shardman.



Работа с резервными копиями это одна из ключевых функций при работе с данными

PostgresPro

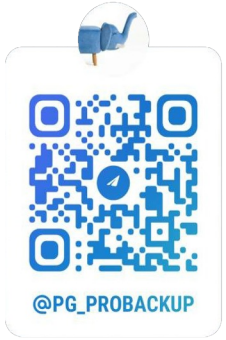
Развиваем pg_probackup с 2016 года

	PG_ProBackup	pg_probackup	pg_base backup	pg_dump/ pg_dumpall	Wal-G	Pgbackrest
2017	**Разные типы инкрементов**	Выходит как часть PGPRO page + ptrack 1.0	В разработке	Только полный dump	Есть	Нет
2018	**Быстрое восстановление, сокращение места на диске**	Выходит как PGPRO Merge	В разработке	Только полный dump	Есть	Нет
2019	**Улучшение инкрементальных режимов, выделение в отдельную сборку**	Версия 2.1 ptrack 2.0	В разработке	Нет	Есть	Есть
2020	**Оптимизация производительности**					
2021	**Максимально быстрая синхронизация отставшей реплики. Встроенная возможность catchup-доливки в реплику**	Версия 2.5 Catchup	Нужны доп инструменты	Нужны доп инструменты	Нужны доп инструменты	Нужны доп инструменты
2022	**Удаленный бэкап S3**	Версия 2.6 S3	Нет	Есть	Есть, без SSH	Нет
2023	**Поддержка специфики PGPRO BiHa, Shardman, CFS**	Версия 2.7	Не оптимально	Не оптимально	Не оптимально	Не оптимально

PostgresPro

Ключевые показатели pg_probackup 2 на текущий момент

- Один из наиболее функциональных инструментов резервного копирования для community версии PostgreSQL
- Лучший инструмент по обеспечению резервного копирования для PGPRO
- 14 поддерживаемых платформ
- Сообщество для пользователей Enterprise Версии PGPRO
- Community ProBackup

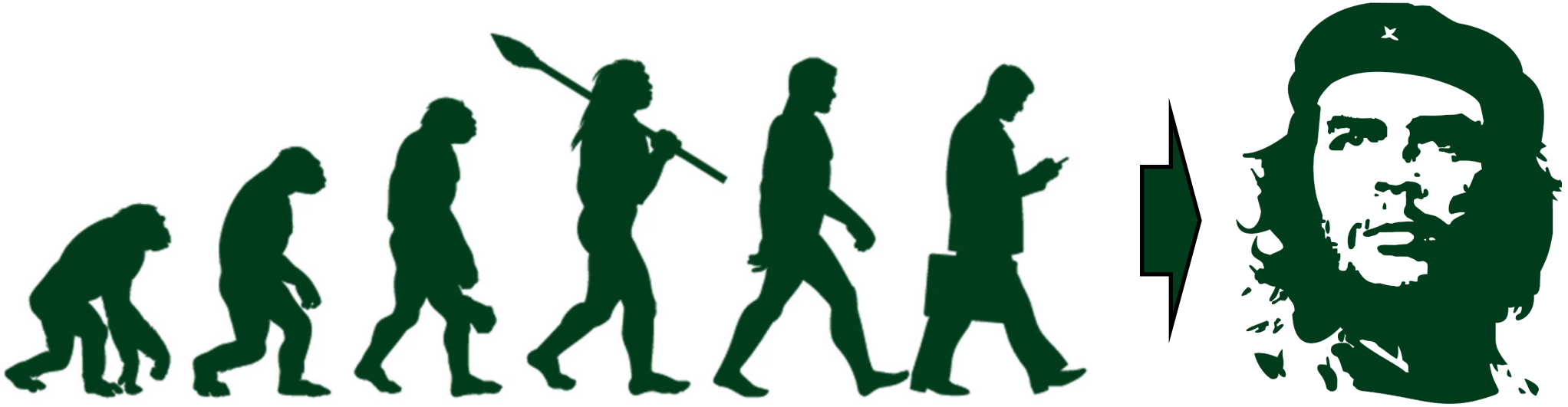


https://t.me/pg_probackup

PostgresPro

Почему появилась потребность в 3 версии

- Новые доработки, сложно/невозможно реализовать на текущей архитектуре
- Накоплен большой объем технического долга
- Появились новые запросы со стороны пользователей



~~Evolution~~ – viva la revolucion!

PostgresPro

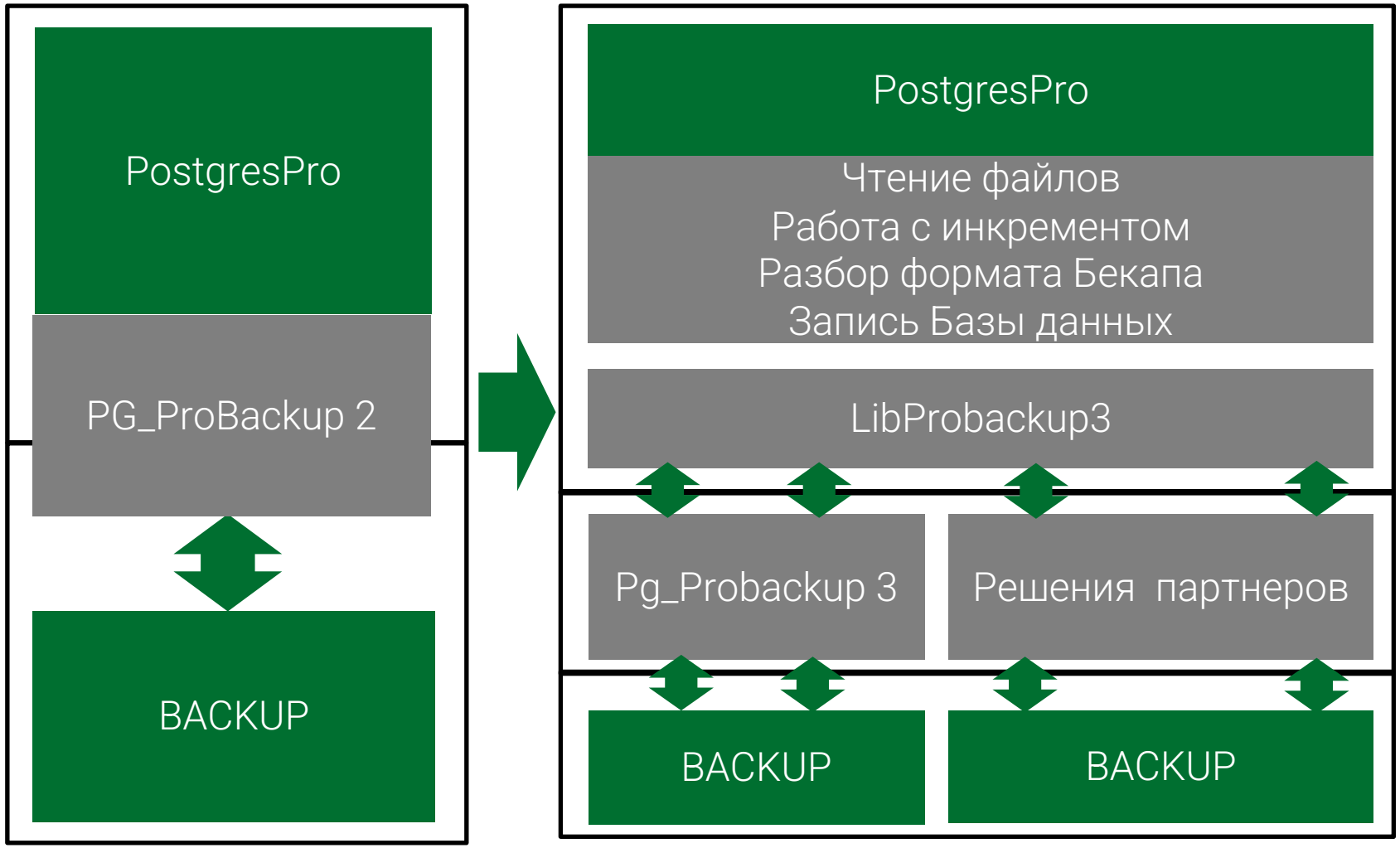
Архитектура pg_probackup3

Новая архитектура решения

- Чтение и запись в несколько потоков
- Интеграция с источниками инкремента: PTRACK, PAGE

Используем доработанный протокол репликации

- Добавляем свои команды в репликационный протокол walsender
- Пересылаем данные в своём формате. PG_PROBACKUP (LABEL=xxx, START_LSN=1123235, COMPRESS_ALG=zstd)



PostgresPro

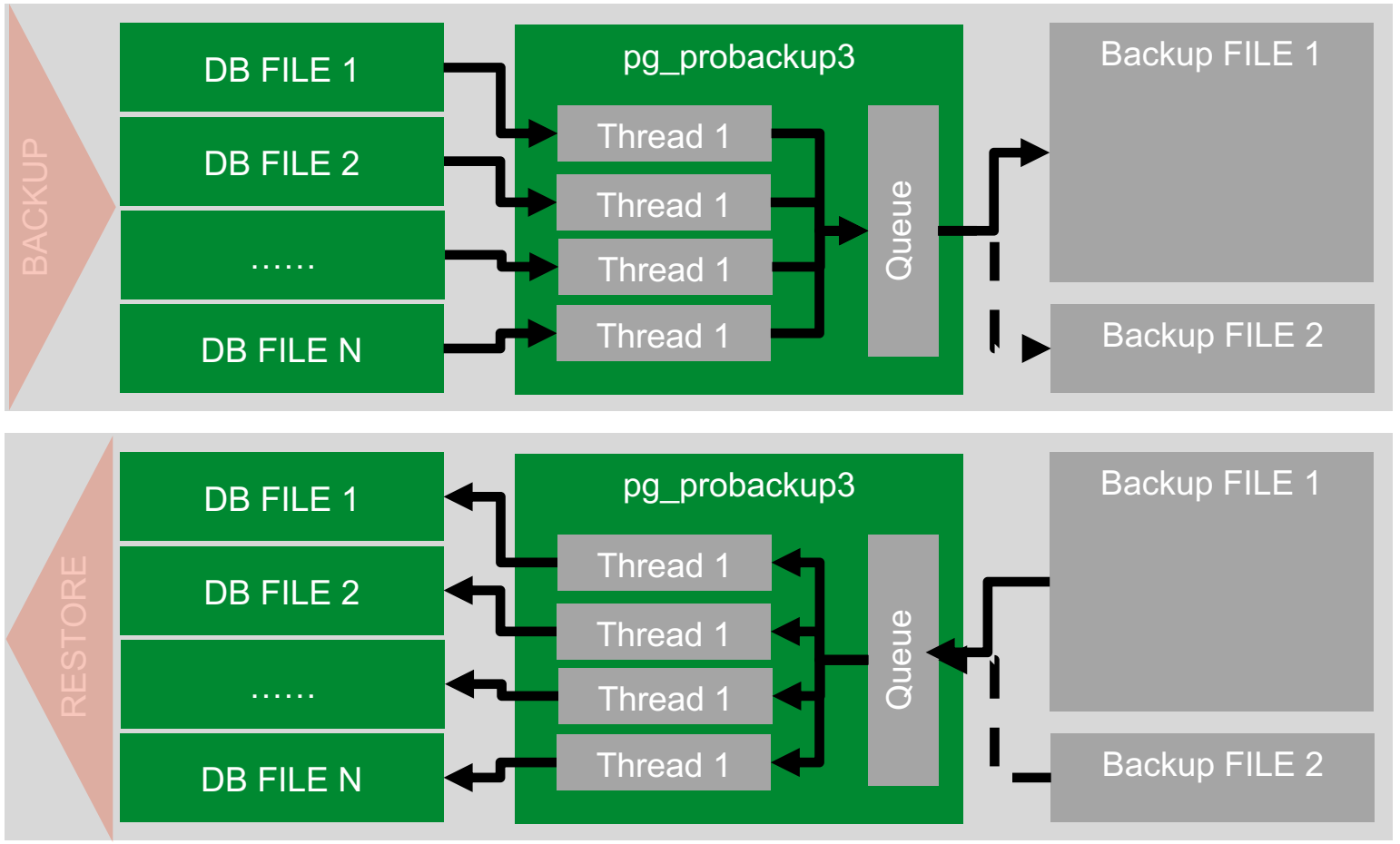
Новые возможности версии 3

- Новый репликационный протокол
  - Добавлен новый режим, в котором применяется новый репликационный протокол (добавляем свои команды в walsender)
  - Не требуется отдельная версия приложения под каждую версию БД
  - Умеет работать как с PGPRO, так и с ванильным PostgreSQL
- Изменен подход с backup файлами
  - Вместо копирования файлов 1:1, перешли к хранению в больших файлах.
  - Работа в многопоточном режиме на чтение и запись
  - Файл меньше по объему, за счет оптимизации хранения информации
  - Можем передавать данные вместе с файлами журнала
  - Можем сжимать передаваемые файлы
  - Оптимизировано для работы с лентами и S3
  - Расширенные настройки retention policy
- Libpgprobackup3
  - Разработана библиотека для интеграции с СРК (Реализован полноценный SDK, использование которого облегчает интеграцию pg_probackup с партнерами)
  - Возможность писать приложения на других языках. SDK поддерживает разработку приложений на Python, C, C++, GoLang
- Не требуется отдельное ssh соединение,
  - Используется соединение по репликационному протоколу like pg_basebackup
- Включены все доработки и оптимизации работы с wal, cfs, Ptrack, S3 имеющиеся в версии 2

PostgresPro

Новый подход к работе с файлами

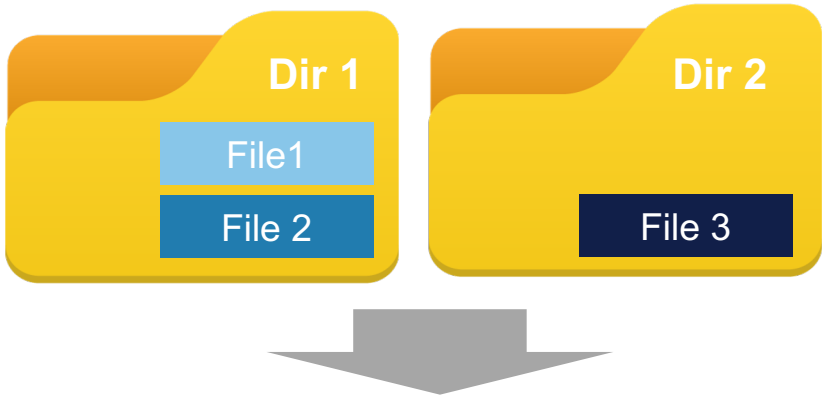
- Несколько читающих потоков (thread) можно мультиплексировать на 1 поток данных (stream) во время сохранения
- 1 поток данных (stream) можно мультиплексировать на несколько пишущих потоков (thread) во время восстановления
- Stream в одном backup процессе может быть несколько, т.е. соотношения будут n:m

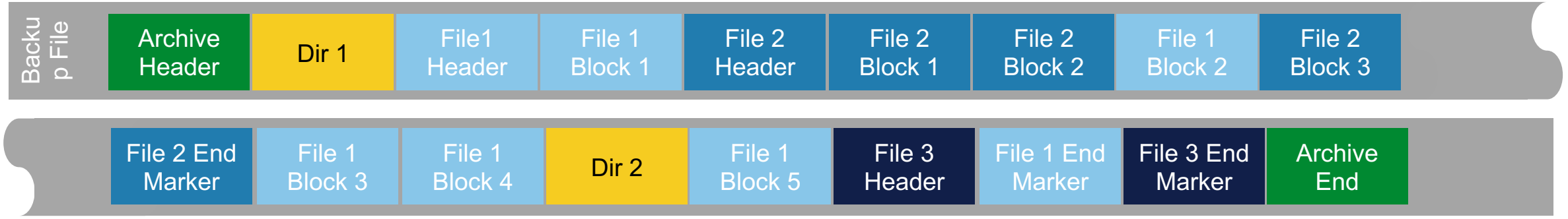


PostgresPro

Новый формат backup файла

- Директориям и файлам присваиваются номера
- Директории пишутся перед файлами
- Файлы ссылаются на директории
- Файлы идут вперемешку
- Для пустых файлов мы также пишем заголовок и end marker (File 3)





PostgresPro

Демонстрация

1. pg_probackup 2 и 3 версии
2. SDK (libpgprobackup3)

PostgresPro

Полный backup в pg_probackup 2

```
daria@daria:~$ pg_probackup init -d $BASE -B $BACKUPDIR2

INFO: Backup catalog '/Users/daria/projects/backup2' successfully initialized

daria@daria:~$ pg_probackup add-instance -d $BASE -B $BACKUPDIR2 -D $DATADIR --instance $INSTANCE

INFO: Instance 'test' successfully initialized
```

PostgresPro

Полный backup в pg_probackup 2

```
daria@daria:~$ pg_probackup backup -d $BASE -B $BACKUPDIR2 --instance $INSTANCE -b FULL –stream

INFO: Backup start, pg_probackup version: 2.7.2, instance: test, backup ID: SBGS11, backup mode: FULL, wal mode: STREAM, remote:
    false, 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: PGDATA size: 42MB
INFO: Current Start LSN: 0/2000028, TLI: 1
INFO: Validating backup SBGS11
INFO: Backup SBGS11 data files are valid
INFO: Backup SBGS11 resident size: 58MB
INFO: Backup SBGS11 completed
```

PostgresPro

Полный backup в pg_probackup 2

```
daria@daria:~$ tree -d $BACKUPDIR2/backups/$INSTANCE
```

```
/Users/daria/projects/pgpro/../backup2/backups/test
├── SBGS11
│   └── database
│       ├── base
│       │   ├── 1
│       │   ├── 16384
│       │   ├── 4
│       │   └── 5
│       ├── global
│       ├── pg_commit_ts
│       ├── pg_dynshmem
│       ├── pg_logical
│       │   ├── mappings
│       │   └── snapshots
│       ├── pg_multixact
│       │   ├── members
│       │   └── offsets
│       ├── pg_notify
│       ├── pg_replslot
│       ├── pg_serial
│       ├── pg_snapshots
│       ├── pg_stat
│       ├── pg_stat_tmp
│       ├── pg_subtrans
│       ├── pg_tblspc
│       ├── pg_twophase
│       ├── pg_wal
│       └── pg_xact
```

PostgresPro

Полный backup в pg_probackup 3

```
daria@daria:~$ pg_probackup3 init -d $BASE -B $BACKUPDIR3

[2024-04-05 13:05:27.294546] [76940] [8000806592] [info] Backup catalog
    '/Users/daria/projects/pgpro/../backup3' successfully initialized

daria@daria:~$ pg_probackup3 add-instance -d $BASE -B $BACKUPDIR3 -D $DATADIR --instance
    $INSTANCE

[2024-04-05 13:05:27.316404] [76941] [8000806592] [info] Instance 'test' successfully initialized
```

PostgresPro

Полный backup в pg_probackup 3 BASE режим

```
daria@daria:~$ pg_probackup3 backup -d $BASE -B $BACKUPDIR3 --instance $INSTANCE -b FULL –
    stream


[2024-04-05 13:05:27.340366] [76943] [8000806592] [info] This PostgreSQL instance was initialized with
    data block checksums. Data block corruption will be detected
[2024-04-05 13:05:27.340473] [76943] [8000806592] [info] IDENTIFY_SYSTEM SYSTEM-
    ID=7354321999301033057 START-TLI1 START-POS=5000000 DBNAME=''
[2024-04-05 13:05:27.341482] [76943] [8000806592] [info] Backup command BASE_BACKUP(LABEL '',
    TABLESPACE_MAP true, WAL true, WAIT false, CHECKPOINT fast, MANIFEST yes);
[2024-04-05 13:05:27.457620] [76943] [8000806592] [info] Backup time 118
INFO: Backup 2024-04-05-13-05-27-013 completed succcessfully.
```

PostgresPro

Полный backup в pg_probackup 3 BASE режим

```
daria@daria:~$ pg_probackup3 backup -d $BASE -B $BACKUPDIR3 --instance $INSTANCE -b FULL –
    stream


ls -la $BACKUPDIR3/backups/$INSTANCE


drwxr-xr-x  7 daria  staff       224 Apr  5 13:05 .
drwxr-xr-x  3 daria  staff        96 Apr  5 13:05 ..
-rw-r--r--  1 daria  staff       119 Apr  5 13:05 2024-04-05-13-05-27-013
-rw-r--r--  1 daria  staff       255 Apr  5 13:05 2024-04-05-13-05-27-013.meta
-rw-r--r--  1 daria  staff       146 Apr  5 13:05 pg_probackup.conf
```

PostgresPro

Полный backup в pg_probackup 3 PRO режим

```
daria@daria:~$ pg_probackup3 backup -d $BASE -B $BACKUPDIR3 --instance $INSTANCE --backup-id=1-full -b FULL --backup-source=pro


[2024-04-05 13:05:27.482794] [76948] [8000806592] [info] This PostgreSQL instance was initialized with data block checksums. Data block corruption will
    be detected
BACKUP COMMAND PG_PROBACKUP(LABEL '1-full', WAL true, VERIFY_CHECKSUMS true, COMPRESS_ALG 'none', COMPRESS_LVL 1);
INFO:  get connection from pg_probackup in backup mode
INFO:  Start backup
INFO:  pbckp begin backup
PG_PROBACKUP 100663336 tli=1
WARNING:  Writing temporary dir ./base/pgsql_tmp
PG_PROBACKUP-STOP 100663608 tli=1 bytes written=161867347 bytes compressed=161867347
[2024-04-05 13:05:29.033927] [76948] [8000806592] [info] Backup time 1552
INFO: Backup 1-full completed succcessfully.
```

PostgresPro

Полный backup в pg_probackup 3 PRO режим

```
daria@daria:~$ ls -la $BACKUPDIR3/backups/$INSTANCE

total 332808
drwxr-xr-x  7 daria  staff       224 Apr  5 13:05 .
drwxr-xr-x  3 daria  staff        96 Apr  5 13:05 ..
-rw-r--r--  1 daria  staff       162 Apr  5 13:05 1-full
-rw-r--r--  1 daria  staff       246 Apr  5 13:05 1-full.meta
-rw-r--r--  1 daria  staff       119 Apr  5 13:05 2024-04-05-13-05-27-013
-rw-r--r--  1 daria  staff       255 Apr  5 13:05 2024-04-05-13-05-27-013.meta
-rw-r--r--  1 daria  staff       146 Apr  5 13:05 pg_probackup.conf
```

PostgresPro

Полный backup в pg_probackup 3 Show

```
daria@daria:~$ pg_probackup3 show -B $BACKUPDIR3 --instance $INSTANCE

BACKUP INSTANCE 'test'
================================================================================================================
Instance Version ID                  Recovery Time Mode WAL Mode TLI Time                 Data  WAL Zalg Zratio Start LSN Stop LSN  Status
================================================================================================================
test     2024-04-05-13-05-27-013 ----         FULL STREAM    1   2024-04-05 13:05:27+0300 0     0   none       0/5000028 0/0       OK
test     1-full                  ----         FULL STREAM    1   2024-04-05 13:05:27+0300 154MB 0   none 1.00  0/6000028 0/6000138 OK
```

PostgresPro

libpgprobackup3 (SDK)

Все знания о формате и протоколе инкапсулируются в подгружаемую библиотеку:

Что делает:

- Написана на C++, но с интерфейсом на C
- Лёгкая интеграция в Go, C++ и Python
- Общается с Postgres Pro с помощью нашего протокола
- Формирует потоки архивных файлов и метаданных
- Реализует merge и restore
- Реализует алгоритмы Retention

Что не делает:

- Библиотека не знает про хранилище (файлы, лента, S3)
- Доступ к метаданным и данным реализован приложением
- Общение библиотеки с приложением-обёрткой на логическом уровне
- Не занимается чтением и записью файлов базы Эта часть (вместе со знанием о CFS) будет в исполняемых файлах Postgres Pro

PostgresPro

libpgprobackup3 (SDK) golang_sample

daria@daria:~$ golang_sample backup -pgport $SOURCEPORT -pgdata $BASE -backup-id $FULL -backup-mode FULL -backup-source=pro -storage='fs'

```
package pgpro

backupCmd.Parse(os.Args[2:])
  copt := pgpro.ConnectionOptions{PGDatabase:
     *pgdatabase_flag, PGHost: *pghost_flag, PGPort:
     *pgport_flag}
  bopt := pgpro.BackupOptions{
   NumThreads: *num_threads_flag,
   PGData: *pgdata_flag,
   BackupMode: backup_mode_flag,
   BackupSource: source_flag,
   Storage: storage_flag,
   BackupId: *backup_id_flag,
   ParentBackupId: *parent_backup_id_flag,
  do_backup(copt, bopt)
```

```
func do_backup(copt pgpro.ConnectionOptions, bopt
pgpro.BackupOptions, root string, tblspc string) {
 enc := &DummyEncoder{}
 enc.Root = root
 enc.DataFiles = make(map[int]*os.File)
 enc.NonDataFiles = make(map[int]*os.File)
 enc.TablespaceMapping = build_tablespaces(tblspc)
 enc.DataBlockSize = 0
 enc.WalDataBlockSize = 0

 res := pgpro.Backup(copt, bopt, enc)
 if res != nil {
  fmt.Println("Backup error: ", res)
 }
}
```

PostgresPro

Планы на будущее и Roadmap развития pg_probackup

Community Версия

- Исправление ошибок

pg_probackup2

- Выпускаем согласно релизному циклу Postgres Pro
- Исправление ошибок и небольшие доработки
- Может стать Community Версией

pg_probackup3

- Май 2024 – альфа-версия, для внутреннего тестирования
- Август 2024 – бета-версия, в том числе для тестирования с партнерами
- Декабрь 2024 – релиз

PostgresPro

Спасибо за внимание!

Вопросы