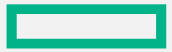


Hewlett Packard
Enterprise

PostgreSQL エラーが出ない話

Noriyoshi Shinoda

August 24, 2021

SPEAKER
篠田典良（しのだのりよし）

- 所属
  - 日本ヒューレット・パッカード株式会社
- 現在の業務
  - PostgreSQL をはじめ、Oracle Database, Microsoft SQL Server, Vertica 等 RDBMS 全般に関するシステムの設計、移行、チューニング、コンサルティング
  - Oracle ACE（2009 年 4 月～）
  - オープンソース製品に関する調査、検証
- PostgreSQL 14 に対して
  - psql コマンドに CREATE OR REPLACE TRIGGER 文のタブ補完（bf0aa7c4）
  - pg_stat_replication_slots カタログの列名変更（03d51b77）など
- 関連する URL
  - 「PostgreSQL 虎の巻」シリーズ
    - http://h30507.www3.hp.com/t5/user/viewprofilepage/user-id/838802
  - Oracle ACE ってどんな人？
    - http://www.oracle.com/technetwork/jp/database/articles/vivadeveloper/index-1838335-ja.html

SPEAKER
篠田典良（しのだのりよし）

- PostgreSQL Unconference #15
  - 2020年7月30日
  - 検知できない破壊の話
- PostgreSQL Unconference #20
  - 2021年2月2日
  - プロセス障害の話
- PostgreSQL Unconference #26
  - 2021年8月24日
  - エラーが出ない話

- スライドはこちら
  - https://www.slideshare.net/noriyoshishinoda

Huge Pages
Huge Pages とは？

- Linux における複数サイズのメモリー・ページを管理する仕組み
- 通常 4KB のページで管理する領域以外に 2MB ページ（デフォルト）の領域を追加
- Huge Pages を意識させない Transparent Huge Pages 機能もあるが DBMS サーバには非推奨
- カーネル・パラメーター vm.nr_hugepages にページ数を指定（デフォルト 0）
- 参考：
  - Huge Page とは何ですか? これを使用する利点は?
    - https://access.redhat.com/ja/solutions/293173
  - Tuning Red Hat Enterprise Linux Family for PostgreSQL
    - https://www.enterprisedb.com/blog/tuning-red-hat-enterprise-linux-family-postgresql

Huge Pages
MySQL では？

- 設定
  - PostgreSQL の huge_pages = try に近い動作

```
# cat /etc/my.cnf
[mysqld]
large-pages
```

- 起動ログ
  - Huge Pages 領域が確保できないので通常メモリーを使用するログが出力される

```
[System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.24) starting as process 116322
[System] [MY-013576] [InnoDB] InnoDB initialization has started.
[Warning] [MY-012677] [InnoDB] Failed to allocate 138412032 bytes. errno 1
[Warning] [MY-012679] [InnoDB] Using conventional memory pool
[System] [MY-013577] [InnoDB] InnoDB initialization has ended.
```

Huge Pages
Oracle Database では？

- 設定
  - PostgreSQL の huge_pages = try に近い動作

```
SQL> SHOW PARAMETER use_large_pages
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
use_large_pages                      string      TRUE
```

- 起動ログ
  - Huge Pages 領域が確保できない場合は、確保できる部分のみ Huge Pages を使用するとログが出力される

```
Supported system pagesize(s):
PAGESIZE  AVAILABLE_PAGES  EXPECTED_PAGES  ALLOCATED_PAGES  ERROR(s)
      4K       Configured               4           309127  NONE
   2048K              600            1200              597  NONE
RECOMMENDATION:
1. For optimal performance, configure system with expected number of pages for every supported system
pagesize prior to the next instance restart operation.
```

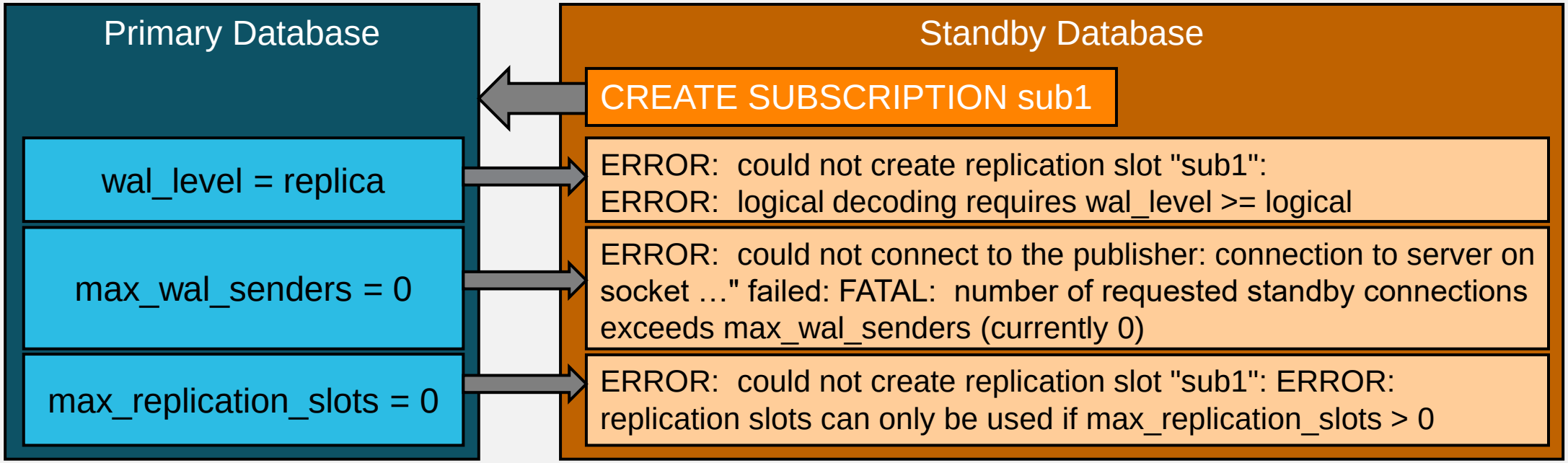
Huge Pages
PostgreSQL では？

- GUC huge_pages = try がデフォルト
  - Huge Pages 領域を確保しようとするが、必要な領域が不足した場合は Huge Pages を一切使わない
- 成功も失敗もログには何も出力されないので /proc/meminfo とかで確認する必要がある
- 確保しようとした共有メモリー量は「log_min_messages = DEBUG3」にしないと出力されない
- ログ出力例

```
DEBUG:  invoking IpcMemoryCreate(size=148324352)
DEBUG:   mmap(148897792) with MAP_HUGETLB failed, huge pages disabled: Cannot allocate memory
```

Logical Replication
必要なリソース

– 多くの機能が関係する
  – WAL にロジカル・レプリケーションに必要な情報を付与（wal_level = logical）
  – WAL Sender プロセスの使用（max_wal_senders > 0）
  – Replication Slot の使用（max_replication_slots > 0）



Logical Replication
必要なリソース

- 現状ではプライマリ・インスタンスのリソース不足時 CREATE SUBSCRIPTION文は成功する
- プライマリ・インスタンスでは以下のエラー・ログが定期的に出力される
  - max_wal_senders 不足時のプライマリ・インスタンスのエラー

```
FATAL:  number of requested standby connections exceeds max_wal_senders (currently 1)
```

  - max_replication_slots 不足時のプライマリ・インスタンスのエラー

```
ERROR:  all replication slots are in use
HINT:  Free one or increase max_replication_slots.
STATEMENT:  CREATE_REPLICATION_SLOT "pg_32786_sync_32778_6997334902875904787" LOGICAL
pgoutput USE_SNAPSHOT
ERROR:  replication slot "pg_32786_sync_32778_6997334902875904787" does not exist
STATEMENT:  DROP_REPLICATION_SLOT pg_32786_sync_32778_6997334902875904787 WAIT
```

Logical Replication
必要なリソース

- PostgreSQL 14 における変更
  - PostgreSQL 13 では、初期データ移行と差分更新が同一のトランザクションで実施されていた
  - PostgreSQL 14 ではそれぞれ独立したトランザクションで実行される（Commit Hash: ce0fdbfe）
  - それぞれレプリケーション・スロットと WAL Sender が割り当てられるため、一時的に2倍のリソースが必要
- 現状では一度リソース不足のエラー・メッセージが出力されると、SUBSCRIPTION を削除するまでエラーが解消されない？

コマンド・パラメーター
数値型のパラメーターに文字列を指定

- エラーが発生するコマンドは？

```
$ pg_basebackup -D data.bck --compress=ABC
$ pg_ctl --wait --timeout=DEF start
$ pg_ctl kill TERM GHI
$ pg_dump --compress=JKL --extra-float-digits=MNO
$ pg_dumpall --extra-float-digits=PQR
$ pg_receivewal -D data.rcv --compress=STU --status-interval=VWX
$ pg_recvlogical --fsync-interval=YZA --status-interval=BCD
$ pgbench pgbench --initialize --partitions=EFG
$ vacuumdb --parallel=HIJ
```

- 全部動作します。
  - PostgreSQL 14 Beta 3 まで

コマンド・パラメーター
数値型のパラメーターに文字列を指定

- ソースコード

```
case 'Z': /* Compression Level */
   compressLevel = atoi(optarg);
   if (compressLevel < 0 || compressLevel > 9)
   {
      pg_log_error("compression level must be in range 0..9");
      ...
```

- PostgreSQL 15dev では改善
  - 2021/7/24: Unify parsing logic for command-line integer options / Commit Hash: b859d94c で修正

```
case 'Z': /* Compression Level */
   if (!option_parse_int(optarg, "-Z/--compress", 0, 9,
                         &compressLevel))
       exit_nicely(1);
      ...
```

UNLOGGED TABLE
クラッシュ・リカバリ中のデータ削除

– マニュアル（CREATE TABLE）

> クラッシュまたは異常停止の後、ログを取らないテーブルは自動的に切り詰められます。

– クラッシュ・リカバリ中のログ

```
LOG:  listening on IPv4 address "127.0.0.1", port 5432
LOG:  listening on Unix socket "/tmp/.s.PGSQL.5432"
LOG:  database system was interrupted; last known up at 2021-08-23 12:54:55 JST
LOG:  database system was not properly shut down; automatic recovery in progress
LOG:  redo starts at 0/96F8E68
invalid record length at 0/FB7E220: wanted 24, got 0
LOG:  redo done at 0/FB7E1B8 system usage: CPU: user: 0.21 s, system: 0.04 s, elapsed: 0.25 s
LOG:  database system is ready to accept connections
```

– 「log_min_messages = DEBUG1」設定時のログ

```
DEBUG:  resetting unlogged relations: cleanup 0 init 1
```

予告
篠田の虎の巻

- Citus 10 の検証資料を作成中
  - Columnar Table
  - Shard Rebalancer
  - Etc.
- Azure Database for PostgreSQL – Hyperscale (Citus) now GA
  - https://azure.microsoft.com/en-us/updates/azure-database-for-postgresql-hyperscale-citus-columnar-compression-now-generally-available/
- 9月前半には公開予定

THANK YOU

Mail: noriyoshi.shinoda@hpe.com
Twitter: @nori_shinoda