

## 目次

目次と本文の見出しの文が一致していませんでした。

### 2章 内部構造

PostgreSQL の構造.....	3
プロセスとデータベースクラスタ.....	5
データベース接続処理.....	6
SQL 実行処理.....	7
ストレージ上の物理配置.....	8
データの読み取り.....	11
データの書き込み.....	13

### 3章 デプロイメント

システム要件の策定.....	3
PostgreSQL の特徴.....	4
[Column] プロジェクトの変遷.....	6
システム構成の設計.....	7
複数台構成 (クラスタ).....	7
HA クラスタの実現方式.....	8
要件を実現する <b>指針方針</b> .....	9
ディスク容量の見積もり対象.....	10
データベースデプロイメント.....	11
PostgreSQL のインストール.....	11
データベースクラスタの作成.....	12
テーブルスペース.....	13
フィルファクタ.....	14
バージョンアップ.....	15
パフォーマンスの事前検証.....	16
pgbench とは.....	18

### 5章 メンテナンス

ログ.....	3
[Column] syslog による書き込み.....	8
バックアップ.....	9
論理バックアップ.....	10
[Column] リストアの高速化.....	13
物理バックアップ.....	14
[Column] pg_basebackup 設定.....	16
Point In Time Recovery.....	18
[Column] PITR の設定例.....	20
[Column] トランザクションログの強制アーカイブ.....	22
VACUUM.....	23
統計情報 ( <b>ANALYZE</b> ).....	27
インデックスの再構築 ( <b>REINDEX</b> ).....	30
ディスク領域の管理.....	33
[Column] pgstattuple.....	35
[Column] システムカウンタの監視.....	36

### 7章 性能分析

性能分析とは.....	3
ボトルネックの原因.....	4
システム監視ツールによる調査.....	5
vmstat の実行結果.....	6
ログからの調査.....	7
ロック状況の調査.....	12

アクセス統計情報.....	14
[Column] 統計情報ビュー <b>pg_stat_archiver</b> .....	20
実行計画.....	22
EXPLAIN.....	23
EXPLAIN ANALYZE.....	26
[Column] バッファの使用状況.....	28
計画タイプ.....	29
スキャン方式.....	29
結合方式.....	32
集約方式.....	35
[Column] その他の計画タイプ.....	36
テーブル/カラム統計情報.....	37
[Column] プランナの得意な処理と苦手な処理.....	43
[Column] 拡張統計情報.....	44

## 8章 性能チューニング

VACUUM のチューニング.....	4
メンテナンスワークメモリの設定.....	5
VACUUM の遅延実行.....	6
自動 VACUUM の設定.....	7
読み書きのチューニング.....	9
共有メモリバッファの設定.....	10
同期書き込み方式の設定.....	11
WAL バッファの設定.....	12
同期書き込み <b>の設定</b> .....	13
同期コミットの設定.....	14
遅延コミットの設定.....	15
チェックポイントの設定.....	16
[Column] バックグラウンドライタの設定.....	17
不要なインデックスの削除.....	18
データロードの高速化.....	19
実行計画のチューニング.....	20
ANALYZE の実行.....	21
インデックスの追加.....	22
ワークメモリの設定.....	24
プランナコスト定数の設定.....	25
計画タイプの無効化.....	26
テーブルの結合順序.....	27
その他のチューニング.....	28
ハードウェアの増強.....	28
デッドロック検出処理の抑制.....	29
コネクションプーリング.....	30
テーブルのパーティショニング.....	31
宣言的パーティショニング.....	32
パラレルクエリ.....	33
【演習】性能チューニング.....	34

## 11章 付録

<b>PostgreSQL</b> のインストール.....	3
ソースコードからのインストール.....	5
RPM パッケージによるインストール.....	8
Windows 版パッケージのインストール.....	11
環境変数について.....	13
OSS-DB Exam Gold 試験範囲.....	14
参考 URL.....	18

### 3章 デプロイメント

- 18 ページ 「パフォーマンスの事前検証 (3) pgbench とは」
  - 誤: Pgbench\_history  
正: pgbench\_history

### 4章 セキュリティ管理

- 9 ページ 「クライアント認証 (2) 設定例 1」
  - 誤: Hostssl  
正: hostssl
- 19 ページ 「4章 演習 (2) セキュリティ設定」
  - 誤: 1. ex1.sql をエラーなく実行するためのアクセス権限をロール ex1 に与えてください  
正: 2. ex1.sql をエラーなく実行するためのアクセス権限をロール ex1 に与えてください

### 5章 メンテナンス

- 4 ページ 「ログ (2)」
  - 誤: #log\_destination = 'stderr' (デフォルト)  
他には 'syslog','csvlog','eventlog' (Windows)  
正: #log\_destination = 'stderr' (デフォルト)  
他には 'syslog','csvlog','eventlog' (Windows)
  - 誤: ・複雑ホストのログを集約することはできない  
正: ・複数ホストのログを集約することはできない
- 14 ページ 「物理バックアップ (1) オフラインバックアップ」
  - 誤: Tar や cp,  
正: tar や cp,

## 6章 レプリケーション

- 11 ページ 「ストリーミングレプリケーション (6) スタンバイの昇格」 補足
  - 誤: `pg_rewind` を使用するには、事前に `wal_log_hints` を `on` に設定して WAL にヒントビットを書き込むようにしておくか、**i** データベースクラスタの作成時にデータチェックサムを有効にしておく必要があります。
  - 正: `pg_rewind` を使用するには、事前に `wal_log_hints` を `on` に設定して WAL にヒントビットを書き込むようにしておくか、データベースクラスタの作成時にデータチェックサムを有効にしておく必要があります。

## 7章 性能分析

- 13 ページ 「ロック状況の調査 (2) ロックの競合」

- 誤: ACCESS SHARE / EXCLUSIVE でロック待ちは発生しません。

	ACCESS SHARE	ROW SHARE	ROW EXCL.	SHARE UPDATE EXCL.	SHARE	SHARE ROW EXCL.	EXCL.	ACCESS EXCL.
弱 ↑ ロックモード	ACCESS SHARE						●	●
	ROW SHARE						●	●
	ROW EXCL.				●	●	●	●
	SHARE UPDATE EXCL.			●	●	●	●	●
	SHARE		●	●		●	●	●
	SHARE ROW EXCL.		●	●	●	●	●	●
	EXCL.	●	●	●	●	●	●	●
↓ 強	ACCESS EXCL.	●	●	●	●	●	●	●

● ロックの競合が発生 EXCL. = EXCLUSIVE

正:

	ACCESS SHARE	ROW SHARE	ROW EXCL.	SHARE UPDATE EXCL.	SHARE	SHARE ROW EXCL.	EXCL.	ACCESS EXCL.
弱 ↑ ロックモード	ACCESS SHARE							●
	ROW SHARE						●	●
	ROW EXCL.				●	●	●	●
	SHARE UPDATE EXCL.			●	●	●	●	●
	SHARE		●	●		●	●	●
	SHARE ROW EXCL.		●	●	●	●	●	●
	EXCL.	●	●	●	●	●	●	●
↓ 強	ACCESS EXCL.	●	●	●	●	●	●	●

● ロックの競合が発生 EXCL. = EXCLUSIVE

- 20 ページ 「Column」

- 誤: Point!  
pg\_stat\_archiver
- 正: 統計情報ビュー  
Point!  
pg\_stat\_archiver  
(コラムの見出しが抜けていました)

## 9 章 障害対応

- 5 ページ 「起動できない (2) リソース不足/アクセス不可」
  - 誤: 起動できない (2) リソース不足/アクセス不足
  - 正: 起動できない (2) リソース不足/アクセス不可
- 9 ページ 「ハングアップした (2) 原因の調査方法」
  - 誤: **• gdb**  
(gdb のインデントレベルが適切ではありませんでした。gdb は「OS ツールで調査」に含まれるべきでした)
  - 正: **• gdb**

## 10 章 演習回答

- 4、5 ページ 「4 章 演習解答 (2) セキュリティ設定」
  - 誤: GRANT 文がエラーの発生順序と一致していませんでした。  
mb\_tag の SELECT 権限は不要でした。  
mb\_comment\_cid\_seq に付与するのは UPDATE 権限でした。
  - 正: 

```
SELECT bid, substr(message, 1, 50) FROM miniblog ORDER BY ts DESC LIMIT 10;  
→ GRANT SELECT ON miniblog TO ex1;  
  
UPDATE mb_tag SET tags[5] = 'aaa' WHERE bid = 123;  
→ GRANT UPDATE (tags) ON mb_tag TO ex1;  
  
SELECT bid, substr(message, 1, 50) FROM miniblog WHERE bid  
IN (SELECT bid FROM mb_tag WHERE tags && ARRAY['aaa']) ORDER BY ts  
LIMIT 10;  
SELECT * FROM miniblog WHERE bid = 123;  
SELECT * FROM mb_tag WHERE bid = 123;  
  
INSERT INTO mb_comment (bid, message, uid) VALUES (123, 'good job', 45);  
→ GRANT INSERT ON mb_comment TO ex1;  
→ GRANT UPDATE ON mb_comment_cid_seq TO ex1;  
SELECT * FROM mb_comment WHERE bid = 123 ORDER BY ts DESC LIMIT 10;  
→ GRANT SELECT ON mb_comment TO ex1;
```