

## OSS-DB Silver 技術解説セミナー

2022/12/04 開催

主題	運用管理
副題	標準付属ツールの使い方

本日の講師



株式会社デジタル・ヒュージ・テクノロジー  
滝澤 正大

**LPI-JAPAN**

## ■ 滝澤 正大

### • 株式会社デジタル・ヒュージ・テクノロジー

- モバイルアプリ開発
- OSSコンソーシアム OpenEDA部会 リーダ
- PostgreSQLとLaravel ハンズオンセミナー
- twitter: @wakawa\_13

## ■ 株式会社デジタル・ヒュージ・テクノロジー

- OSSシステムを活用したシステム構築
- サーバからフロントエンドまで
- LinuC, OSS-DB Silver/Gold, HTML5 講座（会社様向け）

<https://kusanagi.dht-jpn.co.jp>



#OSS-DB



## ■一緒に働いてくださる方募集中！



OSSを活用して回る、業務システムの職人集団です。



会社概要

ソリューション

課題別解決案

研修

採用情報

コラム

お問い合わせ



DHTは、お客様のニーズを  
ITパワーで支える  
縁の下の力持ちであり続けます。

DHTソリューション

詳細へ >



<https://kusanagi.dht-jpn.co.jp>

## ■ OSS-DBとは



オープンソースのデータベースソフトウェア「PostgreSQL」を扱うことができる技術力の認定です。様々な分野でPostgreSQLの利用拡大が進む中でOSS-DBの認定を持つことは、自分のキャリアのアピールにもつながります。

- ✓ OSS-DB Goldは設計やコンサルティングができる技術力の証明

PostgreSQLについての深い知識を持ち、データベースの設計や開発のほか、パフォーマンスチューニングやトラブルシューティングまで行えることが証明できます
- ✓ OSS-DB Silverは導入や運用ができる技術力の証明

PostgreSQLについての基本的な知識を持ち、データベースの運用管理が行えるエンジニアとしての証明ができます
- ✓ 対象のバージョンはPostgreSQL 11



## ■DBを使える人から使いこなす人へ

- 利用するだけの情報は溢れている  
「銀の弾はない」のように最適な設定や使い方は目的によって異なる
- 幅広い知識を有することで問題把握・解決する力がつく

## ■公式ドキュメントを読む

- 公式ドキュメントが1番信用できる情報
- PostgreSQLはバージョンによって利用できる機能が異なるため  
利用するバージョンと参照ドキュメントの対象バージョンが一致しているか確認

## ■実際に動かしてみる

- 実際に動かしてみることで、正しく理解できているか確認することができる



## ■ 今回のテーマ

- データベースとは何か知る
- 標準付属ツールの種類と使い方を知る

## ■ 過去のオンラインセミナー

- PostgreSQLのバックアップ方法  
<https://oss-db.jp/event/20200719>
- トランザクションの概念とSQLコマンド  
<https://oss-db.jp/event/20201017>
- 標準ツールの使い方およびSQLコマンド  
<https://oss-db.jp/event/20210410>
- 設定と運用  
<https://oss-db.jp/event/20210629>
- DBの環境設計・運用設計の基礎  
<https://oss-db.jp/event/20210725>



## ■データベースとは

- データの集まり
- データベース管理システム (DBMS) で制御
- PostgreSQLはデータベース管理システム

## ■データベース管理システムの種類

- 階層型データベース, ネットワーク型データベース, リレーショナル型データベース, . . .

## ■データベースが必要な理由

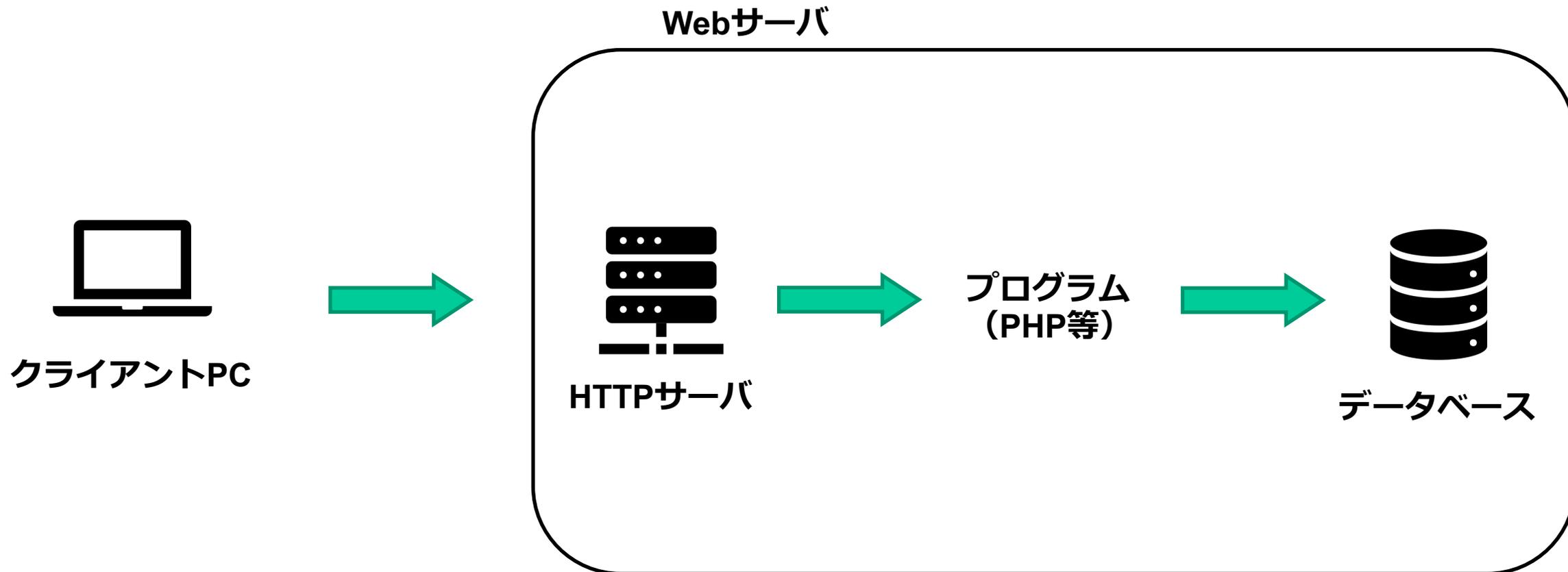
- 大量のデータを効率的に読み書きするため
- データの整合性を保つため

データベース管理システムはデータの保存という複雑な処理を安全に簡単にしてくれる

## ■Webサーバとデータベース



#OSS-DB





# 標準付属ツール

標準付属ツールの種類と使い方



## ■ pg\_ctl

### pg\_ctl サブコマンド (オプション)

- PostgreSQLの起動, 停止, 状態確認や制御等に使う
- PostgreSQLの管理ユーザで実行する必要がある

## ■ データベースクラスタの作成

### pg\_ctl initdb (オプション)

- **覚えておきたいオプション**
  - -D, --pgdata= : データベースクラスタを作成するディレクトリ
  - -o : initdbにわたすオプション



## ■ 起動

```
pg_ctl start (オプション)
```

### • 覚えておきたいオプション

- **-D, --pgdata=** : 起動対象のデータベースクラスタ
- **-w,, --wait** : 起動するまで待機する最大待ち時間  
最大待ち時間を超えた場合はバックグラウンドで起動処理は継続される

## ■ 停止

```
pg_ctl stop (オプション)
```

### • 覚えておきたいオプション

- **-D, --pgdata=** : 停止対象のデータベースクラスタ
- **-m, --mode=** : smart, fast, immediateから選択



## ■データベース停止時のモード

- オプションの指定方法はモード名か最初の一文字目で指定する

## ■smart

- すべてのクライアントからの切断, オンラインバックアップの完了を待つ
- 新しいクライアントからの接続は受け付けない

## ■fast

- すべてのクライアントからの接続を強制的に切断される
- 実行中のトランザクションはロールバックされる
- 停止モードを指定しない場合のデフォルトのモード

## ■immediate

- クリーンアップ処理を行わずにサーバプロセスを即座に中断する
- クラッシュしたときと同じ動作となる



## ■再起動

```
pg_ctl restart (オプション)
```

- 実質的な処理は"stop"してから"start"という処理
- 停止中の場合は起動する
- **覚えておきたいオプション**
  - -D, --pgdata= : 再起動の対象のデータベースクラスタ
  - -m, --mode= : smart, fast, immediate

## ■設定ファイルの再読み込み

```
pg_ctl reload (オプション)
```

- postgresql.confとpg\_hba.confの2つの設定ファイルの再読み込みを行う
- **覚えておきたいオプション**
  - -D, --pgdata= : 設定ファイル再読み込み対象のデータベースクラスタ

## ■ 状態確認



#OSS-DB

```
pg_ctl status (オプション)
```

## • 覚えておきたいオプション

- -D, --pgdata= : 状態確認の対象となるデータベースクラスタ

## ■ データベースを利用するためのステップ

- ① PostgreSQLのインストール
  - ② データベースクラスタの作成
  - ③ PostgreSQLの起動
  - ④ データベースユーザの作成
  - ⑤ データベースの作成
- データベースの利用



#OSS-DB



## ■ データベースユーザ

- OSのユーザと考え方は同じ
- OSのユーザ名とPostgreSQLのユーザ名は一致させる必要はない

## ■ データベースユーザの権限

- **CREATEROLE**権限：ユーザを作成する権限
- **CREATEDB**権限：データベースを作成できる権限
- **SUPERUSER**権限：スーパーユーザ権限
- **LOGIN**権限：データベースに接続できる権限



## ■ データベースユーザの作成

`createuser` (接続パラメータ) (オプション) (ユーザ名)

### • 覚えておきたいオプション

- **-p, --pwprompt** : パスワードの設定を行う
  - **-s, --superuser** : スーパーユーザ権限の設定をする
  - **-S, --no-superuser** : スーパーユーザの権限を設定しない (デフォルト)
  - **-d, --createdb** : データベースの作成権限を設定する
  - **-D, --no-createdb** : データベースの作成権限を設定しない (デフォルト)
  - **-r, --createrole** : ユーザの作成権限を設定する
  - **-R, --no-createrole** : ユーザの作成権限を設定しない (デフォルト)
  - **-l, --login** : ログイン権限を設定する (デフォルト)
  - **-L, --no-login** : ログイン権限を設定しない
- ユーザの作成ができるのはスーパーユーザもしくはユーザの作成権限があるユーザ
  - スーパーユーザの作成はスーパーユーザが行う必要がある



## ■ データベース

- データを格納する領域
- データベースユーザごとにアクセス権限を設定できる

## ■ データベースの作成

`createdb` (接続パラメータ) (オプション) (データベース名)

### • 覚えておきたいオプション

- `-E, --encoding=` : 新しいデータベースで使用されるエンコーディングを指定する
- `-l, --locale=` : 新しいデータベースで使用されるロケールを指定する
- `-O, --owner=` : 新しいデータベースの所有者となるデータベースユーザを指定する
- `-T, --template=` : データベース構築に使用するテンプレートデータベースを指定する

## ■作成したデータベースの削除



#OSS-DB

```
dropdb (接続パラメータ) (オプション) データベース名
```

- データベースの削除ができるのはスーパーユーザか削除対象のデータベース所有者
- 覚えておきたいオプション
  - **-i, --interactive** : 削除前に確認のためのプロンプトを表示する
  - **--if-exists** : 指定したデータベースが存在しない場合でもエラーにならない

間違えて意図しないデータベースを削除してしまわないように-iオプションはつけましょう



## ■クライアントアプリケーションとPostgreSQLの接続

- UNIXドメインソケット：同じマシンで動作するプロセス感で通信を行うIF
- TCP/IP：インターネットプロトコル

## ■覚えておきたい接続パラメータ

- **-h, --host=XXX**
  - ホスト名, IPアドレス
  - 省略時には環境変数PGHOSTが参照
  - 環境変数が設定されていない場合は, UNIXドメインソケット
- **-p, --port=XXX**
  - TCP/IP通信で用いられるポート番号
  - 省略時には環境変数PGPORTが参照
  - 環境変数が設定されていない場合には, 5432が使用
- **-U, --username=XXX**
  - 接続するデータベースユーザ名
  - 省略時には環境変数PGUSER
  - 環境変数が設定されていない場合には, コマンドを実行したOSのユーザ名が使用

## ■データベースへの接続



#OSS-DB

psql (接続パラメータ) (オプション) (データベース名 (ユーザ名))

### • 覚えておきたいオプション

- -f, --file= : ファイルからコマンドを読み取って実行する。標準入力からは読み取らない
- -c, --command= : 指定されたコマンド文字列を実行する。標準入力からは読み取らない
- -d, --dbname= : 接続するデータベースの名前を指定する

## ■実行例

```
$ psql -h 192.168.11.23 -p 5432 -U user db
psql (11.5)
Type "help" for help.
```

db=> ← 一般ユーザで操作

db=# ← スーパーユーザで操作



## ■メタコマンド

- クォーテーションで囲まれていないバックスラッシュから始まるものはメタコマンドとして解釈される

## ■覚えておきたいメタコマンド

- `\copy` : テーブルデータをコピーする
- `\d` : テーブル・ビュー・シーケンスの一覧を表示
- `\df` : 関数の一覧を表示
- `\dp` : テーブル単位の権限を表示
- `\ds` : シーケンスの一覧を表示
- `\dS` : システムカタログの一覧を表示
- `\dt` : テーブルの一覧を表示
- `\du` : データベースユーザの一覧を表示
- `\l (list)` : データベースの一覧を表示
- `\q (quit)` : psqlを終了

## ■ バッチ処理

- ファイルもしくははpsqlコマンド実行時の-cオプションでバッチ処理ができる



#OSS-DB

■ pg\_ctlコマンドを利用してできることとして、正しいものを3つ選びなさい  #OSS-DB

- A) 新しいデータベースクラスタ作成する
- B) データベースクラスタにデータベースを追加する
- C) データベースのユーザを作成する
- D) データベースサーバを起動・終了させる
- E) データベースの設定ファイルを更新した後、データベースサーバを停止させることなく、それを再読み込みさせる

## ■引用元

[https://oss-db.jp/sample/silver\\_management\\_06/114\\_210603](https://oss-db.jp/sample/silver_management_06/114_210603)

※この例題は実際のOSS-DB技術者認定試験とは異なります

■ pg\_ctlの使い方について述べたものから、誤っているものを2つ選びなさい  #OSS-DB

- A) pg\_ctl initdb とすることで、データベースクラスタを新しく作成できる
- B) pg\_ctl start とすることで、データベースサーバを起動できる
- C) pg\_ctl shutdown とすることで、データベースサーバを停止できる
- D) pg\_ctl reset とすることで、データベースサーバを再起動できる
- E) pg\_ctl status とすることで、データベースサーバの稼働状態が確認できる

## ■引用元

[https://oss-db.jp/sample/silver\\_management\\_02/35\\_131113](https://oss-db.jp/sample/silver_management_02/35_131113)

※この例題は実際のOSS-DB技術者認定試験とは異なります

## ■データベースの終了方法について適切なものを2つ選びなさい



#OSS-DB

- A) コマンドラインから `pg_ctl shutdown` を実行することによりデータベースを終了できる
- B) コマンドラインから `pg_ctl stop` を実行することによりデータベースを終了できる
- C) `psql` でデータベースに接続し、`¥shutdown` メタコマンドを実行することにより、データベースを終了できる
- D) 終了時にオプションで`smart`を指定すると、データベースに接続中のクライアントは自動的に切断され、実行中のトランザクションはロールバックされる
- E) データベースの終了は、データベースの管理者ユーザ(多くの環境では`postgres`)のみが実行可能で、OSの管理者ユーザ(Linuxの`root`)の権限では実行できない

## ■引用元

[https://oss-db.jp/sample/silver\\_management\\_02/24\\_121220](https://oss-db.jp/sample/silver_management_02/24_121220)

※この例題は実際のOSS-DB技術者認定試験とは異なります



## ■まとめ

- 本日は以下の内容を解説しました, 不安な箇所は復習や自分で調べてみましょう
  - データベースの役割
  - Webサーバ内におけるデータベースの位置
  - PostgreSQL サーバアプリケーション (pg\_ctl)
  - PostgreSQL クライアントアプリケーション (createuser, psql…)

## ■まずは公式ドキュメントを参照しましょう

## ■過去のオンラインセミナーも見てみてください