

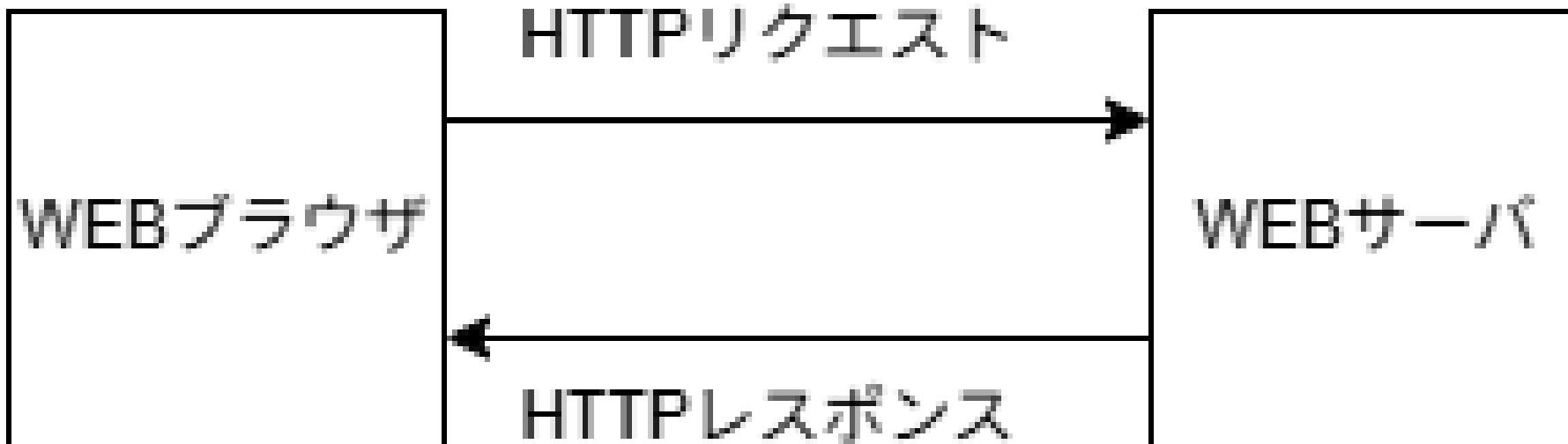
# WEB システムの仕組み

システム構成とアーキテクチャの理解

# 今日のお話

- 時間: 約 10 分
- 内容: WEB システムの基本的な仕組み
- 目標: システム構成をざっくり理解する

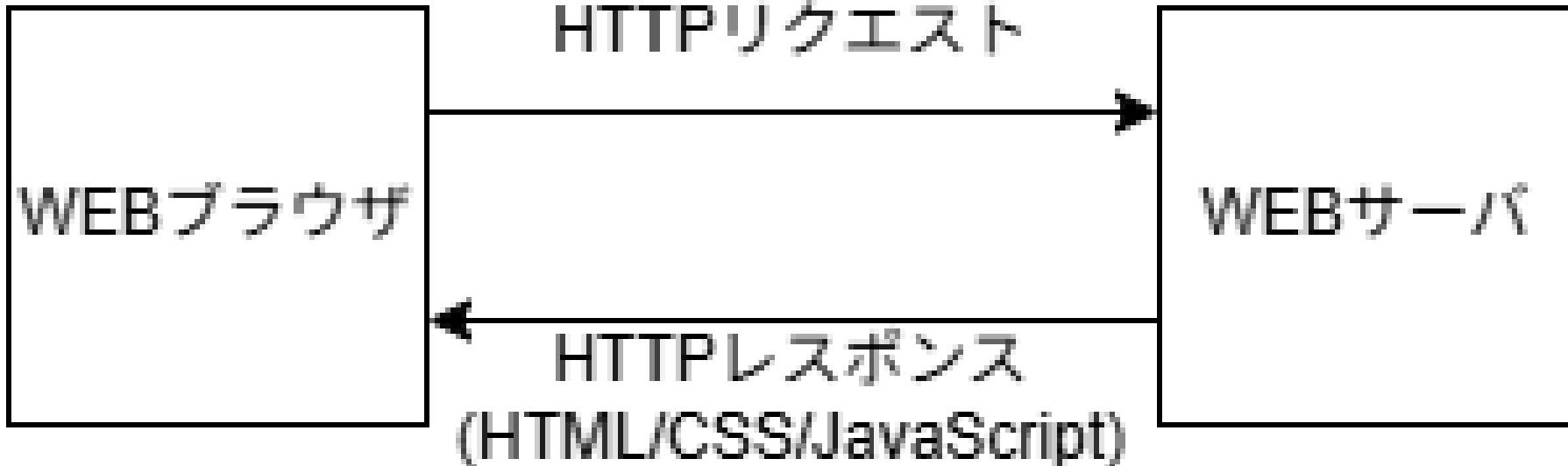
# 静的 WEB サイト



例：阿部寛のホームページ

- 事前に作成された HTML ファイルをそのまま返す

# 静的 WEB サイト（動作付き）



例 : (<https://Shingotti.github.io/MyBlog/>)

- 動きはあるが、WEB サーバ側では処理を行わない
- そのため静的 WEB サイト扱い
- JavaScript はブラウザ側で実行される

# 動的 WEB サイト (WEB アプリケーション)



- アプリケーションサーバと WEB サーバで役割をわける
- 1 台の高性能な PC でアプリケーションサーバを動かす

# WEB サーバの役割

## 静的コンテンツの配信

- HTML、CSS、JavaScript
- 画像ファイル (PNG、JPG、SVG)
- その他のリソース

## HTTP 通信の管理

- HTTP リクエストの受信
- HTTP レスポンスの送信

## プロキシ

- アプリケーションサーバへブラウザの代わりにリクエスト転送

# アプリケーションサーバの役割

## ⚡ 動的コンテンツの生成

- プログラムを実行して HTML を動的に作成

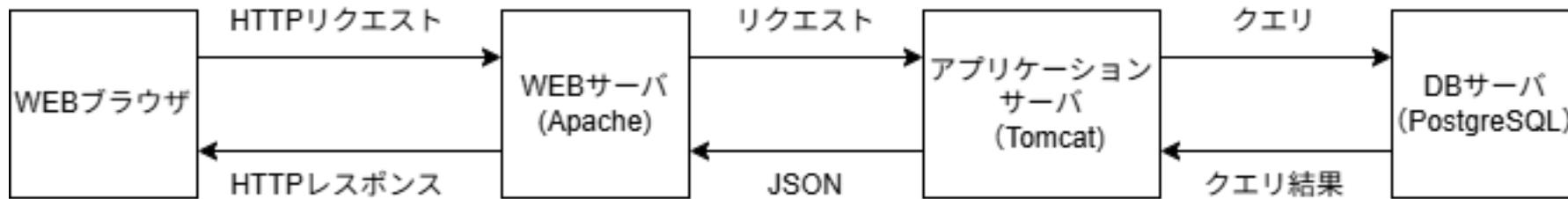
## 💼 ビジネスロジックの実行

- 業務処理の実装
- 認証・認可（ログイン・セッション管理）

## MDB データベース接続・操作

- CRUD 操作（Create, Read, Update, Delete）
- トランザクション管理

# SPA (Single Page Application)



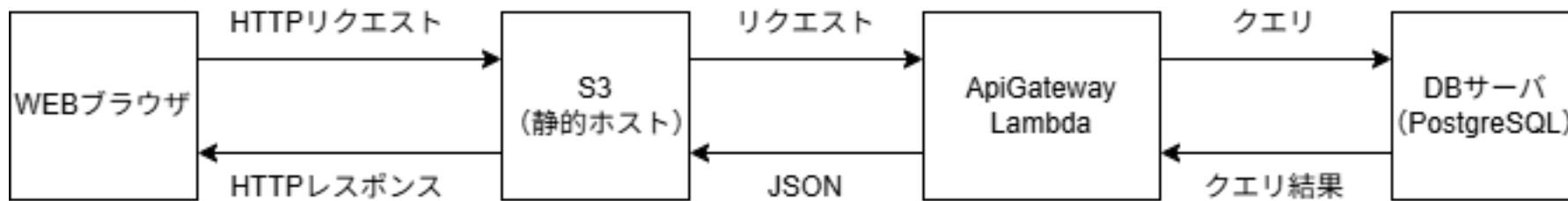
## SPA の特徴

- 初回のみ HTML 読み込み：最初にページ全体を取得
- JSON データ通信：以降は JSON でデータのやり取り HTTP まるごとは送らない
- クライアントサイドレンダリング：JavaScript でページを動的に更新

## ⌚ 従来の WEB アプリケーションとの違い

従来型	SPA
毎回 HTML を取得	初回のみ HTML 取得
サーバサイドで HTML 生成	クライアントサイドで HTML 生成
ページ遷移時にリロード	ページ遷移時もリロードなし

## サーバレスな構成 (AWS)



- アプリケーションサーバの代わりに、Lambda 関数が動く
- クラウドが発展したことにより、サーバーを意識しなくてよくなつた
- API 単位で疎結合な開発が可能

# まとめ

## 今日話したこと

1. 静的サイト 動的サイト SPA の基本構成



## 参考書籍・資料

- 「[改訂新版] プロになるための Web 技術入門」
  - おすすめポイント: ハンズオンが豊富 TODO アプリを作りながら WEB アプリの作り方が学べる
- 「イラスト図解式 この一冊で全部わかる Web 技術の基本」
  - おすすめポイント: 図解が豊富で初心者にも分かりやすい

ご清聴ありがとうございました！ 