

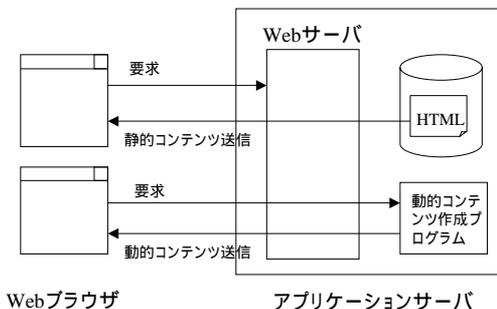
第4講 Servlet (その1)

- 静的コンテンツと動的コンテンツ
- Servletとは
- Servletの基本動作とServletコンテナ
- Javaアプリケーション、アプレット、Servletの比較
- Servletのライフサイクル
- Servletの特徴

静的コンテンツと動的コンテンツ

- 静的コンテンツ
 - Webサーバに保存されたHTMLファイルや画像ファイル
 - Webサーバは、ファイル取得要求が来ると、上記のファイルをWebブラウザに送信
- 動的コンテンツ
 - サーバ上である条件により自動的に作成されるコンテンツ
 - Webブラウザのファイル取得要求が来ると、Webサーバの裏に控えている動的コンテンツ作成プログラムが、さまざまな条件によって異なるHTMLファイルや画像ファイルを作成し、送信する
- 両者ともに、WebブラウザではWebサーバから送られてきたコンテンツが静的か動的かは判断できない

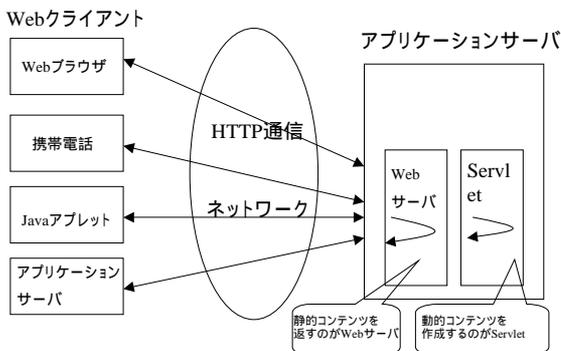
静的コンテンツと動的コンテンツ



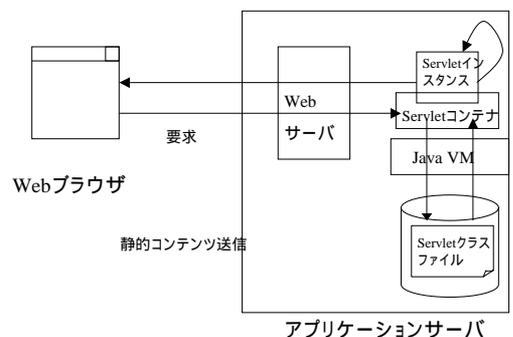
Servletとは

- Servlet
 - 動的なコンテンツを作成するサーバ上のJavaプログラム
 - さまざまなWebクライアントからの要求に応じて、処理をおこない、主に、HTMLを作成し結果として送信する
 - 入力である「要求」と出力である「応答」は、HTTP通信の要求と応答になる

Servletの役割



Servletの基本動作

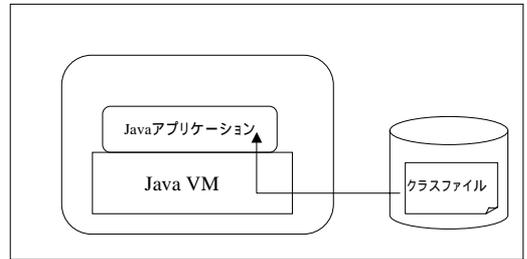


Servletコンテナ

- Java VM上で稼動するServlet実行環境
- Servletの利用者は、Servletコンテナを用意して、その上で稼動するServletプログラムを作成
- Servletを実行するだけでなく、Webクライアントからの要求 / 応答を取り扱い、セッション管理をおこなう
- Servletは、Servletコンテナから呼び出されるひとつのクラスであり、Servletコンテナ上でひとつのインスタンスとして実行される

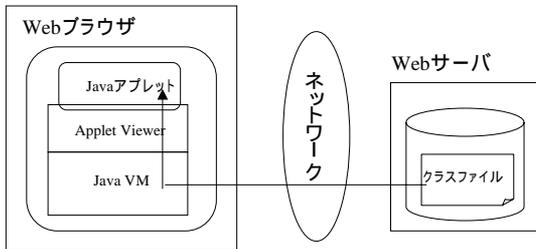
アプリケーション/アプレット/Servletの比較

- Javaアプリケーション
 - Java VM上で直接稼動



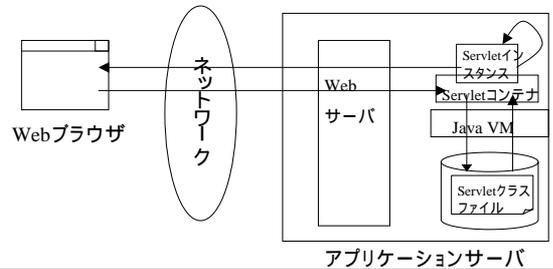
アプリケーション/アプレット/Servletの比較

- Javaアプレット
 - Java VM上で動くApplet Viewer上で稼動



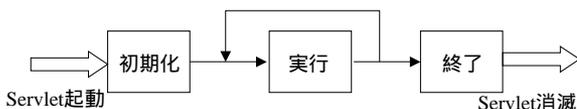
アプリケーション/アプレット/Servletの比較

- Servlet
 - Servletコンテナ上で稼動
 - ServletコンテナはひとつのJavaアプリケーション



Servletのライフサイクル

- Servletのライフサイクル
 - ServletがServletコンテナによって生成されてから消滅するまでの状態の移り変わり(状態遷移)
 - 外部の操作(ブラウザからの要求)によって、Servletの状態は変化していく



Servletのライフサイクル

- Servletの起動
 - はじめて、Servletへの要求がきたとき
 - Servletの起動をServletコンテナの起動時にロードするように設定している場合
- Servletの終了
 - Servletコンテナが終了したとき
 - 意図的にServletが終了するケース

Servletの特徴

- 初期化、終了を繰り返さない(応答処理の効率化)
 - CGI
 - ブラウザから要求されるたびに、そのプログラムを割り当てて、初期化、メインの処理、終了を実行し、終了後、割り当てていたメモリの解放をおこなうという実行形態(プロセス型)になっている
 - Servlet
 - サブレットコンテナそのものはプロセス型で実行されるが、個々のServletはサブレットコンテナ内のメモリを使って実行される
 - 一度実行したServletをコンテナ内のメモリに残しておき、同じServletを実行する

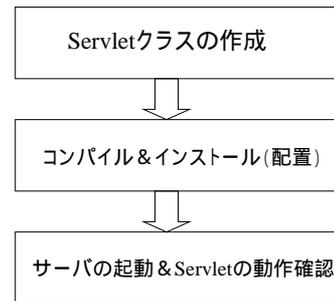
Servletの特徴

- 複数のリクエストを効率的に処理できる
 - 同じ種類のServletに対して、コンテナ内に複数の領域を確保して、同じServletを複数実行可能
 - 各Servletは短時間で制御が切り替えられ、次々と実行されるため、複数の要求が同時にあった場合でも、1つのサブレットの実行終了を待つのではなく、複数のServletを次々と制御を切り替えながら実行できる

Servletの特徴

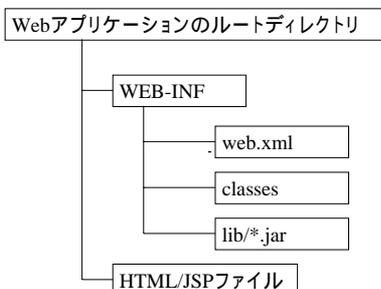
- Javaのメリットをそのまま享受できる
 - CGI
 - ネイティブコードを含んだバイナリであることが多く、プラットフォーム依存
 - Servlet
 - Javaで記述されているので、プラットフォーム非依存

Servletの作成



Servletクラスを格納するディレクトリ

- Webアプリケーションのディレクトリ構成



Servletの呼び出し方法

- Servletの呼び出し(URL)
 - `http://ホスト名:ポート番号/Webアプリケーション識別子/servlet/servlet名`
- HTMLファイルの呼び出し
 - `http://ホスト名:ポート番号/Webアプリケーション識別子/index.html`
- Servletの呼び出し
 - `Click Here`
 - `<FORM ACTION="/Webアプリケーション識別子/servlet/servlet名" METHOD=POST </FORM>`