

# Struts アプリケーションのパブリッククラウド (Google App Engine) への移行に対する考察

倉持 和彦<sup>†</sup> 原田 雅史<sup>†</sup>

三菱電機株式会社 情報技術総合研究所

## 1. はじめに

クラウド・コンピューティング(以降クラウド)とは、ネットワーク上に存在するサーバが提供するサービスを、サーバ群を意識することなく利用可能とするコンピューティングモデルを表す。クラウドは大きくパブリッククラウドとプライベートクラウドに分けられる。

パブリッククラウドは、クラウド提供者が巨大なデータセンターを持ち、コンピュータリソースを仮想化することで、不特定多数の企業や個人が利用することができるクラウドである。代表的なパブリッククラウド構築企業には、Amazon、Google、Force.com、Microsoft がある。中でも Google が提供しているクラウドサービスである Google App Engine for Java (以降 GAE/J) は文字通り Java 言語を用いてクラウド上に業務アプリケーションを構築できるため、従来 Apache Struts (以降 struts) を用いて構築した業務アプリケーションをクラウド環境に容易に移行できるのではないかと考えた。本稿は実際に Struts で構築した業務アプリケーションを GAE/J 上に移行した際の知見を述べる。

## 2. クラウド対応フレームワークの現状

GAE/J に移行するに当たりクラウド対応フレームワークを調査した。クラウド対応フレームワークは、ハイブ曲線の黎明期にあたり、製品そのものが少なく、GAE/J をサポートするフレームワークは Slim3[2] (現状β版) 位しか存在しない。下図に Java 言語を対象としたサーバサイドの代表的な Web アプリケーションフレームワークの動向を示す。

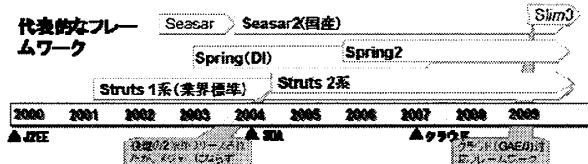


図 1 フレームワークの動向

今後、黎明期から流行期に移行すれば、オープンソースソフトウェアを含め数多くの製品がリリースされるものと思われる。

## 3. Slim3 とは

Slim3 は、GAE/J 上で動作するフル スタック型のアプリケーションフレームワークである。Seaser2 を開発した、ひがやすを氏が開発している。Slim3 のソフトウェアアーキテクチャを以下に示す。

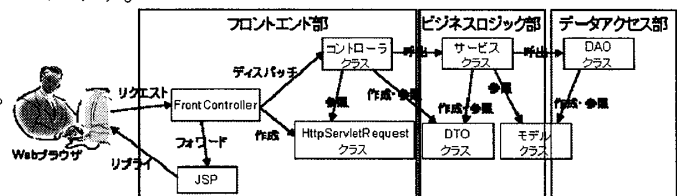


図 2 Slim3 ソフトウェアアーキテクチャ

この構造は、Struts を用いた場合の業務アプリケーションのソフトウェアアーキテクチャと同様の構造をとっているため、移行も容易に行えると思われる。Struts を用いた場合の代表的なソフトウェアアーキテクチャを以下に示す。

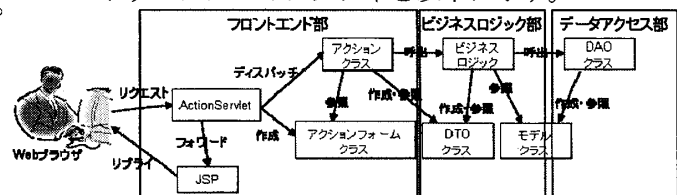


図 3 Struts ソフトウェアアーキテクチャ

## 4. Struts アプリケーションの移行

Struts を用いて作成したサンプル Web アプリケーションを、Slim3 を用いた GAE/J 上のアプリケーションとして移行を試みた。サンプルアプリケーションは展示ルームの見学スケジュールを管理するアプリケーションで、画面数 7、ボタン数 17 のアプリケーションである。以下にアプリケーションの画面例を示す。

Consideration to switch over struts application to public cloud

<sup>†</sup>Kazuhiko Kuramochi, Masahumi Harada

Mitsubishi Electric Corporation, Information Technology R&D Center



図 4 サンプルアプリケーション画面例

Slim3 のアーキテクチャに合わせて、各部の修正箇所を示す。

#### 4.1. フロントエンド部

Slim3 はフロントエンド部に Struts を用いず、独自の機構を用いている。そのため、Slim3 の構造に合わせて修正が必要となる。具体的には以下の修正を実施する。

- JSP ファイルの変更  
Struts 提供カスタムタグから、JSTL (JavaServer Pages Standard Tag Library) へ移行する。
- アクションクラスからコントローラクラスへの変更  
Slim3 独自の ant タスクによりコントローラクラスを生成し、アクションクラスの execute メソッドで実装しているロジックをコントローラクラスの run メソッドへ移行する。なお、ロジックにフォームクラスから値を参照しているコードが含まれる場合は HttpServletRequest からパラメータを直接取得するように変更する必要がある。

#### 4.2. ビジネスロジック部

ビジネスロジッククラスをサービスクラスへ変更する。ビジネスロジック部はフレームワークに依存しないように POJO (Plain old java object) で作成していたため、今回の移行ではビジネスロジックの Java クラスをそのままサービスクラスとして利用可能であった。

#### 4.3. データアクセス部

従来、リレーショナルデータベースの SQL 文で検索/登録/更新/削除を行っているコードを、GAE/J では、BigTable と呼ばれるキー・バリュー型データストアに対する検索/登録/更新/削除に変更する必要がある。Slim3 では、データストアに対する薄いラッパーを提供している。ただし、キー・バリュー型データストアの検索は、簡単な検索しか行えないため、その点を考慮した検索になるよう設計する必要がある。例えば、キー・バリュー型データストアの検索は複数のカラムを検索条件に指定することができない。また、テーブルのジョインを行うことができない。実際、サンプルアプリケーションにおいて検索条件に like 句を利用しているコードがあり、カラムの範囲条件への手直しが発生した。

#### 5. コード量による比較

Struts を利用した Web アプリケーションと Slim3 を利用した Web アプリケーションのコード量を比較する。

	Struts	Slim3
JSP および設定ファイル	484	387
ソースコード	1680	1661

JSP および設定ファイルは、Struts 設定ファイルのコード量が多いため Struts アプリケーションの方のコード量が多い結果となった。JSP 自体のコード量はほぼ同等である。また、ソースコードについては、ほぼ同等の結果となった。

#### 6. おわりに

本稿では Struts を用いた Web アプリケーションを Slim3 を用いたパブリッククラウド GAE/J に移行を行った。フロントエンド部は技術的に大きな課題はなく、定型的な作業で移行が可能である。ビジネスロジック部は特定のフレームワークに依存しない POJO で開発していれば移行は容易であることが確認できた。しかし、データアクセス部については、アーキテクチャが異なるため、DAO クラスの設計の変更が必要であることを確認した。

#### 参考文献

- [1] 中田秀基, Google のクラウド技術、情報処理 2009 Vol. 50, No11 P1062-1067
- [2] Slim3 プロジェクトサイト、  
<http://sites.google.com/site/slim3appengine>