2006年Javaキーワード解説

～ Webアプリケーション開発の最新動向 ～

WINGSプロジェクト

佐藤 匡剛 <sato@ouobpo.org>

なにを話すの？

- 最新技術をめぐってはさまざまなキーワードが飛び交うが、その本質を理解していないと、単に buzz word に踊らされるだけになる。
- 本プレゼンでは、JavaでWebアプリケーションを開発する上で知っておくべき 9つのキーワード を解説する。

Ⅰ．Java開発はEoD、軽量化の方向へ

ここで解説するキーワード：

① EoD

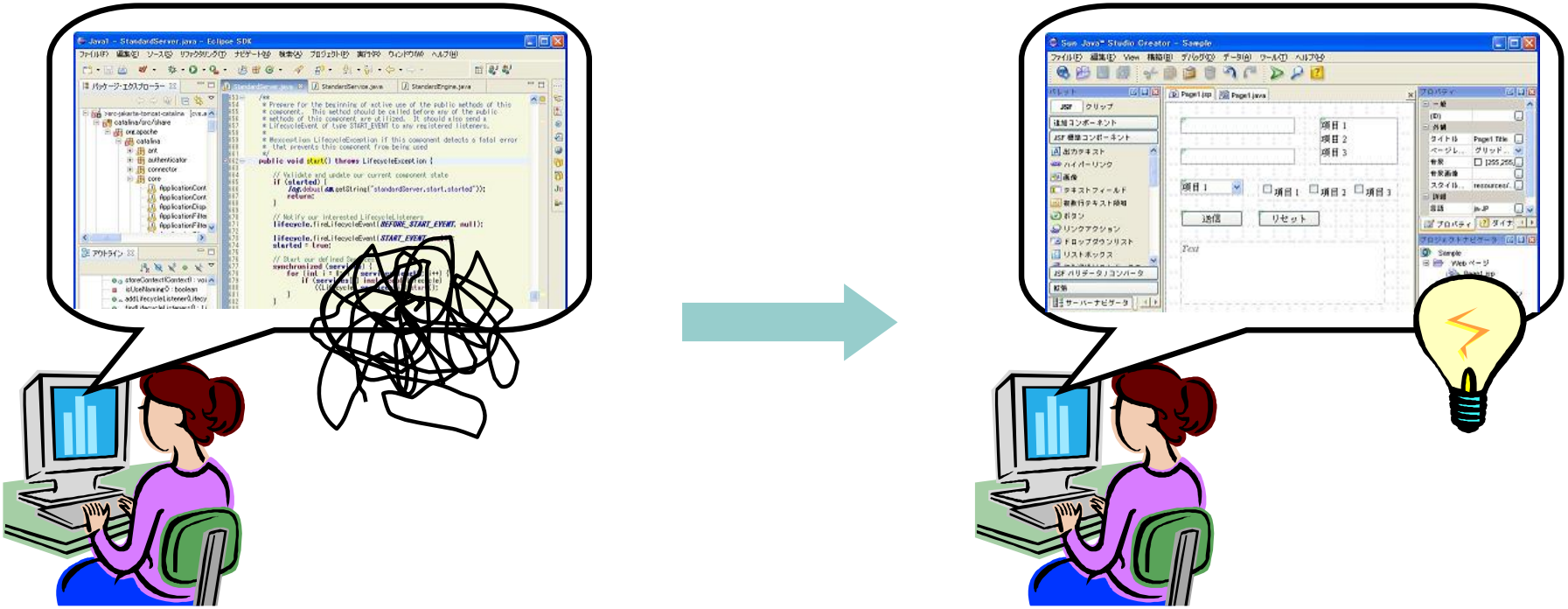
② JSF

③ EJB 3.0

④ DIコンテナ

EoD
（Ease of Development）

- 一言でいうと・・・
  - Javaの開発をもっと楽に、というスローガン



- Javaの特徴は、きちんとした設計で作りこめること
  - オブジェクト指向、セキュリティ、フレームワーク
  - 堅牢さの反面、手軽さに欠ける
    - PerlやPHPの方が、ちょっとしたプログラミングは楽
- Java 5以降、手軽に開発できるよう改善が進んでいる
  - Java言語構文の簡略化、オートボクシング
  - Javaベースのスクリプティング技術（Groovy, Jython）
  - 周辺技術もEoDの方向に
    - JSF、EJB 3.0、DIコンテナ

JSF
（Java Server Faces）

- 一言でいうと・・・
  - 初めて登場した標準のWebアプリケーションフレームワーク

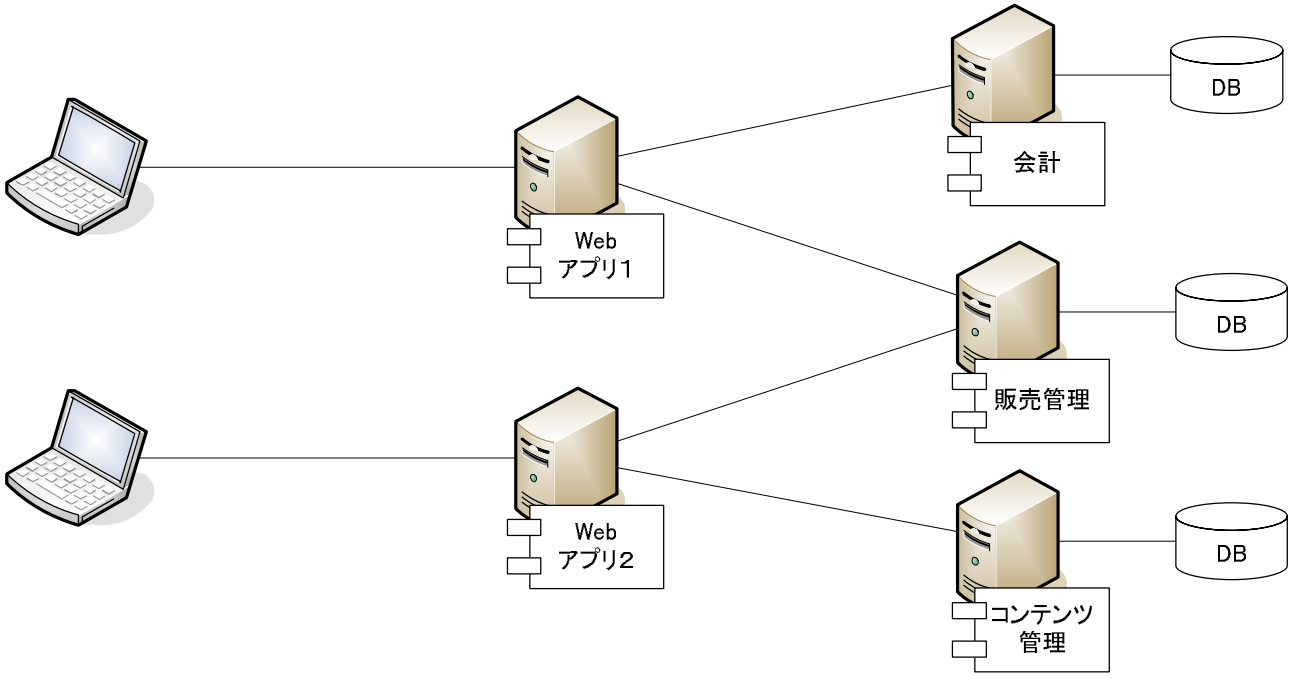
JSF準拠の統合開発環境
Sun Java Studio Creator

- Webアプリケーションフレームワークとは？
  - Webアプリケーションを作るための枠組み
    - Webアプリケーションに共通の基盤を提供
  - 品質の高いアプリケーションを構築できる
  - JSFのほか、Struts、Tapestry、Spring MVCなど

- JSFの特徴
  - 後発の新しいフレームワーク（2004年3月登場）
    - Strutsの反省を元に設計されている
  - サーバサイドJava（Java EE）標準であること
    - 今後、すべてのJava EEベンダにサポートされる
  - GUIベースの開発ができるよう設計されている
    - .NETを強く意識

EJB 3.0
（Enterprise JavaBeans 3.0）

- 一言でいうと・・・
  - 分散配置が必要な大規模アプリケーションのための標準技術



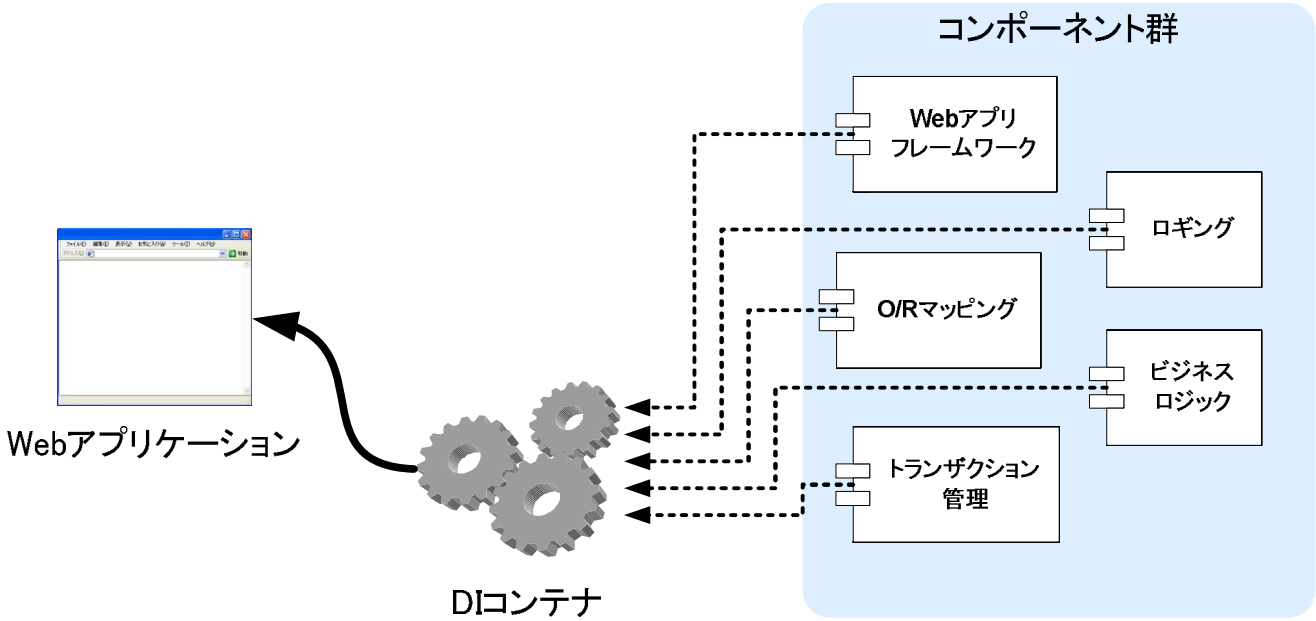
- EJBが実現する機能
  - 本格的なビジネスアプリケーションに必須の機能群
    - コンポーネントの分散配置、オブジェクトの永続化、自動トランザクション
  - Webアプリケーションフレームワークとは補完しあう
    - EJBは、ビジネスロジック部分を担当
- EJBの動作には、サーブレットコンテナのほかにコンテナが必要
  - JBoss、Apache Geronimo、IBM WebSphere、BEA WebLogic など
  - 基本的に、重量級のサーバサイド技術

- EJB 2.0までの問題点
  - EJBを書くのが非常に面倒
    - インタフェース実装などの規約、XMLの記述
  - テストが難しい
    - EJBコンテナがないとテストできない
- EJB 3.0では、こうした問題点を解決
  - 大規模アプリケーション開発もEoDへ

- しかし・・・
  - EJB 3.0でも重量級の技術なのは変わらない
  - 分散配置が必要なアプリケーションはごくわずか
    - 他の機能（トランザクション、永続化）は、DIコンテナなど、別の技術で実現可能
  - 単体で動作するWebアプリケーションレベルであれば、EJBは不要

DIコンテナ（Dependency Injectionコンテナ）

- 一言でいうと・・・
  - アプリケーションの部品（コンポーネント）をつなぎ合せる「のり」の役割をする軽量フレームワーク



- EJB批判の中から生まれた軽量のコンテナ
  - 多くのアプリケーションにとってEJBは必要ない
    - 必要なのは、トランザクション管理とオブジェクト永続化だけ
    - コンポーネントの分散配置は不必要
  - 既存のコンポーネントを組み合わせて、必要な機能を揃えるのがDIコンテナ
    - 複数コンポーネントを組み合わせる方法をDependency Injection（依存性注入）と呼ぶ

      ex) Struts + <ビジネスロジック> + Hibernate
    - サーブレットコンテナのほかに特別なサーバは必要としない

- 代表的なDIコンテナ
  - Spring Framework
    - http://www.springframework.org/
  - Seasar
    - http://www.seasar.org/

Ⅱ．ユーザインタフェースはよりリッチに

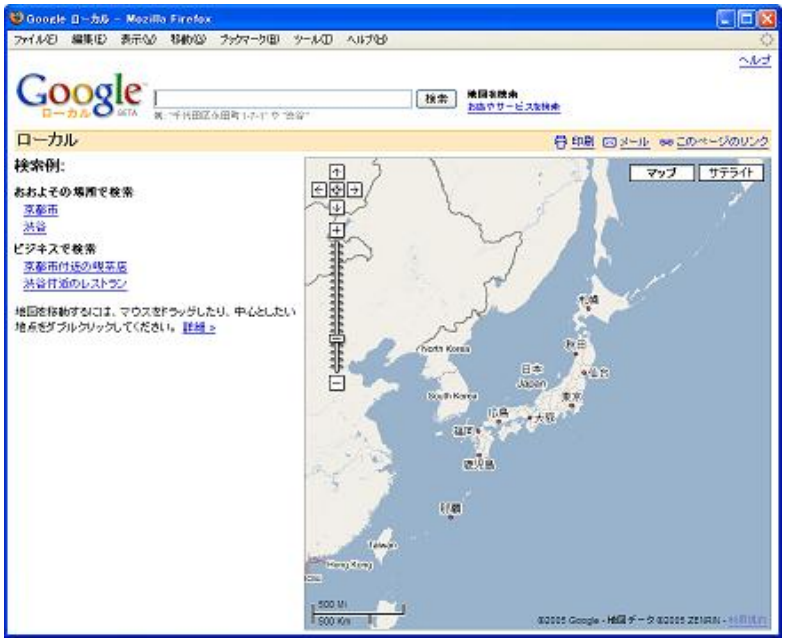
ここで解説するキーワード:

⑤ Ajax

⑥ Flex, OpenLaszlo

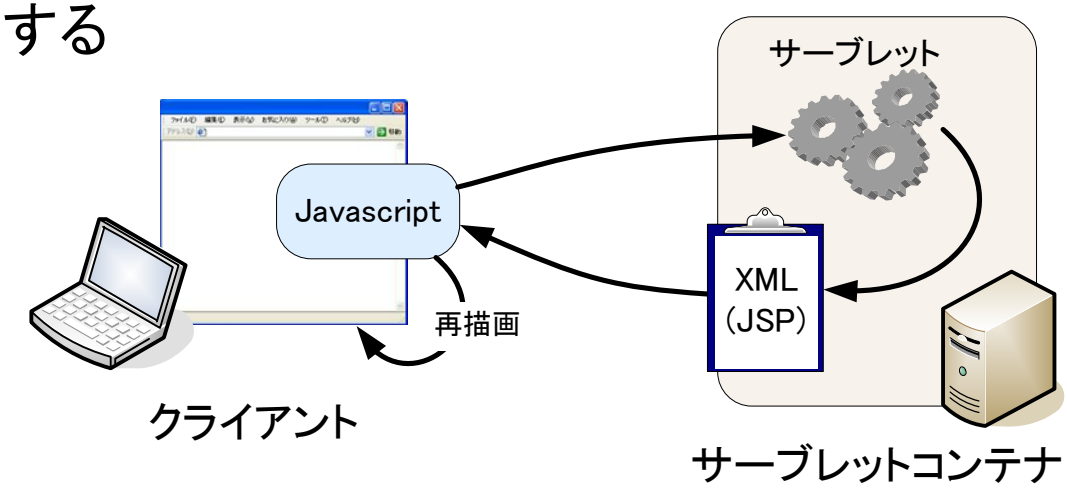
Ajax
（Asynchronous Javascript and XML）

- 一言でいうと・・・
  - Javascriptの機能を最大限に使って、Webアプリケーション上にこれまでにないユーザインタフェースを構築する技術

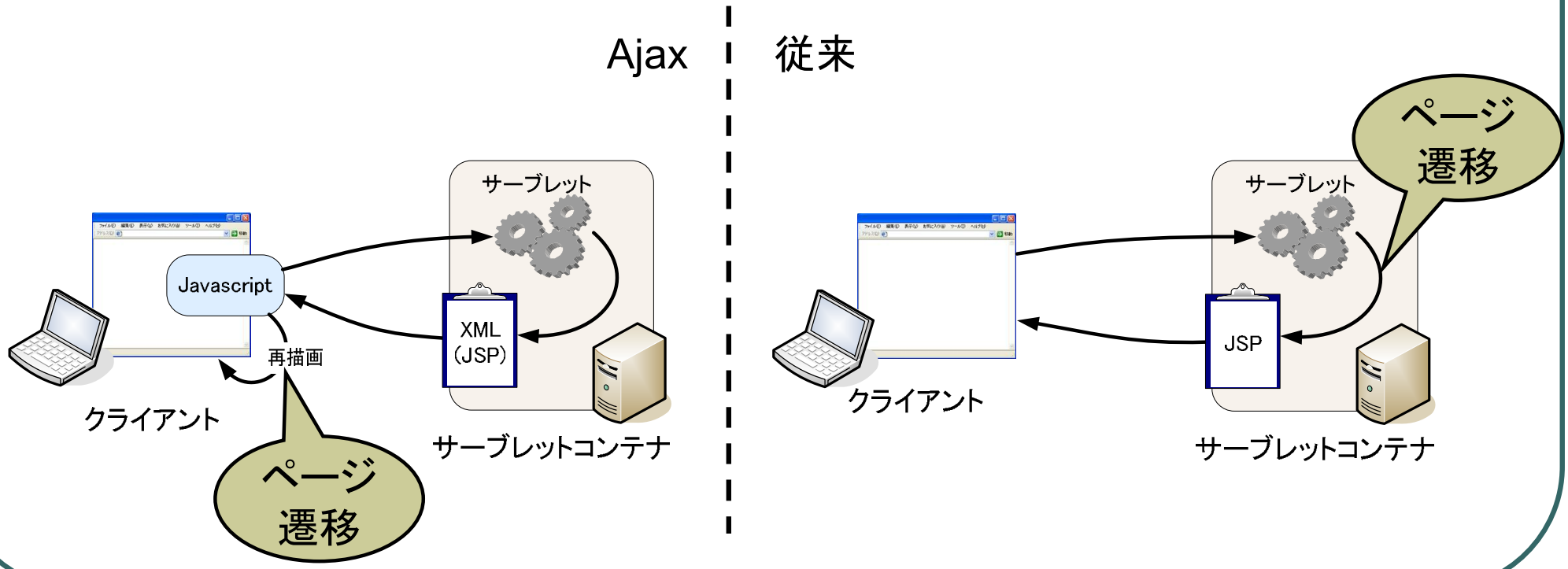


例: Google Map

- Webアプリケーションの操作性がネイティブアプリケーションに近づく
- Ajaxの仕組み（サーバサイドJavaの場合）
  - ブラウザを再ロードせずにサーバと通信し、リアルタイムで画面を再描画するのがポイント
  - JavascriptのXMLHttpRequest（XML通信）、DOM（再描画）を駆使する



- 従来とは異なる開発ノウハウが必要に
  - ページ遷移のコントロールがサーバサイドからクライアントサイドに移る



Flex, OpenLaszlo

- 一言でいうと・・・
  - サーバサイドJava技術と連動するプログラマブルなFlashインタフェースを構築する技術
- Ajaxとは反対に、Flashをサーバサイドで生成できるようにする
  - JSPのようにXMLで記述
  - Flash+ActionScriptと違い、本格的なアプリケーションを簡単に構築できる

Ⅲ. 開発手法はより柔軟で直感的に

ここで解説するキーワード:

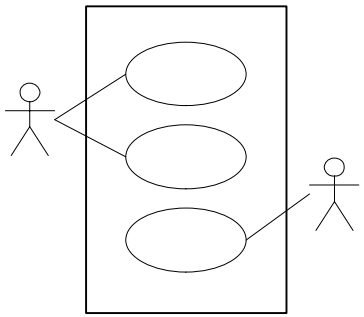
⑦ UML

⑧ アジャイル

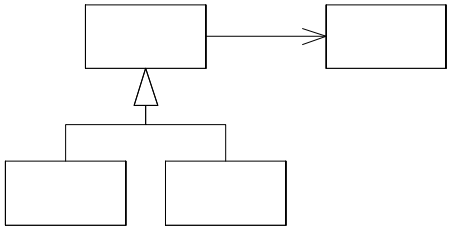
⑨ パターン

UML
（Unified Modeling Language）

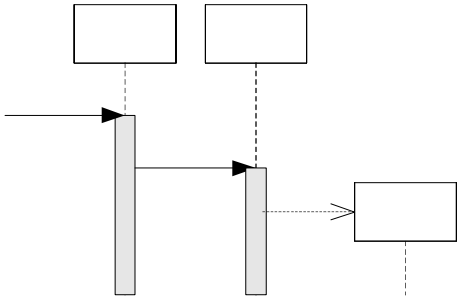
- 一言でいうと･･･
  - オブジェクト指向に基づいた要件定義書／設計図を記述するための標準のモデリング言語



ユースケース図

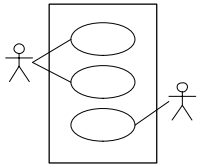


クラス図

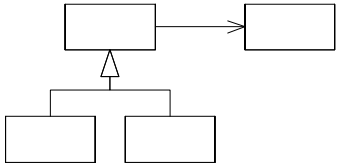


シーケンス図

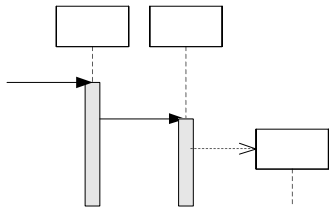
- Webアプリケーションでもドキュメントは必要
  - 無いと、開発契約や保守のときに困る
- Java開発では、UMLを使うのがトレンド
  - 要件定義
    - ユースケース図



  - システムの静的構造
    - クラス図、パッケージ図、配置図

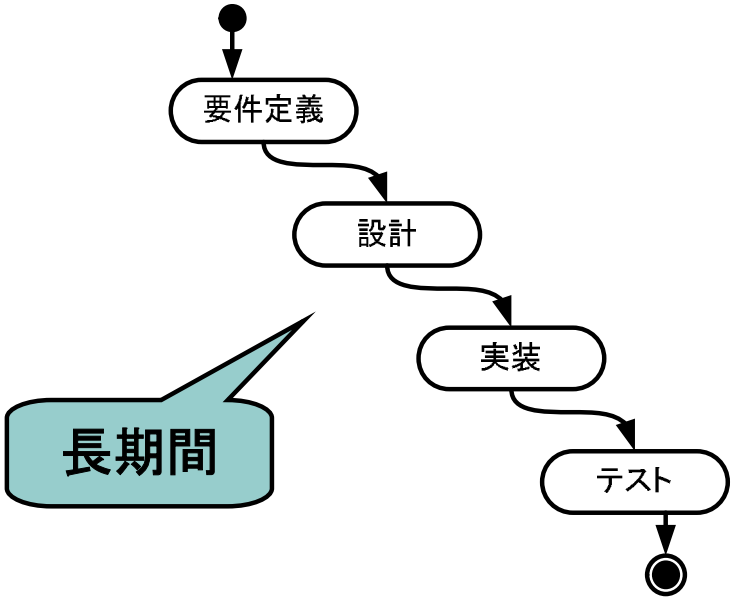


  - システムの動的振舞
    - シーケンス図、コミュニケーション図
      ステートチャート

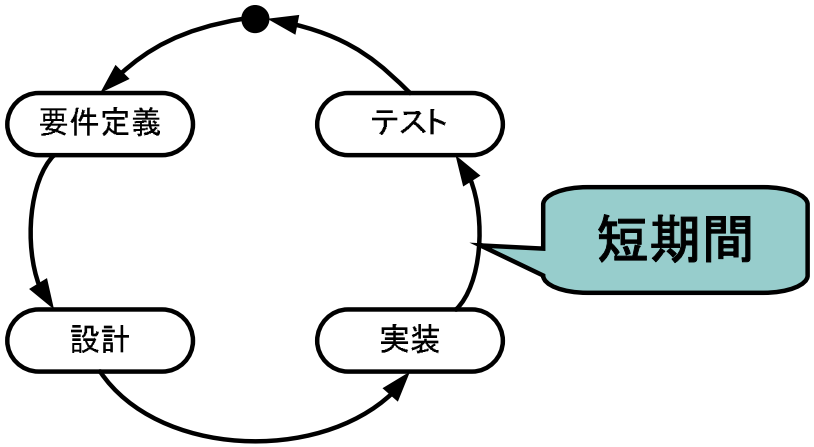


アジャイル

- 一言でいうと・・・
  - 変化する要求に対応し、ビジネス的価値の高いアプリケーションを作るための開発手法の総称



ウォーターフォール型



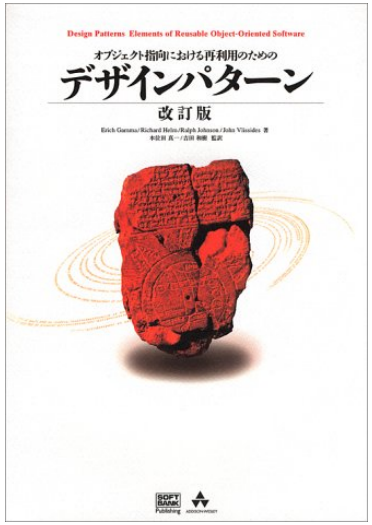
アジャイル開発

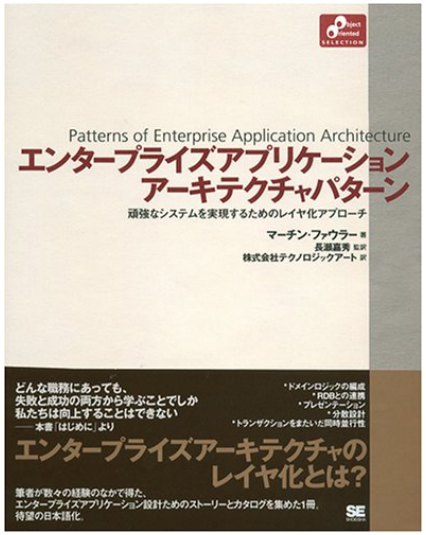
- 従来のウォーターフォール型
  - 要件定義→設計→実装→テスト→納品
  - ユーザの手に届く（納品）までに時間がかかる
  - 前のフェーズへの手戻りが許されない
- アジャイル開発の方法
  - 機能を分けて、2週間～2ヵ月に1回ずつ納品
  - 早い段階でユーザがシステムに触れられる
  - 各フェーズの境界はあいまい

- アジャイルの代表的な開発プロセス
  - XP、スクラム、FDD、クリスタル・・・
- XP（Extreme Programming）の場合
  - テストファースト
  - ペアプログラミング
  - 40時間労働
  - リファクタリング
  - 継続したビルド　・・・　などの実践

パターン

- 一言でいうと・・・
  - 熟練した技術者の優れたノウハウを共有するための、知識のフォーマット





- パターン ＝ 問題 ＋ ソリューション
  - 先人たちの優れた問題解決法を学べる
- 円滑なコミュニケーションツールとなる
  - パターン名が開発者のボキャブラリに
- Webアプリケーション開発の重要な知識がパターンの形でまとめられつつある
  - 『デザインパターン』
  - 『J2EEパターン』
  - 『エンタープライズアプリケーションアーキテクチャパターン』
  - Web 2.0 デザインパターン

まとめ

- Java技術はどんどん開発しやすくなる
  - Javaの信頼性、安全性はそのまま
  - 技術的な煩雑さのみが解消されていく
  - アプリケーションそのものの難しさに専念できる
- ユーザインタフェースはより使いやすいものが求められていく
  - 開発者にとっては、試行錯誤の時代に
- 開発手法の導入が重要になる
  - ドキュメントのある開発
  - 変更要求に強い開発プロセス
  - 先人たちのノウハウを参考にしよう

ありがとうございました

- サーバサイド技術の学び舎 – WINGS
  - http://www.wings.msn.to/
  - サーバサイド技術に関する記事や書籍の情報を公開

2006年.NET最新動向解説

～ Webアプリケーション開発の動向 ～

WINGSプロジェクト

青木 淳夫

<atsuo_aoki@nifty.com>

なにを話すの？

- 2005年12月に登場した最新開発ツール「VisualStudio2005」の話題を中心に、.NET開発をめぐる周辺技術、今後の方向性について説明する。

Ⅰ．.NET最新動向

VisualStudio2005がもたらすインパクト

・VisualStudio2005の概要
・VisualStudio2005のラインナップ
・機能の向上
・開発効率の向上

VisualStudio2005の概要

- Micorosoftが提供している統合開発環境
  - Webアプリケーション開発
  - Windowsアプリケーション開発
  - データベース連携アプリケーション開発
  - Office連携アプリケーション開発

- .NET Framework2.0をベースに動作
  - Windowsのバージョンに依存しない実行環境
  - 多言語に対応した開発・実行環境
    - Visual Basic、C#、C++、J#(Java)・・・

VisualStudio2005のラインナップ

- Team System
  - 大規模チーム開発向け（設計、モデリング、テスト機能）
- Professional Edition
  - 64bit対応アプリケーションを開発可能
- Tools for the Microsoft Office System
  - Office連携アプリケーションを開発可能
- Standard Editon
  - 基本的な機能
- Express Edition
  - 無償公開。評価・教育・ホビー向け。

機能の向上（1）

- Webアプリケーション関連
  - メンバーシップとプロファイル
    - ユーザの管理とセキュリティの管理の簡易化
  - マスターページ
    - Webサイトのレイアウトの統一
  - テーマとスキン
    - Webサイトのデザインとスタイルの統一
- Windowsアプリケーション関連
  - 各種リッチコントロール追加
    - ユーザインターフェースの向上

機能の向上(2)

- ClickOnce
  - Windowsアプリケーションを利用可能
    - リッチなユーザインターフェース画面
    - ショートカットキーの利用
  - 便利なアプリケーション配布展開機能
    - 自動的に[スタート]メニューに登録
    - Webサーバ上の更新ファイルを自動的にチェック
    - 以前のバージョンにロールバック可能
    - 各種設定をVisualStudioから簡単に設定

開発効率の向上

- 組み込みのWebサーバ
  - 1人に1つのWebサーバを準備
- プログラミング環境の向上
  - コードスニペット
    - 良く使う文法や命令の挿入機能
  - インテリセンス
    - 入力候補を表示して入力ミスを減らす機能
  - ステップ実行とエディットコンティニュー
    - プログラムを1行ずつ実行。編集後の継続実行が可能。

Ⅱ．.NET関連の注目技術

.NETに関連する将来の注目技術

・Windows Vista
・リッチUIテクノロジ
・その他の注目テクノロジ

Windows Vista

- WindowsXPの後継OS(開発コードLonghorn)
  - ユーザ向け拡張
    - Windows Sync Center･･･PDAや携帯の同期管理
    - Windows Mobility Center･･･モバイルPCの集中設定管理
    - Windows Photo Gallery･･･写真の管理･修整
    - Windows Media Player 11･･･動画再生
  - WinFx（内部コンポーネント）
    - Win32 APIの後継API
    - .NET Frameworkを元に構成
    - 高度なグラフィック(Avalon)･ネットワーク(Indigo)･ストレージ技術(WinFS)･ワークフロー の機能を提供予定

リッチUIテクノロジ

- Atlas
  - Ajaxアプリケーション開発ツール
  - Visual Studioに統合予定
  - Atlas実験サイト(Start.com)
    - http://www.start.com/
- Sparkle
  - Flashキラー技術（ただし詳細は未定）

その他の注目テクノロジ

- IE7.0
  - タブブラウザ
  - RSS対応
  - 印刷機能の向上
- Office12
  - 標準のデータ形式がXML
- Metro
  - PDFキラー技術（ただし詳細は未定）

Ⅲ．.NETを取り巻く環境

.NETを取り巻く環境とMicorsoft戦略

・オープンソース
・コミュニティ

オープンソース

- .NET関連のオープンソースの増加
  - Enterprise Library
    - マイクロソフトが推進するオープンソース
  - N～,～.NET(一般的なオープンソース)
    - Nunit（テスト）
    - iBATIS.NET(データアクセス)やNHibernate
    - Sprint.NETやSeasar.NET(DIコンテナ)

コミュニティ

- .NET開発者との意思疎通
  - Blogブームによって、開発者の意見を直接聞けたり、コメントが可能に。
- VisualStudioや.NETのコミュニティが活発
  - Micorsoft News Group
  - GotDotNet Japan
  - Visual Studio Users Group
  - フォロー率が高いため、初心者に優しい（MicrosoftMVPによる功績が大きい）

まとめ

- .NETはVisualStudio2005と共にさらなる普及へ
  - Micorosoftのアプリケーション（Windows、Office、IEなど）はバージョンアップとともに機能と品質を向上させてきた。
  - .NET Frameworkも同様に2.0のリリースと共に機能と品質を向上させた。
  - VisualStudioの使いやすさには定評があっただけに、これからさらなる普及が期待される。
- リッチなユーザインタフェースの要望が増えている
  - Ajaxへの期待は高いが、Windowsサーバの場合ClickOnceも検討。
- .NETでも大規模な開発が増加
  - 設計やテストに特化したVisual Studio Team SystemのようなEditionが登場。今後、開発方法論も含めて、注力していく分野と思われる。

ありがとうございました

- サーバサイド技術の学び舎 – WINGS
  - http://www.wings.msn.to/
  - サーバサイド技術に関する記事や書籍の情報を公開