

# ZK

## SIMPLY RICH

### The Simplest Way to Make Web Applications Rich.

インターネットはアプリケーションを開発する際欠かせない土台になってきました。インターネットの開発・使用はテクノロジー市場に大きく貢献しました。IDCによれば、EIP、B2B & B2C アプリケーション、E-コマース市場のそれぞれ 2006 年の CAGR は 41.2%、20%、57.2% に成長する見込みです。

ウェブがわれわれの日常生活に浸透しつつある中、伝統的なページをベースにしたウェブアプリケーションは、アプリケーションの複雑さを視覚的に表現できないという本質的な問題に直面しています。その結果、ユーザが不愉快な経験をしたことと、膨大なコストの二つの大きな問題が発生しました。

10 年の発展を終え、ウェブアプリケーションは静的な HTML ページから動的なページへ、さらにアプレットとフラッシュへ、そして、最終的には Ajax 技術へと進化してきました。Ajax はウェブの "reach" という遍在する特性と、デスクトップの "rich" という使いやすさを結びつけ、ウェブアプリケーションに新たな生命を吹き込みました。Google マップや Netflix は最も知られているその例です。

#### Ajax アプリケーションの挑戦

満足できるアプリケーションの開発は決して簡単ではありません。Ajax でリッチ UI を盛り込むことは、元々コスト高のウェブ開発に、さらに費用やリスクを増やすに違いありません。

また、互換性がなく複雑でバグだらけの JavaScript DOM API を使用するには知識と忍耐力が必要です。非同期プログラミングも簡単ではありません。間違えやすいため、ベテランの開発者しか手に負えません。ビジネスロジックの入った何千台もあるクライアント側の機器のメンテナンスもとても困難です。

そこで、2005 年、ウェブアプリケーションにて Ajax を実現するため、多数のフレームワークやライブラリーが開発されました。JavaScript ライブラリーのものものと、ウィジェット、ウィジェットの入った拡張された HTML、サーバ・クライアント間で通信するのに独自のルールを定義したものなど様々なものが現れました。

しかし、確認した結果、それらの解決案は、すべてひとつの問題、つまり JavaScript の使いにくいという問題しか解決していません。サーバ・クライアント間の非同期の通信は、依然として、アプリケーション開発者が自分で処理しなければなりません。

#### ZK

この問題を解決するため、Potix は "リッチ UI" と "低コストの開発費用" を兼ね備えるウェブアプリケーションのできるツールを開発しました。コアソリューション ZK は、Ajax をベースにしたイベント駆動型エンジンで対話機能を自動的に実現してくれます。また、豊富な XUL・XHTML コンポーネントはツールの便利さを実感させます。さらに、UI のデザインはマークアップ言語を使用するため、プログラミングの必要がなくなりました。

ZK を使えば、アプリケーションを豊かで特徴的な XUL・XHTML コンポーネントで表現することができます。また、デスクトップアプリケーションと同様にユーザー側の操作が行われた場合、その操作はイベントトリガーの形で通知されます。すべてのアプリケーションコードはサーバ側で実行され、ユーザー側の情報の更新はサーバ・クライアント間の同期を通し

て ZK より行います。

#### 特徴 & メリット

**Ajax をベースにしたリッチ UI.** 応答する対話型インターフェースです。ユーザーからみれば、使い方はデスクトップのアプリケーションと同様で、ブラウザでのプラグインも不要です。

**イベント駆動型モデル.** 簡単で直観的です。90 年代デスクトップアプリケーションが発展した一番の理由は簡単さです。ZK はそのアイデアを受け継ぎます。

**XUL をベースにしたコンポーネント.** リッチながら標準的です。60 種類以上 ZK が用意した豊富なコンポーネントを使えば、ウェブアプリをすぐ作成できます。すべて標準的なもので、ベンダロックインの心配はありません。

**ZK ユーザーインターフェースマックアップ言語.** ストレートでプログラミングはいりません。設定不要です。コンポーネントはすべて独立して、XUL・XHTML もしくは混在使用はいずれも結構です。XUL・XHTML のメリットを最大限活用します。

**サーバ中心の処理方法.** クライアント側の処理は不要です。非同期プログラミングもいりません。RMI・RPC 不要。使い慣れているロジックとデータ層がそのまま使えます。

**Java & EL でスクリプトを作る.** プロトタイプをカスタマイズすれば OK です。コンパイルはいりません。JavaScript や DOM もいりません。POJO (Plain Old Java c Ode) ですべてカバーできます。

**モーダルダイアログ.** 選択肢の処理において、ZK は最も直観的な方法でユーザーと対話します。複雑な UI をいくつか処理しやすいダイアログに分解します。

**簡単なスレッドモデル.** スレッドの知識はいりません。サーバは別々のページからの要求を同時に処理できます。

**ライブデータ.** ビジュアルとデータを分けて処理します。大量の資料も効率的に読み込めます。

**GPL.** ベンダロックインの心配はありません。世界中の使用者同士の協力を推奨し、使用者からのフィードバックを活用します。グローバルコミュニティ・お客様に認定されながら、さらに品質を向上させます。