

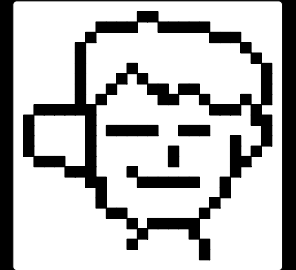
ウェブアプリケーションの 現状と課題

ひらた だいじ

dh@uva.jp, <http://www.daijihirata.com/>

自己紹介

平田大治 (ひらた だいじ)



- シックス・アパート 顧問
- ネオテニー: チーフリサーチマネジャー、他
- 2000年からベンチャー投資
- 2002年からブログと周辺技術の紹介、啓蒙活動
- Movable Type, TypePad の開発、日本語化
- ping.bloggers.jp, [Moblog GW](http://moblog.gw)などを個人でサービス提供
- 個人ブログ: dh memoranda (<http://uva.jp/dh/mt/>)
- その他、詳しくは <http://www.daijhirata.com/> へ

本日の流れ

- あたらめて Web 2.0 とは何か
- 現在のウェブ・アプリケーションの状況
- ウェブアプリケーションの直面する問題
- 今後の行方

あらためて Web 2.0 とは何か

- いつバージョンアップしたのか
 - 2004年の Web 2.0 Conference
 - その後、Where 2.0, When 2.0, Identity 2.0 などいろいろ
- なにが変化したのか？
 - インターネット？
 - ウェブ？
 - ブラウザ？

Web 2.0 前後の変化

- Web 2.0 前夜
 - 検索サービスの飛躍的精度向上
 - Google, Yahoo, MSN, etc...
 - ユーザーコンテンツの飛躍的増大
 - ブログの流行、デジタルカメラの普及、CD から mp3
- そして「Web 2.0」
 - 「ウェブ」をインフラとした、新しいサービスやアプリケーション
 - ウェブサービスや XML Syndication の一般化
 - コンテンツマッチ広告による新しいエコノミーの成立

「ウェブをインフラとする」とは？

- アプリケーションやサービスをブラウザを通して利用する
 - スケジューラ、ワープロ、表計算、プレゼンテーション
- あらゆるコンテンツにブラウザを通してアクセスする
 - 地図、イベント、日記
- ウェブのインフラを利用してサービス連携
 - ウェブサービス、XML Syndication

ウェブ・アプリケーションの状況

- ウェブ・サイトとウェブ・アプリケーションの違い
- レンタルサーバでも、初期状態で様々なアプリを準備
- ブログ/CMS の導入で Web-DB 連携も一般化
- パッケージ製品や開発環境の簡易化で、導入もしやすい
- 特別な存在ではなくなった

ウェブ・アプリケーションが直面する状況

- 利用者サイドとして
 - 急速な普及、一般化への対処
- 提供者、開発者サイドとして
 - 新しい技術への対応
 - 開発期間、品質
 - スケーラビリティ
 - 運用体制
 - セキュリティ
 - プライバシー

ウェブアプリケーションの普及拡大

- 一般化による問題
 - 技術者、詳しい人がいない環境へのパッケージ導入
- セキュリティを考慮したイントラネット内での利用
- パッケージではなくサービスを利用
- より簡単、より安全な製品、サービスを提供しなければならない
 - セキュリティ問題の放置はスパムや DDoS への利用の可能性を増大させる。

利用者側の負担軽減のためにも

開発、提供側の責任は、より重大になってきている

モダンブラウザ

- IE6/7, FireFox, Safari, Opera, etc...
- xhtml/css/JavaScript
- 以前よりもずっと洗練された美しく使いやすいインターフェイスを実現できるようになった
- prototype.js を始めとする JavaScript ライブラリの普及
- クロスブラウザでの利用も現実的になってきた



非同期ウェブアプリケーション

- JavaScript の活用
 - Ajax (Asynchronyzed JavaScript and XML)
 - JSON (RFC4627) / JSONP
- Comet
- サーバサイドの非同期タスクとの連携
- ウェブサーバの設計からの変更を伴なう
 - 新しいノウハウの吸収、構築

開発工期、品質

- ベータ版でのリリースの一般化
 - 永遠にベータ？ 何がベータ？
 - 品質も重要な価値の一部、サポートコストなどにも反映される
 - ラボ？
- 製品/サービス開発プロセスの短縮
 - リリースサイクルが短縮されている
 - マニュアルやサポートにも影響する
- 開発プロセス
 - ウォーターフォールからアジャイルへ
 - 各種支援ツールの充実

開発の環境、現場

- Lightweight Language による開発の普及
 - 本格的な高級プログラミング言語としての活用
 - ハードウェアの性能向上によりオーバーヘッドを吸収
- オープンソースベースのソフトウェアの普及
 - 新しい技術がスピーディーに導入できるようになった
 - 利用者の増大による品質の向上
- 安定志向と機能拡張のバランス
- 利用者がいないと性能, 品質向上のループが働かない
 - みずからコントリビューションを率先しよう

開発の環境、現場 (cont.)

- ブラックボックス化の利点、弊害
 - 各種オープンソース・ライブラリの活用による開発成果の共有
 - 品質は高いがバグがないことは保証されない -- 検証が重要
 - クリティカルな局面には、中身も理解しているほうが有利
- 検索結果への過度の依存はないか
 - 性能、セキュリティ、著作権的に検証なしの利用はよろしくない...

スケーラビリティへの要求

- ビジネスからの要請、要望
 - 当初はスモールスタート
 - 利用者、収益の拡大にあわせて規模を拡大
- スモールスタートの予算
 - サーバ数台 - 場合によっては一台から
 - ラボ?
- 規模拡大のスピード
 - 早いときはとても早い
 - まったく伸びない可能性もある

スケーラビリティの確保

- スモールスタート
 - サーバの台数、規模だけの問題ではない
 - アプリケーションの設計、開発段階にも影響すること
 - ビジネスとエンジニアリングが共同してサービス設計しないと対処できない -- ベンチャー企業の得意なエリア
- スケールしやすい設計
 - 複雑にしない
 - 最初から複数台での提供も視野にいれる
 - 安定性確保の面でも必須
 - アーキテクチャ自体の変更もおりこんで、ロードマップを持っておく

システム構成

- 冗長構成、二重化の重要性
 - 稼働率 99.9% のシステム
 - 二重化したときには 99.9999%
 - 二重化せず、単に二系統にすると 99.8%
- High Availability(HA) と Scalability は違う
 - 両方とも重要、両立させないといけない
 - どこまでやるか...
ビジネス上のインパクトを睨みながら可能なリスクをとる
- サービスの種類、規模、ブランド、会社の成長段階とも関連

運用体制

- 24h/365d 運用はとても大変
 - スモールスタート時のコスト高要因
- かけられるコストとのバランス
 - 安定性確保のためにも SPOF (Single Point of Failure) がないシステムが望ましい -- 大規模運用では必須
- SLA や利用規約との整合性
- 日常のヘルスチェックの重要性
 - 同時に複数台こわれたら...
- サポートコストとのバランス

セキュリティ

- これまでのウェブでのセキュリティ
 - サーバ設定のミスやバージョン管理、不正スクリプトの存在
- XSS/XSRF
 - フォームを経由したHTMLへのコードの挿入が
css/xml/js へと対象が拡大
- JavaScript
 - 基本的にはブラウザに依存
 - JSONP によるクロスサイト・アクセス -- 3rd party への依存性
- インターフェイスでの十分なパラメータチェックがより重要に

セキュリティ問題について

- 万能薬はない
 - 開発言語の問題ではない
 - Firewall とか IDS であっさり解決する問題ではないが有効な場面もしばしば存在する
- 旧来からのやり方を踏襲することで対応可能な部分も多い
 - パラメータのサニタイズ、フィルタリングを厳重に行えばかなりの部分は防止できる
- 利用者としての自衛手段
 - リテラシーのさらなる向上?
 - No Script

プライバシー

- 名寄せリスクの増大
- 本人だけでなく、家族や友人にも被害が及ぶ可能性
 - SNS やブログのリンクによるリレーシヨンの可視化
- 一度公開された情報を取り消すのは非常に困難

プライバシーコントロール

- 見せたい人、見せたくない人をコントロール
- 技術的には十分に可能
 - メンバーシップ制
 - アクセスコントロール
- ネットの外の動きはコントロールできない
 - 画面を覗き込まれた
 - 勝手にコピーされて散蒔かれた
- 本当に大切な情報は、技術だけでなく十分に配慮してアクセスされるようにする必要がある

まとめ: 今後のゆくえ

- 短期的には
 - 開発環境の変化への対応
 - mush up によるクロスサイトでのセキュリティリスク
 - SNS におけるプライバシーリスク
- 中長期には
 - セキュリティの対策方法の浸透
 - プライバシー情報の保護と利用のバランス
- 地道な努力なしには次のステージにはいけない

ありがとうございました

contact: dh@uva.jp, <http://www.daijhirata.com/>,

シックス・アパート株式会社 <http://www.sixapart.jp/>

dh@uva.jp, <http://www.daijhirata.com/>