

注目を浴びる Ajax

実は老舗の Ajax

JavaFest in Sapporo 2005/12/9

株式会社エクサ

安藤幸央 yukio-ando@exa-corp.co.jp

Who am I ?

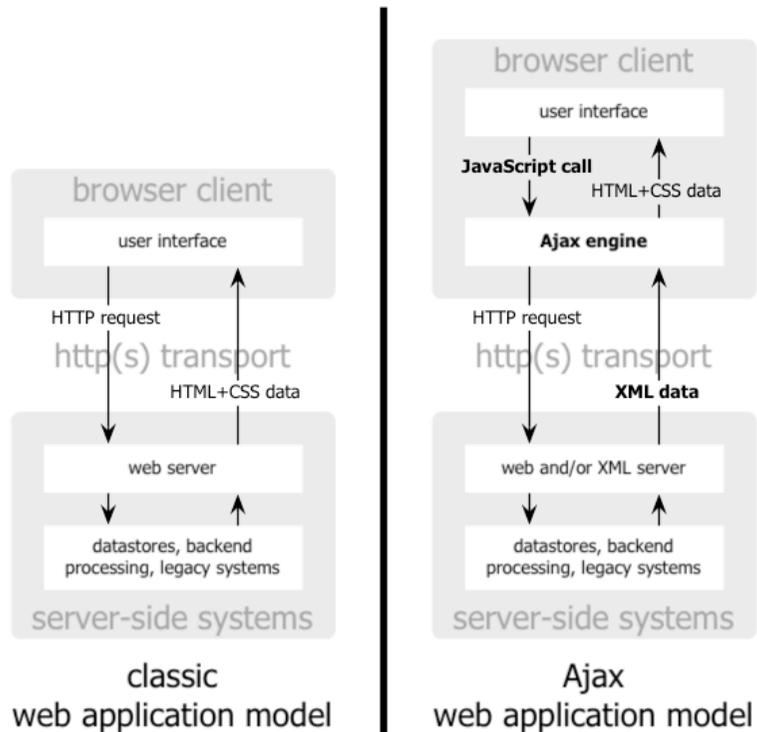
- ▶ [JavaNews.jp](#)
- ▶ [Java FAQ](#)
- ▶ [Ajax blog](#)
- ▶ [OpenGL FAQ](#)
- ▶ [Google WebLog](#)
- ▶ [Digital Gadget / Software Design](#)
- ▶ [SIGGRAPH Tokyo Chair](#)

Agenda

- ▶ Ajax とは？
- ▶ Ajax の利用例
- ▶ Ajax の仕組み
- ▶ Ajax のユーザビリティ
- ▶ Ajax の落とし穴
- ▶ Ajax ベースの開発とその周辺
- ▶ Ajax フレームワーク・ツール活用

Ajax とは？

- ▶ Asynchronous JavaScript And XML
- ▶ Adaptive Path / Jesse James Garrett

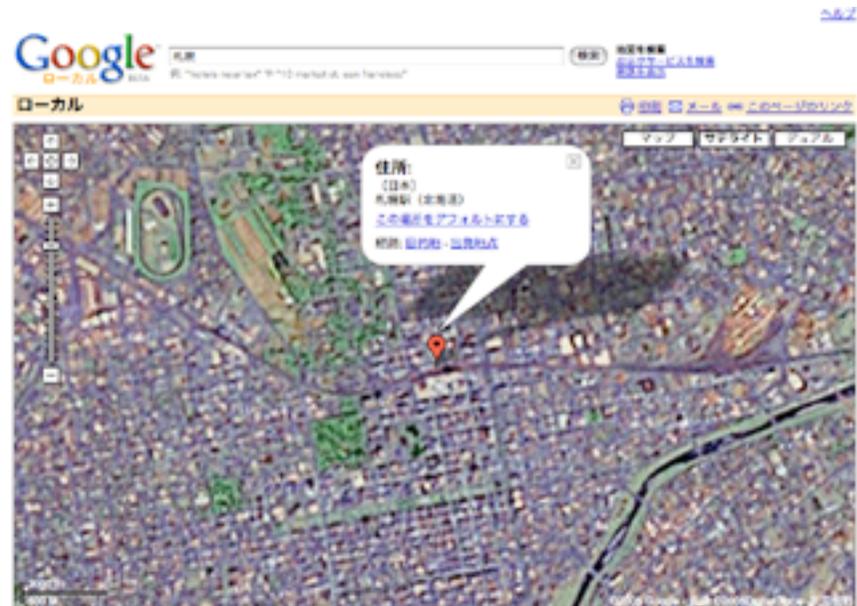




Another Ajax

Ajax の利用例

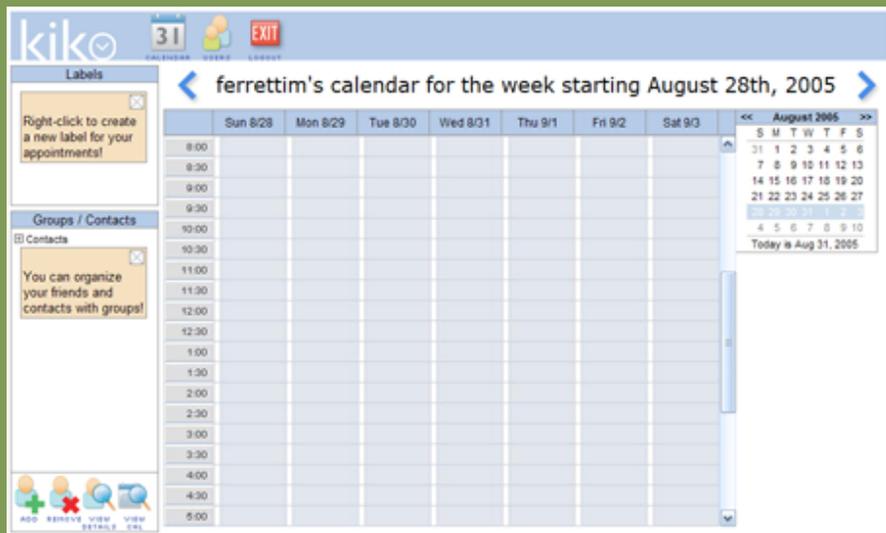
- ▶ Google Maps / Google Moon
- ▶ Google Suggest
- ▶ Google RSS Reader
- ▶ Amazon's diamond search (f)
- ▶ Amazon A9
- ▶ Microsoft Start.com (f)
- ▶ goo 地図
- ▶ 訳 GO
- ▶ Amazon Zuggest



Top 10 Ajax Applications

- ▶ Kiko : ウェブカレンダー
- ▶ Backbase's RSS Reader : ブログ更新チェックツール
- ▶ Backpack :ToDo リスト+プロジェクト管理
- ▶ Writely : オンラインワープロ
- ▶ Amazon Zuggest : ダイナミックな検索
- ▶ TimeTracker : 個人向け時間管理ツール
- ▶ Del.icio.us Director : ブックマーク管理
- ▶ Backbase's Information Portal : 情報管理ページ
- ▶ Protopage : 情報ポータル
- ▶ Periodic Table of the Elements : 元素記号表

Ajax Office



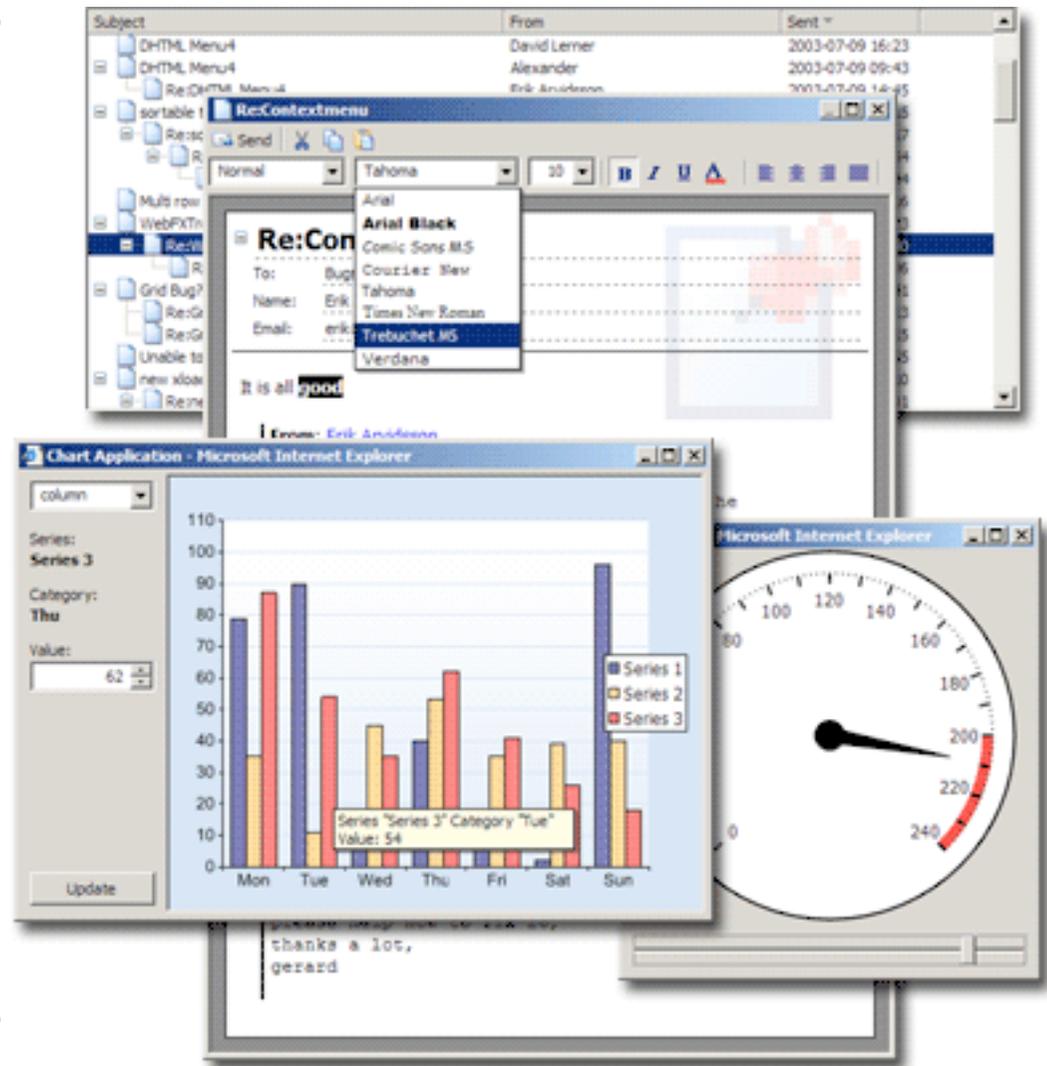
- ワープロ
Writely / JotSpot
<http://www.writely.com/>
<http://www.jot.com/>
- 表計算
Num Sum
<http://numsum.com/>
- プレゼンテーション
S5 / Presentacular
<http://www.meyerweb.com/eric/tools/s5/>
<http://labs.cavorite.com/presentacular>
- メール
Yahoo! Mail / Gmail
<http://whatsnew.mail.yahoo.com/>
<http://gmail.com/>
- カレンダー
Kiko / JuicyCal
<http://www.kiko.com/>
<http://juicycal.com/>
- メモ帳
Webnote
<http://www.aypwip.org/webnote/>
- ホームページビルダー
inetWord
<http://www.inetword.com/>

Ajax Office or Not?

- ▶ Google (Ajax) + Sun (OpenOffice)
- ▶ セキュリティが心配
- ▶ プライベートな情報がサーバに転送される不安
- ▶ ネットワークが繋がっているところでないと使えない
- ▶ ローカルアプリケーションの方が高速で高機能
- ▶ 使い慣れているアプリケーションとの使い勝手の違い
- ▶ ASP化のチャンス (ドキュメントの校正、デザイン、清書、翻訳、回覧、大量印刷、製本、郵送、共有、保存)

Bindows

- ▶ <http://www.bindows.net/>
- ▶ GUI 部品のリ再利用



Top 20 Ajax Applications

- カレンダー: CalenadarHub
<http://www.calendarhub.com/>
- 辞書/シソーラス: ObjectGraph Dictionary
<http://www.objectgraph.com/dictionary/>
- メール: Zimbra
<http://www.zimbra.com/>
- メール(2): Groowy
<http://www.goowy.com/>
- コード/部品サンプル: The MidnightCoders
<http://www.themidnightcoders.com/examples/>
- Flickr ビューワ: Lightweight Flickr Viewer
<http://webdev.yuan.cc/lfvr/>
- ゲーム: Boggle
<http://weboggle.shackworks.com/>
- ゲーム(2): Babble
<http://www.playbabble.com/>
- メッセージャー: Meebo
<http://www.meebo.com/>
- メッセージャー(2): Treehouse Chat
<http://treehouse.ofb.net/chat/?lang=en>
- ToDo リスト: TuDu
<http://tudu.ess.ch/>
- スペルチェッカー: Garrison Locke
http://www.broken-notebook.com/spell_checker/
- コードサンプル: script.aculo.us
<http://script.aculo.us/demos/>
- ホームページ用サイト: Netvibes
<http://www.netvibes.com/>

Ajax Q&A

- ▶ Ajax を発明したのは誰？
- ▶ Ajax は商標？商品？
- ▶ Ajax = XMLHttpRequest？
- ▶ 何が新しいの？
- ▶ どんなアプリケーションに適しているの？
- ▶ Ajax は反 Flash？
- ▶ Ajax のアクセシビリティは？
- ▶ Ajax は XML, XSLT を使わないといけなの？
- ▶ 開発は簡単？
- ▶ Ajax を使うとユーザエクスペリエンスが良くなるの？

Ajax を発明したのは誰？

Google でも Adaptive Path / Jesse James Garrett でも 誰でも無い

Ajax は商標？商品？

特定の技術を組み合わせたウェブアプリケーションアーキテクチャの考え方。
Ajax という商品があるわけではない。

Ajax = XMLHttpRequest ?

XMLHttpRequest は非同期通信のための
コンポーネントのこと。

Ajax は非同期通信のアプローチのこと。

Ajax は何が新しいの？

技術的には前からあったものの組み合わせ。
Ajax が顕著に利用され、定着しつつある。
ブラウザ環境が新しくなってきた。

どんなアプリケーションに
適しているの？

まだわからない。

新しいアプローチなので模索中。

従来のアプリケーションが良い場合も。

Ajax は反 Flash ?

共存可能。

ある局面では Ajax、ある局面では Flash。

二つを組み合わせたサービスも！

Ajax のアクセシビリティは？

「戻る」ボタンの自由度は？

特徴と限界を知る。

的確な視覚効果をユーザに提示する。

Ajax は XML, XSLT なの？

XML が使いやすい構造であるから。
データ構造は自由。

Ajax の開発は簡単？

難しい。

開発環境を整える。

的確なフレームワークを採用。

ユーザエクスペリエンスは？

無限の可能性と柔軟性を与える。
慣れていないので注意深く利用する。
使いやすく。分かりやすく。

Ajax の利用率

1. I am currently using Ajax frameworks, toolkits, or libraries:

		Response Percent	Response Total
In production.		31.2%	238
In development.		32.8%	250
In a proof-of-concept or prototype.		26.5%	202
In nothing at this time.		9.6%	73
Total Respondents			763
(skipped this question)			0

Ajax と言語

2. I'm using Ajax in conjunction with:

		Response Percent	Response Total
	ColdFusion	3.5%	27
	Java (J2EE, JSP, JSF, Tomcat, etc.)	35.1%	268
	Microsoft.NET (C#, ASP.NET, VB.NET)	17.6%	134
	PHP	40.2%	307
	Perl	6%	46
	Python	6.9%	53
	Ruby on Rails	14.5%	111
View	Other (please specify)	8.8%	67
Total Respondents			763

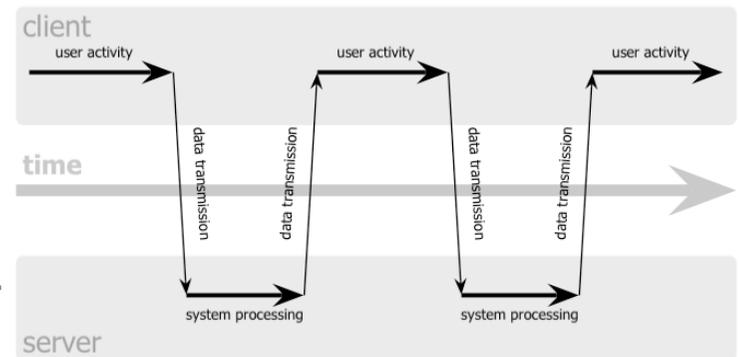
Ajaxに必要なスキル

- ▶ JavaScript (ECMA Script / JScript)
- ▶ DHTML (DynamicHTML)
- ▶ CSS (Cascading Style Sheets)
- ▶ http (http status)
- ▶ XML (eXtensible Markup Language)
- ▶ XSLT (eXtensible Stylesheet Language Transformations)
- ▶ DOM (Document Object Model)
- ▶ Java(Servlet/JSP/JSF) or PHP or .NET

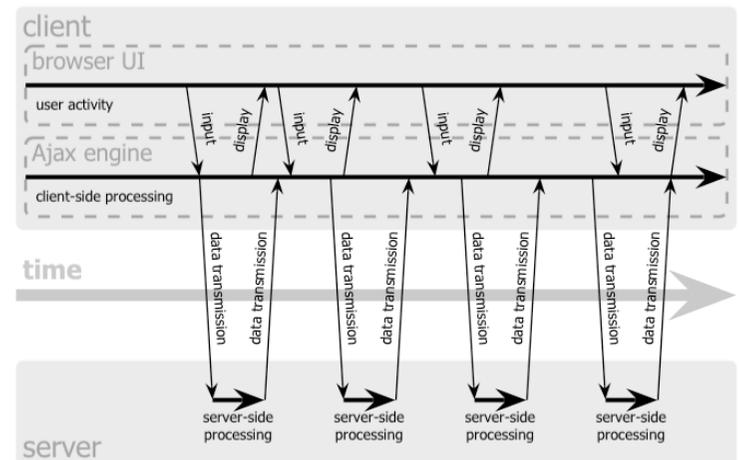
Ajax の仕組み

- ▶ (XHTML+CSS)DHTMLを用いた,Web標準に基づくプレゼンテーション
- ▶ DOM(Document Object Model)によるダイナミックな表示と相互作用
- ▶ XMLとXSLT (XSL Transformations)による,データの変換や操作
- ▶ XMLHttpRequestによる,データの非同期的な取得
- ▶ 上記の技術をJavaScriptによって相互に結び付けるもの

classic web application model (synchronous)



Ajax web application model (asynchronous)



Ajax の本質

- ▶ アプリケーションやプラグインのインストール必要なし
- ▶ サービスのバージョンアップがサーバー側だけで可能
- ▶ データの自動保存により、直前の状態が呼び戻せる
- ▶ サーバーとの非同期に通信する、もたつきの無い GUI
- ▶ サーバーとのやりとりは RPC ではなくメッセージング
- ▶ UI の再構成/再表示はサーバ側ではなくクライアント側
- ▶ GUI 側にもロジックの組み込みが可能。操作のたびにサーバーに問い合わせずとも的確な表示ができる
(検索リストの絞り込み、ソーティングなど)

Ajax 開発スタイル

▶ XMLHttpRequest を生で使う

▶ Prototype.js

▶ Script.aculo.us

▶ DWR

▶ Dojo

▶ Ajax.NET

▶ SAJAX



結局はこれ？ 約40%の開発者

XMLHttpRequest

- ▶ IE 5.0+ (1999.3~)
- ▶ Opera 7.6+
- ▶ Safari 1.2.4+
- ▶ Firefox 0.8+
- ▶ Netscape 7.1+
- ▶ Mozilla 1.4+
- ▶ Konqueror 3.0+

Ajax の通信方法

- ▶ XMLHttpRequest : 高速だが互換性が無い
- ▶ iframe タグ : 遅い、utf8 以外にも対応
- ▶ script タグ : 外部ドメインの .js スクリプトを利用可能

```
var xmlhttp = this.XMLHttpRequest ?  
    new XMLHttpRequest() : new ActiveXObject("Microsoft.XMLHTTP");  
xmlhttp.onreadystatechange = function() {  
    if (xmlhttp.readyState == 4) {  
        if (xmlhttp.status == 200) {  
            alert(xmlhttp.responseText);  
        } else {  
            alert("Error: " + xmlhttp.status + " " + xmlhttp.statusText);  
        }  
    }  
}  
xmlhttp.open("GET", "/cgi-bin/test-ctrl", true);  
xmlhttp.send(null);
```

▶ XMLHttpRequest Properties

onreadystatechange	状態が変化した時に呼び出されるイベントハンドラ
readyState	オブジェクトの状態 0:未初期化、1:読込、2:読込済、3:対話中、4:完了
responseText	通信結果を文字列として取得できるプロパティ
responseXML	通信結果の XML オブジェクト (DOM 互換ドキュメントオブジェクト)
status	HTTP レスポンスステータス 200 : OK、404 : Not Found、 etc.
statusText	HTTP レスポンスステータスの文字列 “OK”, “Not Found”、 etc.

▶ XMLHttpRequest method

abort()	通信を中断。 連続で send する場合、send の前に呼び出す
getAllResponseHeaders()	HTTP レスポンスヘッダを文字列として取得
getResponseHeader(label)	指定したレスポンスヘッダの内容を取得
open(method, url, asyncFlag, username, password)	指定した URL に接続 method : get または post を指定 asyncFlag : 通信の非同期フラグ (true で非同期) username, password : Basic認証用、Digest認証用
send(content)	HTTP リクエストを送信 content として送信データもしくは null を送信
setRequestHeader(label, value)	HTTP リクエストヘッダを付加、または置き換える

Edit the text and click me

```
<html>
<head>
<script type="text/javascript">
var xmlhttp

function loadXMLDoc(url)
{
// code for Mozilla, etc.
if (window.XMLHttpRequest)
{
xmlhttp=new XMLHttpRequest()
xmlhttp.onreadystatechange=state_Change
xmlhttp.open("GET",url,true)
xmlhttp.send(null)
}
// code for IE
else if (window.ActiveXObject)
{
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP")
if (xmlhttp)
{
xmlhttp.onreadystatechange=state_Change
xmlhttp.open("GET",url,true)
xmlhttp.send()
}
}
}

function state_Change()
{
// if xmlhttp, check "loaded"
```

Edit the text above, and click on the button to see the result.

Using the HttpRequest Object

status: 200

status text: OK

response:

Tove Jani Reminder Don't forget me this weekend!

XMLHttpRequest sample

Ajax の通信データ形式

- ▶ XML
- ▶ plane text
- ▶ Base64 binary encode
- ▶ JavaScript eval
- ▶ JSON (JavaScript Object Notation)

Ajax のユーザビリティ

- ▶ ブラウザが JavaScript を必要なことをユーザに通知
- ▶ ページが動的に更新されることをユーザーに知らせる
- ▶ 情報が更新されたことユーザに知らせる
- ▶ 情報が更新された領域をハイライト表示 (YFT*)

Ajax を使うのがお薦め

- ▶ フォームによる多量の数値・文字入力
- ▶ 深いツリー構造を操作するところ
- ▶ ユーザとのコミュニケーションを行うところ
- ▶ 投票や「はい/いいえ」、「良かった」など単純評価
- ▶ フィルタやソートなどデータリスト表示に関わるところ
- ▶ テキスト入力時のヒントや、自動入力補完するところ

Ajax を使わないところ

- ▶ 入力が単純で、失敗しにくいところ
- ▶ 単純な検索（Google Suggest は未だベータ扱い）
- ▶ サイトナビゲーションなど、Webとして基本的なところ
- ▶ 大量のテキストを更新して見せるところ
- ▶ 表示の変化が大きく切り替わる場所
- ▶ Ajax の必要のない複雑なユーザインタフェース部品

Ajax の落とし穴

- ▶ 動作中にURLが変化しないため、ブックマークが困難だったり、人にメールして知らせるのが難しくなる
- ▶ ページビュー、アクセス数の概念が成り立たなくなる
- ▶ 単なるアクセス数ではなく、テレビのようにそのサービスにどれだけ滞在したかという視聴率が重要になる
- ▶ 各種Webブラウザごとにわずかに違う挙動に対応するのが難しい
- ▶ 開発/デバッグが困難(的確なフレームワークが求められる)
- ▶ ブラウザのJavaScriptがONになっていないといけない
- ▶ 安易にリロードしてしまうと、今までの操作が破綻する

Ajax が損なところ

- ▶ URL が変化しないため、リンクできない
(location.hash で逐一用意することもできる)
- ▶ ロボット（検索エンジンのクローラ）用のページを用意しないと効果が無い
- ▶ 非同期通信のシーケンス設計がとても難しい
- ▶ JavaScript/CSS 実装の異なるブラウザが存在
- ▶ ツールがまだ整備されていない。デバッグが困難

Ajax Usability Mistakes

- ▶ Not giving immediate visual cues for clicking widgets.
- ▶ Breaking the back button
- ▶ Changing state with links (GET requests)
- ▶ Blinking and changing parts of the page unexpectedly
- ▶ Not using links I can pass to friends or bookmark
- ▶ Too much code makes the browser slow
- ▶ Inventing new UI conventions
- ▶ Not cascading local changes to other parts of the page
- ▶ Asynchronously performing batch operations
- ▶ Scrolling the page and making me love my place
- ▶ Blocking Spidering

Ajax ベースの開発とその周辺

- ▶ JSMIn : JavaScript Minifier
- ▶ Squish : JavaScript code compactor
- ▶ ShrinkSafe : JavaScript compressor
- ▶ Swell : 難読化された JavaScript コードを整形
- ▶ Ajax debugging Greasmonkey : firefox plugin
- ▶ IE Leak Detector : IE のメモリーリーク発見ツール
- ▶ DOM Inspector : DOM 構造視覚化ツール
- ▶ JSDoc : JavaScript documentation tool
- ▶ JsLint : JavaScript 文法チェックツール
- ▶ JsUnit : JavaScript Unit Test tool
- ▶ Selenium : JavaScript auto test tool

Dojo's Compressor

- ▶ JavaScript の圧縮、難読化
- ▶ JSMIn, Squish, ShrinkSafe, and more....
- ▶ 高速化と隠蔽化に貢献
- ▶ Swell で難読化されたスクリプトを見て勉強

Ajax Debug

- ▶ サーバー側のエラーログをチェック
- ▶ Venkman JavaScript Debugger
- ▶ debug.js : デバッグプリントを効率的に
- ▶ JsUnit : JavaScript の単体テスト
- ▶ JsLint : JavaScript の文法チェック。バグを回避

Ajax Security

- ▶ sandbox
- ▶ Cross Site Scripting
- ▶ CGI Proxy
- ▶ Servlet Proxy
- ▶ script tag

Ajax Performance

- ▶ Drip : Internet Explorer Leak Detector
- ▶ Venkman Profiler
- ▶ Process Explorer
- ▶ JsLint

Ajax Accessibility

- ▶ 最初に JavaScript が必要であることを通知、もしくは自動検出
- ▶ ページが自動更新されることを意識させる
- ▶ ページが更新されたこと、更新完了したことを知らせる (Spinner アイコン)
- ▶ 最近更新された箇所を短時間だけでもハイライト表示 (Yellow Fade Technique, Fade Anything Technique)

Ajax フレームワーク・ ツール活用

- ▶ Dojo : UI ツールキット
- ▶ DWR(Direct Web Remoteing) : Java 用ライブラリ、テスト環境
- ▶ Prototype.js : 低レベル JavaScript ライブラリ
- ▶ SAJAX : 対 PHP, Perl, Python 用の Ajax ライブラリ
- ▶ script.aculo.us : Web2.0 スタイル用のライブラリ
- ▶ Rico : リッチクライアント用 JavaScript ライブラリ
- ▶ MochiKit : テストドリブンな開発のためのライブラリ

Prototype.js

- ▶ 複数のブラウザの JavaScript 実装の違いを吸収・隠蔽化
- ▶ 公式ドキュメントが皆無
- ▶ Ajax に必要なオブジェクトをカプセル化して平易に利用
- ▶ IEは 6.0 のみ対応。その他新しめのブラウザに対応
- ▶ Ajax.Request
- ▶ Ajax.Updater
- ▶ Ajax.PeriodicalUpdater

Ajax tag library

- ▶ Ajax Pages
- ▶ JavaScript Template Engine
- ▶ テンプレートの拡張子は .ajp (ajAX pAGES)
- ▶ PHP, JSP, ASP 風のテンプレートエンジン
- ▶
- ▶ Ajax Anywhere
- ▶ JSP/Struts/Spring/JSF で活用

Ajax IDE/Tools

- ▶ Microsoft Atlas
- ▶ Genuitec MyEclipse Enterprise Workbench
- ▶ TIBCO General Interface
- ▶ Turbo TIBET
- ▶ JackBe Visual Ajax GUI Development Tool
- ▶ telerik r.a.d.callback
- ▶ RIALTO
- ▶ Interakt JS Eclipse
- ▶ AjaxBuilder

Ajax より簡単 AHAH

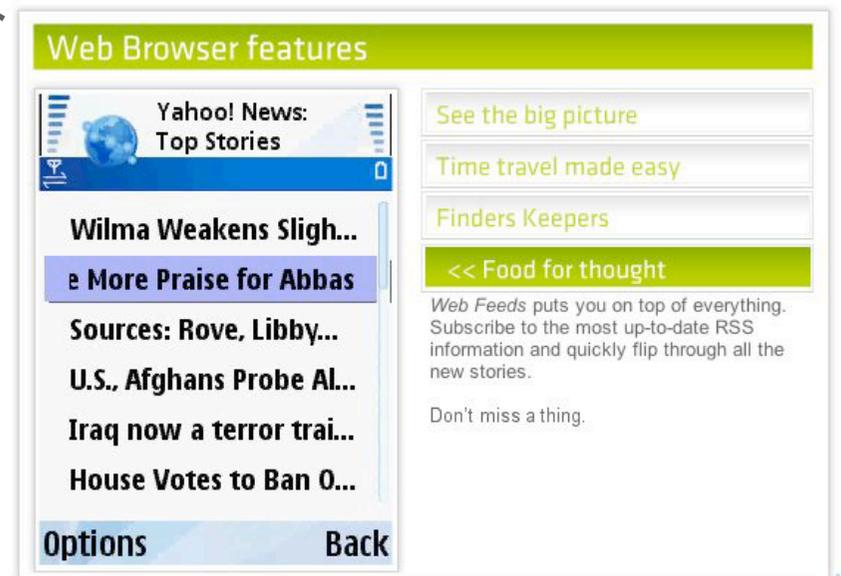
- ▶ AHAH (Asynchronous HTML and HTTP)
- ▶ <http://microformats.org/wiki/rest/ahah>
- ▶ XML を利用している部分をただのHTML で表現

Ajax より複雑 Arax

- ▶ Arax (Asynchronous RPC and XML)
- ▶ RPC (Remote Procedure Call)
- ▶ http://www.opensource.co.jp/Members/admin/news/dyamic_cockpit_technology

Ajax 携帯電話

- ▶ Nokia S60 OSS Browser
- ▶ <http://opensource.nokia.com/projects/S60browser/index.html>
- ▶ DHTML, CSS, SVG-Tiny サポート
- ▶ JavaScript 1.5 サポート

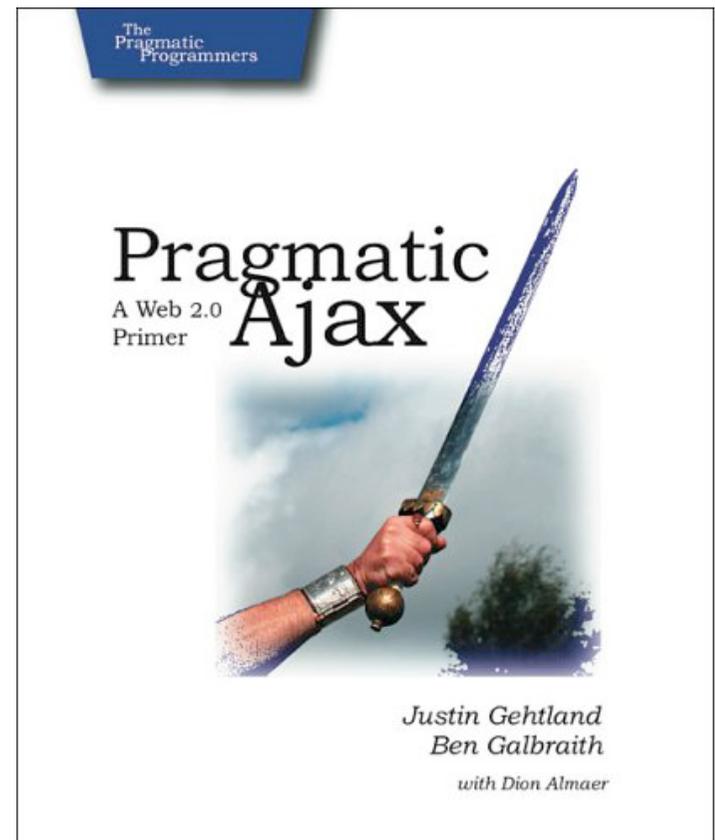


Ajax のビジネスモデル

- ▶ ページビュー、アクセス数の意味の無い世界
- ▶ どれだけページに停滞したかという視聴率的評価が必要
- ▶ Flash と違い、ライセンスに縛られない
- ▶ Ajax 化されていないものを Ajax 化することによる利点
- ▶ 検索エンジンへの見せ方。クローラー対策
- ▶ コンサルティングとサービスがビジネスモデル (JBoss他)
- ▶ 競合は各種リッチクライアント技術 (Curl, Laszlo, Flash他)

Ajax Books

- ▶ 入門 Ajax
- ▶ Ajax 実装のための基礎テクニック
- ▶ Ajax in Action
- ▶ Foundations of Ajax
- ▶ Pragmatic Ajax (PDF公開)
 - ▶ Professional Ajax
 - ▶ Ajax Patterns and Best Practices
 - ▶ Beginning AJAX
 - ▶ AJAX Programming with Java
 - ▶ Ajax Design Patterns
 - ▶ Ajax for Dummies
 - ▶ Head Rush Ajax



Ajax Resources

- ▶ Ajaxian
- ▶ Ajax Matters
- ▶ del.icio.us <<http://del.icio.us/tag/ajax>>
- ▶ Technorati <<http://www.technorati.com/tag/Ajax>>
- ▶ fiftyfourevelen.com
- ▶ feedmarker <<http://www.feedmarker.com/tags/ajax>>
- ▶ Ajax blog <<http://ajaxblog.com/>>
- ▶ Ajax Magazine <<http://ajax.phpmagazine.net/>>

Summary

- ネーミングの勝利！
- 今すぐ Ajax を始めよう！
- 使いやすい Web を！
- Web UI 以上の新体験を！
- ノウハウの共有と蓄積！



Gmail :
yukio.ando@gmail.com