Get Rich Now!
Selecting A Rich Client Technology

Vladimir Vivien
vmatters@gmail.com

Objective

Provide an overview of emerging Rich Client technology options that are available.

Agenda

- Introducing the Rich Client
- The AJAX Options
- Mozilla's XUL
- Flash Platforms: *OpenLaszlo* & *Adobe Flex*
- Other Rich Clients Options
- Summary
- Resources
- Q/A

The Rich Client Movement

By 2010, 30% of new projects will be based on rich internet clients and 80% of those will be AJAX-driven

- Gartner Research Group (2006)

What Is A Rich Client?

- To answer, we must look at client/server architecture
- Traditional Client/Server Model
  - Provides a rich set of GUI components
  - A granular processing of local event cycles
  - Seamless connection to back-end data source
  - Large binaries installed on each client desktop
  - No middle tier server for business components
  - Little (or no) business component re-use
  - Costly and difficult update/deployment cycles
  - Proprietary and little standards

The Rich Client Proposition

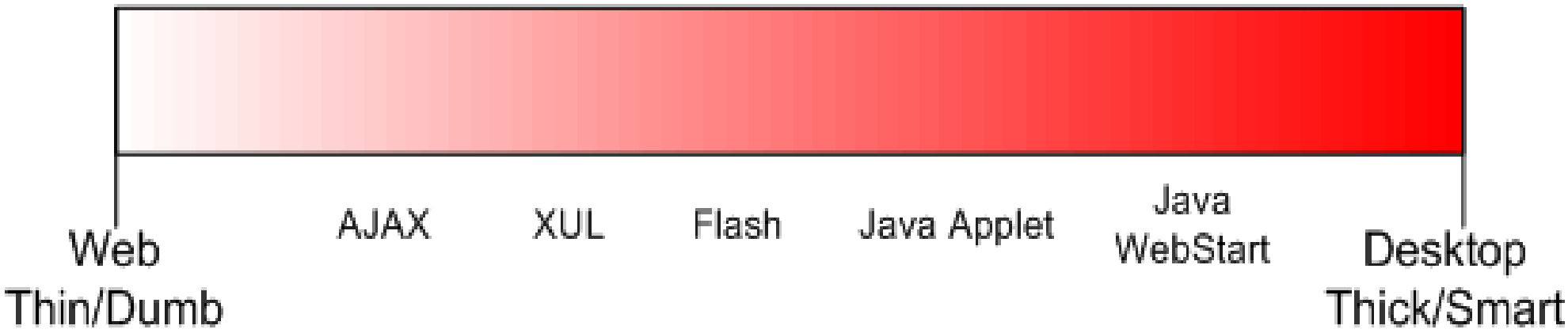
- Evolution of Client/Server
- A Multi-tiered Architecture
  - Highly productivity and interactivity
  - Rich set of GUI components
  - Granular processing local event cycles
  - Seamless connection to middle tier *service layer*
  - Reusable middleware business components
  - Zero (or little) deployment / update costs
  - GUI layout, data binding, and validation services

Rich Client Technology Tiers

- Highly decoupled technology tiers
- View tier only handles GUI concerns
  - Component event cycles
  - Data binding and data validation
- Middle / Business Tier
  - Adopt a service architecture
  - Expose a well-defined service points
  - Use standards including Xml-based SOAP / REST
  - Decouple service points from client technology

Rich Client Technology Spectrum

- Browser-based thin clients
  - Traditional web-based apps, *Ajax*-based apps, and Mozilla's *XUL*
- Browser-hosted (Plug-in)
  - Java Applets and Flash-based
- Browser-Deployed
  - Java WebStart
- Stand-Alone Desktop-hosted

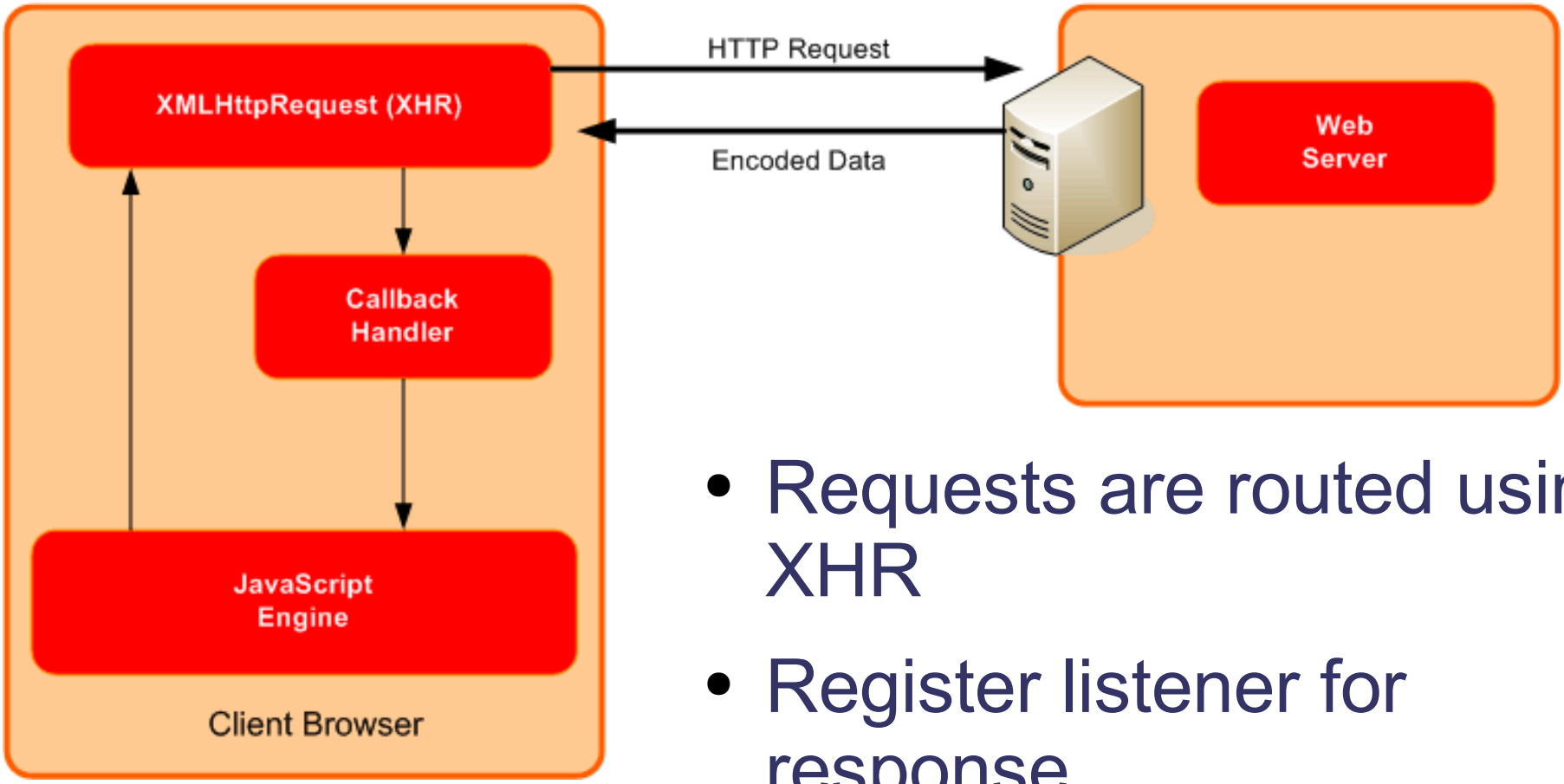


AJAX

AJAX

- Asynchronous JavaScript And XML (AKA DHTML)
- Pattern for Web-Based Development
  - Use standards: *HTML*, *JavaScript*, *XML*, *CSS*, *DOM*
  - Asynchronous requests through *XMLHttpRequest*
- Browser-Based Technology
  - Runs on modern browsers: *Firefox*, *Opera*, *IE*, *Safari*
  - Application rendered within web browser
  - Supports for data-binding
  - Zero-Install / Zero-Update costs on client device
  - Security constraints – no access to local storage

AJAX's XMLHttpRequest



- Requests are routed using XHR
- Register listener for response
- May be synch'd or asynch'd

Taking A Look

```
<script>
function sayHello() {

   var label = document.getElementById("label");
   label.style.backgroundColor = "red";
   msg = "Hello script”;

   var name = document.getElementById("name");


   request = new XMLHttpRequest();

   request.open(“GET”, “some/url/page.htm?msg=”+msg , true);

   request.onreadystatechange=function(){
       if(request.readyState==4){
          label.innerHTML = request.responseText;
       }
   };
}
</script>
```

Adopting AJAX

- AJAX space is crowded
- Several adoption strategies
  - Homegrown – write your own JavaScript solution
  - Functional abstraction frameworks
    - Abtracts browser events, data-binding, and browser implementation differences
  - Widget frameworks
    - Rich GUI components, windowing, and animations
  - Server-side frameworks
    - Ajax Code generated at runtime and push to client
  - Client- and server-side frameworks

AJAX Frameworks

- Popular Open Source Frameworks
  - *Prototype*, *Scriptaculous*, *Rico*, *Dojo*
    - Provides abstractions, widgets, animations
    - Great place to start exploring AJAX
  - Yahoo API: *Mapping, searching, YUI widgets*
  - Google API:*Mapping, searching, calendering, Google Web Toolkit, and data processing through Google data*
  - DRW: *JavaScript data-remoting API for Java*
  - Zimbra: Up and coming, great component model
  - *And much more ...*

AJAX Code Compilers/Emitters

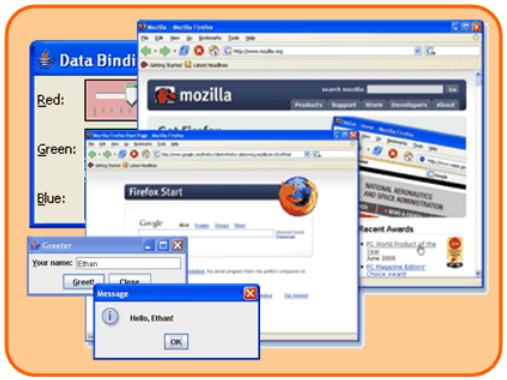
- Generated JavaScript
  - Use other languages to generate JavaScript
  - Provide JavaScript abstraction for windowing, widgets, animation, data-remoting and event handling.
- Frameworks
  - Echo2: Server-side Java
  - Google Web Toolkit: Compiles Java into JavaScript
  - Java Server Faces: Server-side Java
  - OpenLaszlo Legals (featured): Tag-based XML language

AJAX Considerations

- JavaScript skills are required / learning curves
- Knowledge of HTML, CSS, and DOM are necessary.
- Start simple / start small (try Prototype / Open Rico)
- No standard component models / risk of lock-in
- Debugging may be an issue
- Separate code maintenance / management required
- Both open source and commercial options available
- IDE support non-existent for most open source options
- Rich set of visual web services:
  - Mapping, calendaring, advertisement, video, content, etc
- Excellent platform for composite applications (mashups)

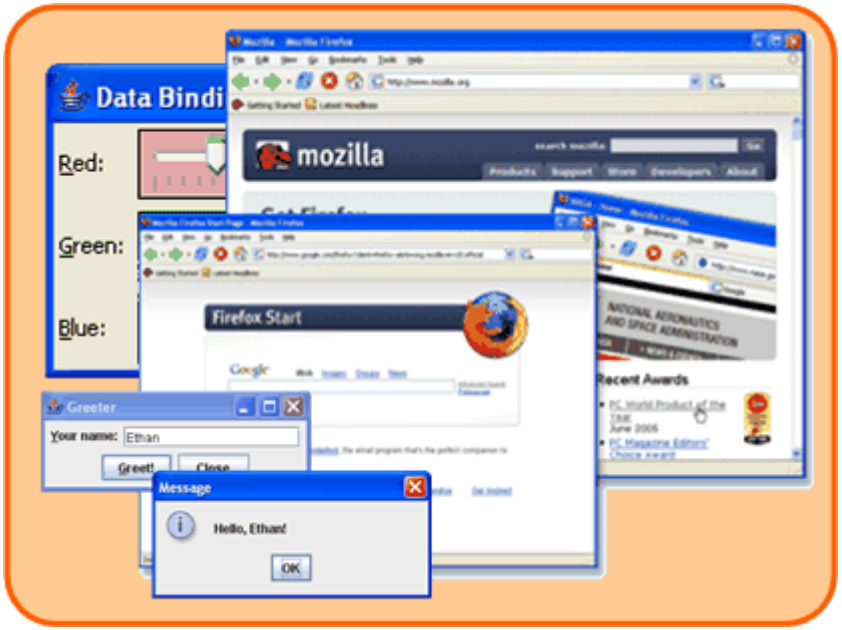
Demo

XUL


Data Bindi
Red:
Green:
Blue:
mozilla
Firefox Start
Greeter
Your name:
Ethan
Greet!
Close
Message
Hello, Ethan!
OK
Recent Awards

XUL

- eXtensible User-interface Language
- Build GUI declartively using XML
  - Specifies screen element layout
  - Style through CSS
  - Provides scripting language
  - Facilitates data binding
  - Abstraction of event management



- Compiled XML
  - Compiled into executable for a runtime engine
  - Runs as stand-alone or hosted in a browser
- Diverse Family of Runtime Environments
  - Mozilla Browser, Flash, Java, and MS.Net.

XUL On Mozilla

- Extension of Standard Web Technologies
  - Uses JavaScript, CSS, DOM, and XML
  - *Native look-and-feel* components:
    - *Menus, popup menus, lists, trees, grids, panels and tabs*.
- Proven
  - Firefox and Thunderbird are XUL applications
- Support Browser-Based Deployment
  - XUL files downloaded and rendered within browser
  - Zero-Install / Zero-Update costs
- Support Stand-Alone Deployment
  - XULRunner runtime tools
  - Installation and upgrade management

Taking A Look

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="chrome://global/skin" type="text/css"?>
<?xml-stylesheet href="xul.css" type="text/css"/?>


<window title="Client Management" onload="resetScreen()"
xmlns="http://www.mozilla.org/keymaster/gatekeeper/there.is.only.xul">
    <script type="text/javascript" src="../js-lib/utils.js"></script>
    <groupbox>
    <hbox>
        <button id="saveClient" label="Save"/>
        <button id="newClient"  label="New Client"/>
     </hbox>
    </groupbox>
</window>
```

XUL: Other Family Members

- Java
  - JAXX: compiled into Swing
  - SwiXml: compiled / Applet runtime
  - Thinlet: Applet runtime
  - Canoo's ULC XML: Applet runtime
- MS XAML – .Net Windows Forms
- Flash (featured)
  - Flex
  - OpenLaszlo

XUL Considerations

- No standard XUL implementations / risk vendor lockin
- Requires learning new XML language constructs
- Documentation maybe an issue for non-commercial offerings
- Minimal open source IDE support
- Select XUL platform closer to targeted runtime
- Mozilla XUL
  - Leverage knowledge of web technologies
  - Plug-in architecture for feature scaling
  - Skinable look-n-feel
  - Runtime available for several OS's (Windows, Linux, Mac)
  - More features in Firefox = more features for your apps

Flash
Rich Client Platforms

Flash

- Vector Multimedia Runtime
  - Produced by Adobe (obtained from Macromedia)
  - Compressed small vector file format (swf)
  - From time-based movie paradigm to desktop app model
  - Support for multimedia assets and animated components
- Flash-Based Rich Applications
  - Desktop-like GUI with discrete component event cycles
  - *Browser-hosted:* Zero-Install / Zero-Update costs
  - Browser plug in with large install base (97%)
  - Ability for stand-alone deployment (via Flash Player)
  - Abstraction of event handling, animation, and data services
  - Rich visual components: trees, menus, sliders, grids, etc

Flash Rich Client Platforms

OpenLaszlo

- Open source project from Laszlo Systems
- Generates Flash binaries
- New version generates AJAX
- Laszlo Platform
  - Cinematic GUI widgets
  - OO-based declarative XML language: LZX
  - Scripted with ECMAScript (standard JavaScript)
  - *Customizable controls*: data grid, tree, menus, tabs
  - Web-based runtime and admin tools

OpenLaszlo Tools

- Open Laszlo Framework
  - Full SDK (written in Java)
  - Runtime compilation of LZX into
    - Flash binaries (version 7, 8, and 9)
    - AJAX / DHTML files
    - Scalable Vector Graphics (SVG, in early phase)
    - Plans for others
- Laszlo Services
  - Just-in-time compilation via Laszlo Servlet
  - *Data Services:* caching, optimization, data remoting
  - Inherent support of SOAP, XML-RPC, JavaRPC
  - Built-in security service

Laszlo Considerations

- Leverage knowledge of JavaScript
- Entire technology stack is open source / free
- Excellent documentation / tutorials / training available
- Laszlo Legals targets multiple runtime engines
- Provides comparable features as Flex for free
- Default GUI components not crisp / dated look
- Extensible object-oriented API and components
- No Standard CSS model
- Slow, but growing adoption
- No third party component market place
- Not standard / risk of vendor lockin

Introducing Flex

- Adobe Flex from Adobe System Software
- Commercial Platform
  - SDK (only tool free)
  - Flex Builder IDE (Eclipse-based visual tool)
  - Flex Charting
  - Flex Data Services
- Flex Platform
  - MXML: XML-based language for GUI layout
  - Scripted with full OO ActionScript (ECMAScript)
  - *Extensible components*: data grid, tree, menus, tabs
  - Implementation of CSS for GUI styling

Flex Tools

Adobe Flex

- Flex Framework
  - Compile-ahead model – then deploy
  - MXML/ActionScript Compilation
    - SWF binaries only
    - ActionScript 3 compiled to native code at runtime
- Data Services
  - RCP / WebService proxies
  - Data synchronization
  - Messaging for ESB integration
  - Just-in-time compilation through URL requests

Flex Considerations

- Easy to get started with SDK
- Extensive documentation/ tutorials / training
- Leverage of Adobe and Macromedia mindshare
- Support for enterprise environment
- MXML is a simpler abstraction of GUI model
- GUI components are nice and modern
- Yahoo Map Flex component available
- Entire technology stack is not free (only SDK)
- Data Service may be costly
- Targets only Flash runtime environments

Demo

Other Rich Client Options

Eclipse RCP

- Based on Eclipse IDE
- Desktop Application Framework
- Java-Based Plugin Model
- Built-In Updater
- GUI Based on Non-Standard SWT
- Uses Native Widgets
- Well-Designed Component Model
- Extended Event Model

Eclipse Considerations

- Runs on all popular OS's
- Great IDE support
- Large mindshare and community support
- Has support from numerous commercial and open source backers
- SWT is not standard and can present moderate learning curve
- GUI component model vastly different from standard Swing
- Expected native widget behaviors may differ

NetBeans Platform

- Based on NetBeans IDE
- Desktop Application Framework
- Java-based Plugin Model
- GUI Uses Standard Swing
- Built-in update mechanism

NetBeans Considerations

- Uses standard Java
- Great IDE Support
- Leverage knowledge of Swing
- Simple application component framework
- Increasing popularity and acceptance
- Support Java Web Start

Conclusion

Summary

- Support for web-based rich clients is growing fast and there are plenty of tools available.
- Select a technology that best target your environment.
- Only implement rich client where necessary. If site is text driven, then use HTML, if form or requires interactivity, then consider rich client solutions.
- AJAX-based solutions can target the largest user base since only a browser is required.
- Flash solutions are gaining popularity but should be tested first before adoption.
- While open source options are plenty, also consider your commercial options if applicable.

Resources

- AJAX
  - Rico - http://openrico.org/
  - Dojo - http://dojotoolkit.org/
  - Yahoo - http://developer.yahoo.com/
  - Google - http://code.google.com/
- XUL - http://developer.mozilla.org/
- OpenLaszlo - http://openloaszlo.org
- Adobe Flex - http://www.adobe.com/products/flex/
- Eclipse - http://www.eclipse.org/rcp/
- NetBeans Platform - http://www.netbeans.org/products/platform/
- Vladimir Vivien Blog - http://vladimirvivien.com/blogs/ot/