JavaScript and AJAX Accessibility

Becky Gibson
Web Accessibility Architect

Agenda

- **What is AJAX and why do I care?**

- **What are the Issues with AJAX?**

- **How do I make my AJAX applications Accessible?**

  – Using HTML Techniques

  – Using DHTML Accessibility Techniques

- **Demonstration of techniques**

- **Summary**

What is AJAX?

- **AJAX = Asynchronous JavaScript and XML**

- **Allows incremental update of Web pages.**

- **Built using standard web technologies - HTTP, (X)HTML, CSS, JavaScript, Document Object Model (DOM), XML**

- **Used by many popular internet companies.**

- **Examples:**

  - Google Suggests, Google & Yahoo! Maps

  - Amazon A9 Search

  - Flickr, BaseCamp, Kayak
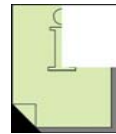
  - Yahoo! AJAX Library

Traditional Web Interaction

Client makes http request

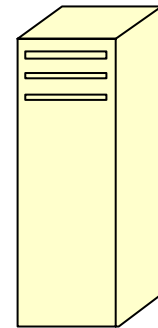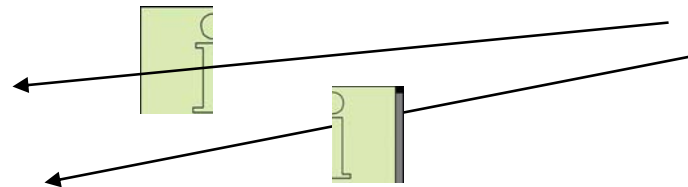Server returns a new page
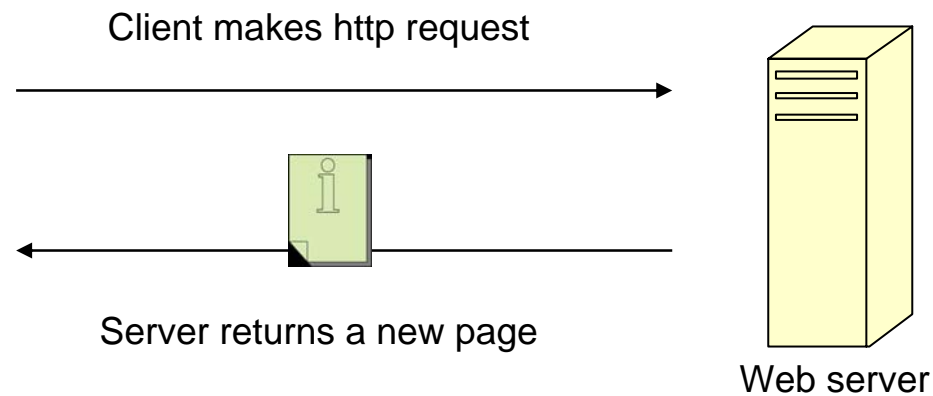
Web server

How AJAX works

Client makes http request for specific information

Server returns the requested information

Web server

Multiple requests are served

Why Do I Care About AJAX?

- **Enables building Rich Internet Applications (RIA)**

- **Allows dynamic interaction on the Web**

- **Improves performance**

- **Real-time updates**

- **No plug-ins required**

What are the Issues with AJAX?

- **User does not know updates will occur.**

- **User does not notice an update.**

- **User can not find the updated information.**

- **Unexpected changes in focus.**

- **Loss of Back button functionality*.**

- **URIs can not be bookmarked*.**

*These issues will not be discussed as they are not specific to accessibility.

Specific Accessibility Issues

- **Assistive Technology users are not aware of updates**

  – Updates occur on a different section of the page than the user is currently interacting with.

  – Clicking a link updates a different section of the page.

  – Auto-complete fields or generated options not available to assistive technology.

  – User has no idea how to find new or updated content

  – Changes in focus prohibit complete review of the page

  – Changes in focus cause disorientation and additional navigation.

Informing the User

- **Explain the interaction to the user**
  - Before accessing the AJAX enabled page
  - Within the AJAX enabled page
- **Where possible, provide a manual update option**
  - Necessary for WCAG 2.0 Guideline 2.2
- **Save the user's update preference**

Make updates Noticeable

- **Change the background color of updated data**
  - Use a subtle color
  - Change only for a few seconds
  - Best for small areas of the page
- **Briefly blink the updated data**
  - Blink for 3 seconds or less (WCAG 2.0 Guideline 2.2)
  - Avoid the flash threshold (WCAG 2.0 Guideline 2.3)

Help Users Find Updated Information

- **Provide option for updates via an Alert**

- **Provide option to set focus to new data.**

- **Use HTML header tags to mark sections with updated content.**

- **Use DHTML Accessibility Alert role in conjunction with a floating pop-up box.**

- **Use DHTML Accessibility Description role to describe new content.**

DHTML Accessibility Techniques and AJAX

- **DTHML Accessibility allows the creation of desktop style widgets on the web**

- **Full keyboard navigation without excessive use of the tab key**

- **Allows easy navigation from component to component and to updated content.**

- **Allows non-intrusive notification of updated content via Alert**

DHTML Accessibility Overview

- **Add keyboard event handling**

- **Add role and state information**

- **Make items focusable via tabindex attribute**

- **Update state information dynamically**

DHTML Accessibility Overview (continued)

- **Works in HTML or XHTML**

  - XHTML

    - Uses XHTML 1.x role attribute
    - Uses namespaces to add state attributes

  - HTML

    - Embeds role and state information into class attribute
    - Accessibility Script library sets the role and state using DOM apis

Example using XHTML

```
<html lang="en" xml:lang="en" xmlns="http://www.w3.org/TR/xhtml2"
xmlns:wairole="http://www.w3.org/2005/01/wai-rdf/GUIRoleTaxonomy#"
xmlns:waistate="http://www.w3.org/2005/07/aaa">


………..


<span id="slider" class="myslider myselector2"
role="wairole:slider"
waistate:valuemin="0"
waistate:valuemax="50"
waistate:valuenow="33">
</span>
```

Example Using HTML

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd" >
<html lang="en">
<head>
<script type="text/javascript" src="enable.js"></script>
</head>
………………..

<span id="slider" tabindex="0"
class="myslider myselector2
axs slider valuemin-0 valuemax-50 valuenow-33" >
</span>
```

Technique described at:

http://www.w3.org/WAI/PF/adaptable/HTML4/embedding-20060318.html

Summary

- **AJAX is a powerful technology for making dynamic web applications.**

- **AJAX applications can present Accessibility barriers.**

- **Simple changes to application design can make AJAX usable for all.**

- **DHTML Accessibility Techniques fit well with AJAX**

Questions?