

1 ☐ Technical Aspects of Usability: How Developers can Improve the User Experience

Jeff Lash

St. Louis Web Developer's Association

September 20, 2001

2 ☐ Outline

- Usability at the site level
 - The #1 usability problem -- “The Ws”
 - Error messages
 - Improving Searching
 - Log Analysis

3 ☐ Outline

- Usability at the file system level
 - Removing filenames
 - Directory browsing

4 ☐ Outline

- Usability at the page level
 - Breadcrumbs
 - Conditional Navigation
 - Sitemap on every page
 - Print/email this page
- Other elements of the user experience
 - Mailing List Usability

5 ☐ The #1 Usability Problem

- What is the worst usability problem?
 - Most usability problems occur on your site itself, but this happens when people can't even get to your site.

6 ☐ Yale -- no Ws, does not work

7 ☐ Yale -- with Ws, works

8 ☐ “The Ws” (Path of Resistance)

- Serious usability problem if people can't even get to your site
- Bad: only works with or without Ws

- Better: works with or without Ws
- Best: works with or without but automatically removes Ws

9 ☐ Removing “The Ws”

- Why?
 - No longer necessary
 - People not confused when they’re removed
 - Four fewer characters to deal with
 - Browsers interpret Ws/no Ws differently (i.e. for visited links)
 - Consistent domain for advanced scripting
 - Stats packages treat as different domains

10 ☐ Removing “The Ws”

- How?
 - In Apache

```
<VirtualHost 123.45.67.890>
ServerAlias www.domainname.com
Redirect permanent / http://domainname.com/
</VirtualHost>
```
 - All URLs within domain will work with or without Ws

11 ☐ Error messages

- Bad: Using the server default pages
 - They do not provide any helpful or site-specific information

12 ☐ Google 404 page

13 ☐ Error messages

- Bad: Customizing error page with branding elements but nothing useful
 - Better than server default but not any more helpful for users

14 ☐ Amazon 404 page

15 ☐ Error messages

- Better:
 - Including the following elements:
 - Search box
 - Sitemap
 - Most popular pages

16 ☐ Microsoft 404 page

17 ☐ Error messages

- Best:
 - Including the following elements:
 - Most popular pages
 - Email link to webmaster or suggestion form
 - Common mistakes
 - Suggest related pages

18 ☐ Useit 404 page

19 ☐ TBid 404 page

20 ☐ Tbid 403 page

21 ☐ Tbid 401 page

22 ☐ Tbid 500 page

23 ☐ Error messages

- Best
 - Include the following elements:
 - Do not list specific error number
 - Keep URL that gave error

24 ☐ Error messages

- Even better -- technical tricks
 - Automatically send an email (or log to a file/database) when an error is encountered
 - Track referring page, error page, browser info, IP address
 - Inform the user of what happened in plain English, allow them to email or fill out a form to provide more information

25 ☐ Error messages

- In PHP

```
<? $referer = getenv("HTTP_REFERER");
$remaddr = getenv("REMOTE_ADDR");
$browser = getenv("HTTP_USER_AGENT");
$uri = getenv("REQUEST_URI");
$headers = "From: 404 Page <404@yoursite.com>\n";

mail("404@yoursite.com",
"404 error",
"There is a bad link on this page: $referer
Someone was trying to go to: http://yoursite.com$uri
Thank the user at $remaddr
who uses $browser", $headers);
?>
```

26 ☐ Error messages

- Even better -- for Intranet:
 - Find out what user generated error
 - Windows network: "LOGON_USER" server variable
 - Or detect the IP address
 - In a small office, you can put together an IP address --> Employee array or database

- Detect the user, grab page info, forward to a real person

27 ☐ Improving Searching

- Search pages
 - Have search box on every page
 - Search entire site by default; allow users to choose specific sections
 - Have an advanced search page
 - Don't push people there by default
 - Have regular and advanced search on that page

28 ☐ Improving Searching

- Search results
 - Analyze logs: See what people are searching for and make sure your results are appropriate
 - Learn how your engine works: Can you change the rankings manually, use special tags, alter algorithm?
 - Manually insert results; differentiate them from “normal” results

29 ☐ Improving Searching

- Search results
 - Are people trying to search “the web”? Are they trying to search for a specific technical term? Give them options on the results page
 - Lots of searches for the same terms may indicate a faulty Information Architecture
 - Place often-searched terms on your home page (or most viewed pages) and see how/if your search logs change

30 ☐ Log Analysis

- Logs can provide multitudes of data about your site, but the trick is reading them correctly
- For example:
 - What pages are being seen the most? What is the top entry page? Does it reflect your site/brand? Does it explain the purpose of your company/site?

31 ☐ Log Analysis

- Logs provide great data about how users come to your site through search engines
 - Which ones drive the most traffic?
 - What terms are being used most often?
 - What pages/results are being used most?

32 ☐ Log Analysis

- “External Search Usability Test”
 - Have user go to their preferred search engine
 - Give them generic topic to search for
 - See what words they use to search
 - Results:
 - Does your site come up?
 - Does the user go there? Why or why not?

- Does going to your top result provide answer?

33 ☐ Removing filenames

- Why?
 - Conventions change, operating systems change, CM systems change
 - Allows you to change backend without affecting the user experience
 - Assures links from other sites will work even if you make changes

34 ☐ Removing filenames

- Why?
 - URLs are part of the user experience
 - Helps users orient themselves
 - Forces you to use better file structure and organize directories better
 - Helpful for new developers working on project
 - Allows you to incorporate effective directory browsing into your site design

35 ☐ Directory browsing

- BAD: Disallowing it
 - Why?
 - Users can be smart and may want to explore.
 - There are better ways to approach it than turning it off at the server level.

36 ☐ MaxJazz 1

37 ☐ MaxJazz 2

38 ☐ MaxJazz 3

39 ☐ Directory browsing

- Better: Allowing it, but using server defaults
 - Could be bad if you don't want contents to be viewable

40 ☐ Boulder

41 ☐ Directory browsing

- Best
 - If you want directory to be off-limits...
 - Put branded index/default page in the folder
 - Redirect errors to your common error page
 - Don't use a redirect

42 ☐ Xplane nopeeking

43 ☐ Directory browsing

- Best
 - If you want directory to be browsable...
 - Incorporate that concept into your site design
 - Use one script to drive the browsing and allow users to navigate within the directories

44 ☐ Olga directory browsing

45 ☐ Directory browsing

- Benefits
 - Easy to update
 - Familiar concept for users
- With script-based solution, users can browse directories without you having to turn directory browsing on at the server level
- Good rule of thumb is to only turn directory browsing on for directories where you know you'll need it

46 ☐ Breadcrumbs

- What are they?
 - Show users the hierarchical path
 - Not necessarily the path they used to get there
- Why use them?
 - Help users understand where they are and where they can go

47 ☐ Yahoo breadcrumbs

48 ☐ Ancestry breadcrumbs

49 ☐ Msdn breadcrumbs

50 ☐ Breadcrumbs

- Important Aspects
 - Easiest way is to already have a good directory structure in place
 - URLs can be breadcrumbs
 - Users are savvier; technological advances like Google Toolbar make it easier for users to traverse directories
 - You can print the URL but it's easier to convert it to "normal English"

51 ☐ Breadcrumbs

- Important Aspects
 - "Normal English" doesn't need to rely on directory structure
- Implementation
 - JavaScript
 - ASP/PHP
 - Grab URL and convert
 - Read page title and print on page

52 ☐ Conditional Navigation

- What is it?
 - Navigation that understands your location and adapts itself accordingly

53 ☐ Camworld condnav1

54 ☐ Camworld condnav 2

55 ☐ Evhead condnav

56 ☐ Conditional Navigation

- Why?
 - People get confused and don't realize where they are
 - Usually exacerbated by other site problem (i.e. Bad information architecture)
 - Remember: People want to know where they are and where they can go
- Works best with good breadcrumbs

57 ☐ Conditional Navigation

- How?
 - Manually
 - Accurate but time-consuming
 - Using includes
 - Easy to update but don't on their own support conditional statements

58 ☐ Conditional Navigation

- How?
 - Scripting (ASP/PHP)
 - Set a page title variable on each page
 - Include all page titles on template/script page
 - Use if/then
 - If page title on content page matches page title in template, then don't print a link to that page
 - Problems
 - You have to explicitly declare page titles and syntax must match exactly

59 ☐ Conditional Navigation

- How?
 - Scripting (ASP/PHP)
 - You can also detect the URL rather than declaring page titles
 - You can put all the pages in an array
 - When it matches the current page, set a flag
 - All of the links on the page will be the same except for the flagged link

60 ☐ Sitemap on every page

- Why?
 - Not taking up screen real estate
 - Users are willing to scroll, especially if they know what is lower down on the page
 - People use them

61 ☐ Sitemap -- poorbuthappy

62 ☐ Xplane sitemap bottom

63 ☐ Xplane sitemap top

64 ☐ Sitemap on every page

- How?
 - Create a modified (condensed/abridged) sitemap and include it on every page

- Possibly add conditional navigation as well
- Track use of it
 - Change links according to the most popular categories
 - Test users' mental models of the site and alter categories accordingly

65 ☐ Print/Email this page

- How would you like printable versions of all your pages to automatically be created?
- If your site is built on templates, it's very easy

66 ☐

67 ☐ Xblog -- printable

68 ☐ Print/Email this page

- Print this page: How?
 - In PHP:

- In template:

```
<? if(strpos($format, "print")) {
  include "/home/yoursite/templates/top_print.php";
} else {
  include "/home/yoursite/templates/top.php";
} ?>
```

- On Page

```
<a href="<? print "$uri" ?>format=print" title="Printer-friendly formatting for this page.">
```

69 ☐ Print/Email this page

- Email this page: How?
 - In PHP:

- On page:

```
<? $uri = getenv("REQUEST_URI") ?>
<a href="javascript:recommend
('/recommend.php?link=http://yoursite.com<? print "$uri" ?>&title=<? print "$title" ?>')">
```

- In mailscript:

- Print Slink and Stitle in email
- Good (free) script at
- <http://www.snipe.net/geek/scripts/tellfriend.phps>

70 ☐ Xblog -- email this page

71 ☐ Print/Email this page

- Email this page: things to remember
 - Popups are okay since users may be more likely to want to return to page
 - Disclose your privacy policy
 - Log usage for tracking purposes
 - Send “Thank you” email
 - User experience extends into email
 - Most people don't use them

72 ☐ Mailing list usability

- Make it easy to subscribe/unsubscribe

- Tell users what email they're subscribed as
- Give users multiple ways to unsubscribe
 - Via email
 - Have dedicated email address
 - Allow "unsubscribe" to be in body or subject line
 - Any attempts at unsubscribing that the list software doesn't process should be forwarded to a real person
 - Via URL
 - Page should have info about what lists person is on and what they will be unsubscribed from

73 ☐ Mailing list usability

- Make it easy to subscribe/unsubscribe
 - Reply -to address should go somewhere
 - To a real person
 - To an auto-respond with list information
 - HTML, plain text, don't know?
 - Use JavaScript to detect known domains that can support HTML (aol.com, hotmail.com, etc.)
 - "Don't know" should be default and used if blank -- always subscribe these as plain text

74 ☐ Closing

- Feel free to email me with questions, comments, or suggestions
jeff@jefflash.com

- Slides and notes should appear on the stlwebdev.org website