Content Repurposing for Small Devices

Neil C. Rowe
U.S. Naval Postgraduate School, USA

INTRODUCTION

Content repurposing is the reorganizing of data for presentation on different display hardware (Singh, 2004). It has been particularly important recently with the growth of handheld devices such as personal digital assistants (PDAs), sophisticated telephones, and other small specialized devices. Unfortunately, such devices pose serious problems for multimedia delivery. With their tiny screens (150 by 150 for a basic Palm PDA or 240 by 320 for a more modern one, vs. 640 by 480 for standard computer screens), one cannot display much information (i.e., most of a Web page); with their low bandwidths, one cannot display video and audio transmissions from a server (i.e., streaming) with much quality; and with their small storage capabilities, large media files cannot be stored for later playback. Furthermore, new devices and old ones with new characteristics have been appearing at a high rate, so software vendors are having difficulty keeping pace. So some real-time, systematic, and automated planning could be helpful in figuring how to show desired data, especially multimedia, on a broad range of devices.

BACKGROUND

The World Wide Web is the de facto standard for providing easily accessible information to people. So it is desirable to use it and its language—HTML—as a basis for display for small handheld devices. This would enable people to look up ratings of products while shopping, check routes while driving, and perform knowledge-intensive jobs while walking. HTML is, in fact, device-independent. It requires the display device and its Web-browser software to make decisions about how to display its information within guidelines. But HTML does not provide enough information to devices to ensure much user-friendliness of the resulting display: It does not tell the browser where to break lines or which graphics to keep collocated. Display problems are exacerbated when screen sizes, screen shapes, audio capabilities, or video capabilities are significantly different. Microbrowser markup languages like WML, S-HTML, and HDML, which are based on HTML but designed to better serve the needs of small devices, help, but these only solve some of the problems.

Content repurposing is a general term for reformatting information for different displays. It occurs frequently with content management for an organization's publications (Boiko, 2002), where content or information is broken into pieces and entered in a repository to be used for different publications. However, a repository is not cost-effective unless the information is reused many times, something not generally true for Web pages. Content repurposing for small devices also involves real-time decisions about priorities. For these reasons, the repository approach often is not used with small devices.

Content repurposing can be done either before or after a request for it. Preprocessing can create separate pages for different devices, and the device fetches the page appropriate to it. It also can involve conditional statements in pages that cause different code to be executed for different devices; such statements can be done with code in JavaScript, PHP embedded within HTML, or more complex server codes using such facilities as Java Server Pages (JSP) and Active Server Pages (ASP). It also can involve device-specific planning (Karadkar, 2004). Many popular Web sites provide preprocessed pages for different kinds of devices. Preprocessing is cost-effective for frequently needed content but requires setup time and can require considerable storage space, if there is a large amount of content and ways to display it.

Content repurposing also can be either client-side or server-side. Server-side means a server supplies repurposed information for the client device; client-side means the device itself decides what to display and how. Server-side repurposing saves work for the

Copyright © 2005, Idea Group Inc., distributing in print or electronic forms without written permission of IGI is prohibited.

device, which is important for primitive devices, and can adjust to fluctuations in network bandwidth (Lyu et al., 2003) but requires added complexity in the server and significant time delays in getting information to the server. Devices can have designated proxy servers for their needs. Client-side repurposing, on the other hand, can respond quickly to changing user needs. Its disadvantages are the additional processing burden on an already-slow device and higher bandwidth demands, since information is not eliminated until after it reaches the device. The limitations of small devices require most audio and video repurposing to be server-side.

METHODS OF CONTENT REPURPOSING

Repurposing Strategies

Content repurposing for small devices can be accomplished by several methods, including panning, zooming, reformatting, substitution of links, and modification of content.

A default repurposing method of Internet Explorer and Netscape browser software is to show a window on the full display when it is too large to fit on the device screen. Then the user can manipulate slider bars on the bottom and side of the window to view all the content (pan over it). Some systems break content into overlapping tiles (Kasik, 2004), precomputed units of display information, and users can pan only from tile to tile; this can prevent splitting of key features like buttons and simplifies client-side processing, but it only works for certain kinds of content. Panning may be unsatisfactory for large displays like maps, since considerable screen manipulation may be required, and good understanding may require an overview. But it works fine for most content.

Another idea is to change the scale of view, zooming in (closer) or out (further). This can be either automatic or user-controlled. The MapQuest citymap utility (www.mapquest.com) provides user-controlled zooming by dynamically creating maps at several levels of detail, so the user can start with a city and progressively narrow on a neighborhood (as well as do panning). A problem for zooming out is that some details like text and thin lines cannot be shrunk beyond a certain minimum size and still remain legible. Such details may be optional; for instance, MapQuest omits most street names and many of the streets in its broadest view. But this may not be what the user wants. Different details can be shrunk at different rates, so that lines one pixel wide are not shrunk at all (Ma & Singh, 2003), but this requires content-specific tailoring.

The formatting of the page can be modified to use equivalent constructs that display better on a destination device (Government of Canada, 2004). For instance, with HTML, the fonts can be made smaller or narrower (taking into account viewability on the device) by font tags, line spacing can be reduced, or blank space can be eliminated. Since tables take extra space, they can be converted into text. Small images or video can substitute for large images or video, when their content permits. Text can be presented sequentially in the same box in the screen to save display space (Wobbrock et al., 2002). For audio and video, the sampling or frame rate can be decreased (one image per second is fine for many applications, provided the rate is steady). Visual clues can be added to the display to indicate items just offscreen (Baudisch & Rosenholtz, 2003).

Clickable links can point to blocks of less important information, thereby reducing the amount of content to be displayed at once. This is especially good for media objects, which can require both bandwidth and screen size, but also helps for paragraphs of details. Links can be thumbnail images, which is helpful for pages familiar to the user. Links also can point to pages containing additional links so the scheme can be hierarchical. In fact, Buyukkoten et al. (2002) experimented with repurposing displays containing links exclusively. But insertion of links requires rating the content of the page by importance, a difficult problem in general (as discussed later), to decide what content is converted into links. It also requires a careful wording of text links since just something like "picture here" is not helpful, but a too-long link may be worse than no link at all. Complex link hierarchies also may cause users to get lost.

One also can modify the content of a display by just eliminating unimportant or useless detail and rearranging the display (Gupta et al., 2003). For instance, advertisements, acknowledgements, and horizontal bars can be removed, as well as JavaScript code and Macromedia Flash (SWF) images, since most are only decorative. Removed content need not

4 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/content-repurposing-small-devices/17235

Related Content

Computer Simulations and Scientific Knowledge Construction
Athanassios Jimoyiannis (2011). *Gaming and Simulations: Concepts, Methodologies, Tools and Applications (pp. 57-74).*
www.irma-international.org/chapter/computer-simulations-scientific-knowledge-construction/49374

Perceptual Multimedia: A Cognitive Style Perspective
Gheorghita Ghineaand Sherry Y. Chen (2006). *Digital Multimedia Perception and Design (pp. 187-205).*
www.irma-international.org/chapter/perceptual-multimedia-cognitive-style-perspective/8428

Interactive Multimedia Technologies for Distance Education in Developing Countries
Hakikur Rahman (2005). *Encyclopedia of Multimedia Technology and Networking (pp. 447-453).*
www.irma-international.org/chapter/interactive-multimedia-technologies-distance-education/17282

A Scalable Graph-Based Semi-Supervised Ranking System for Content-Based Image Retrieval
Xiaojun Qiand Ran Chang (2013). *International Journal of Multimedia Data Engineering and Management (pp. 15-34).*
www.irma-international.org/article/a-scalable-graph-based-semi-supervised-ranking-system-for-content-based-image-retrieval/103009

Multimedia Authoring: Human-Computer Partnership for Harvesting Metadata from the Right Sources
Brett Adamsand Svetha Venkatesh (2005). *Managing Multimedia Semantics (pp. 223-245).*
www.irma-international.org/chapter/multimedia-authoring-human-computer-partnership/25975