Contributors

Surendar Chandra
University of Notre Dame

0-8493-0052-5/00/$0.00+$.50

Contents

0-8493-0052-5/00/$0.00+$.50

1

Content Adaptation and Transcoding

Surendar Chandra
University of Notre Dame

CONTENTS

In this chapter, we describe content adaptation as a means to customize web objects for constrained environments. We pay particular attention to transcoding as a mechanism to adapt multimedia content. We first describe a number of Internet scenarios wherein the rich web content should be transcoded to more appropriate forms. We discuss some popular transcoding mechanisms. We also discuss mechanisms that consider the collection of objects that form a typical web request. Finally, we describe some of the research challenges involved in utilizing these transcoding technologies.

1.1 Introduction

An explosion in the number and variety of devices is dramatically changing the world of Internet computing. Recent mobile computing devices range in complexity from tablets to palmtops to fully functional jewelry with computational resources. The web is emerging as a preferred technology for ubiquitous data access. The ubiquitous computing future espoused by Mark Weiser (Weiser [1993]) and others call for these connected devices to eventually vanish into the periphery. The cooltown prototype (www.cooltown.org) implements this vision by mapping physical people, places and things to their web presence (Kindberg et al. [2000]. In this model, each object is represented and named by its corresponding URL. Many mobile applications will use the mobile device as a window into vast amounts of data that can be delivered via the Internet, particularly in the form of web content. With the increasing availability of the Internet on a wide range of these devices, the web is emerging as the preferred data delivery mechanism for a number of application scenarios.

Such access dynamics expose a number of inadequacies in the web infrastructure. In the next few sections, we outline some of the issue faced by the clients because of local and network resource constraints as well as constraints at the servers themselves.

1.1.1 Client side constraints

There is a rapid movement toward embedding powerful processors in a large variety of consumer electronic devices. These devices have constraints on the available battery capacity, network, displays and output mechanisms, processing power and local storage. Such constraints severely restrict the quality of web experience for the end users.

0-8493-0052-5/00/$0.00+$.50

- **Networks:** Clients access the Internet using a number of networking technologies; from wired high speed LAN networks, broadband and dialup networks to wireless technologies. Predominant wireless technologies include IEEE 802.11 WiFi LANs (LAN/MAN Standards Committee of the IEEE Computer Society [1999]), 2G and 3G cellular networks. The wide variability in the access latency, connection reliability, available bandwidth and access cost for these network technologies make a single approach for data deliver inappropriate. Accesses from cellular modems using technologies such as GSM typically operate in the range up to 19.2 Kbps, while variants of wireless LAN technologies such as IEEE 802.11 offer bandwidths of tens of Mbps. Mobile networks are also expensive; usually charged by the amount of useful data consumed. For example, in the United States, access costs range in the order of 3-5 cents per kilobyte of data transferred.

  The problem of slow and expensive networks is compounded by the large size of multimedia objects that are becoming such a prevalent part of web content. Studies (All Things Web [1999]) have shown that the average web page is about 60 Kbytes in size. Accessing 60 Kbytes using Cellular Digital Packet Data (CDPD) cellular technologies would take about 3 minutes and cost around 3 dollars. For many users, such access costs are prohibitive.

- **Processing power:** The compression mechanisms used in representing multimedia objects trade-off the amount of compression with the client CPU power requirements to decode an object. In CPU resource constrained mobile devices, the tradeoff can translate to transferring more data (i.e less compression) because of client CPU characteristics. The image may be uncompressed on the servers and the uncompressed images transferred to the device; shifting network capacity as the bottleneck.

- **Displays:** In a typical mobile device, factors such as the screen size and color depth severely restrict the image that could be displayed to an user. Display technologies such as back lit displays can consume significant amounts of energy; but may allow images to be viewed at an apparently better quality than for the same screen with reflective screens. Some devices are also expected to completely lack any display capabilities. Hence, the users may want to view variations of the objects that are better suited for their local display characteristics.

- **Battery capacity:** Mobile device system components such as wireless network interface card (WNIC), displays and processors consume significant amounts of energy. A necessary feature for mass acceptance of a mobile device is acceptable battery duration. Newer hardware improvements are reducing the power consumption of system components such as back-lit displays, CPUs etc. However, WNICs operating at the same frequency band and range continue to consume significant power. For example, a 2.4 GHz IEEE 802.11b Wavelan card (Stemm et al. [1996]; Havinga [2000]; Feeney and Nilsson [2001]) consumes 177 mW while in *sleep* state, 1319 mW while *idle*, 1425 mW for receiving data and 1675 mW for transmitting data. For comparison, a fully operational HP-Compaq iPAQ PDA only consumes 929 mW (Cho [2001]). Future trends in battery technologies alone (along with the continual pressure for further device miniaturization) do not promise dramatic improvements that will make this issue disappear. Hence, it is important to look at techniques to reduce the energy consumed by the network interface. Techniques to reduce the amount of data transferred can be expected to offer significant energy savings (Chandra et al. [2000b]).

In general, there is a huge mismatch between the rich multimedia content available on the World Wide Web and the characteristics of mobile devices that are used to access this web content. Users would like to keep the access wait and cost from becoming unbearable.

1.1.2 Server side constraints

The web is emerging as the primary data dissemination mechanism for a variety of applications. The following important trends are evolving as the economic model for Web sites.

The first trend is toward e-commerce and subscription based services. Sites such as ESPN (espn.com) want to prioritize their consumers based on their subscription status, prior access history and their current status (e.g., a customer with a purchasing history or one with a full shopping cart). To retain their paying customers, these sites need to maintain the quality of service for the preferred customers. In order to convince new users to subscribe, these sites need to provide quality teaser objects. Such teaser objects may include article headlines, abstracts and image thumbnails.

The other trend is towards web hosting services (e.g., Yahoo) which create and maintain web pages on behalf of their customers. The services charge their hosted sites based on the size of the web site and the aggregate bandwidth consumed in a month. One of the major problems encountered by web hosting services is through *flash users*. Flash crowds (sometimes referred as the *slashdot effect*) refers to the sudden upsurge in traffic to a web site because of breaking news events; for example, September 11 attacks on the United States, Victoria Secrets web cast etc. Flash crowds to another web site hosted by the same hosting service can degrade the performance for all the other sites hosted by the service. Current web hosting services use a laissez-faire approach in managing their bandwidth. However, customers of these hosting services will demand performance guarantees, forcing the servers to provide differentiated services so that they can charge the sites, not only for the aggregate bandwidth, but also for guaranteed bandwidth during high-demand peak hours.

In general, a primary goal of a web service is providing low latency access to its contents. However, during times of high demand, this goal is compromised by locally available network bandwidth. Even sites that utilize high capacity networks can be overwhelmed by flash crowds during periods of high demand. In the past, web sites have used ad-hoc solutions to deal with peak loads. For example, when significant news stories break, news sites such as MSNBC and CNN, export a lightweight version of the site with little or no graphics to conserve bandwidth. Caching at local web proxies and content delivery networks (e.g. Akamai) are other traditional techniques to address bandwidth limitations by replicating and careful placement of objects. Sites have also utilized over-provisioning of resources to combat flash crowds. Gratuitous use of over provisioning can lead to wasted resources and may prove too expensive for many applications. Also, much web content is dynamically generated (maps, stock charts, etc.) or un-cacheable (sites selling access to multimedia contents such as images or movies). To fully realize the benefits of the Internet, one needs to solve the fundamental problems exacerbated by these web trends.

1.1.3 Differentiated services to manage resources

In such a scenario, differentiated service can allow the system to provide a more appropriate level of service for a user based on the current operating environment. Differentiated service means that proxies can match object sizes with the network bandwidth available on the last hop to mobile clients. In this way, a network proxy server can provide different versions of the same object to different clients. Traditional human factors research (Nielsen [1993]) has shown that the response time for accessing a resource should be in the one to ten second range for information to be useful. If the response time is longer than this range, the users tend to lose interest and go on to other things. The system server can choose variations of an object such that the objects are served at a uniform latency of less than five seconds based on the type of network link used in accessing the web. Differentiated service can also allow the system to provide better service for certain customers, based on their status and the current network environment (Abdelzaher and Bhatti [1999b,a]; Chandra et al. [2000a]). To summarize, differentiated service enables:

- Web services to customize web objects for mobile users depending on the client and network characteristics.

- Web services to dynamically allocate the available bandwidth among different user classes,
- Subscription services to provide different versions of contents to clients based on customer status (subscriber versus non-subscriber),
- Web hosting services to share their bandwidth for different classes of hosted clients,
- E-commerce sites to allocate their bandwidth to customers who are making a sale,
- Flexibility in redirecting unused preferred resources to non-preferred customers.

The key insight is that allocation of critical network resources should be done dynamically in response to client access patterns. The feasibility of such a differentiated service scheme depends on the availability of a range of variations for the content so that the server can choose the correct variation for the current network operating environment. While the content provider can manually provide a number of different variations for use by the system, an automatic technique may be preferable to allow the system to dynamically adapt to variability in network performance and client characteristics.

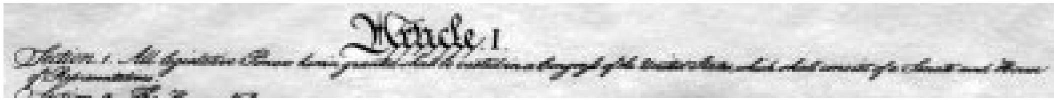
1.2 Transcoding techniques

One promising technique for providing differentiated quality of service is transcoding which can be used to serve variations of the same multimedia object at different sizes. Transcoding is defined as a transformation that is used to convert a multimedia object from one form to another, frequently trading off object fidelity for size. Transcoding operation is a lossy transformation; the user has to explicitly request the original to restore the lost quality. By their very nature, multimedia objects are amenable to *soft access* through a quality-versus-size tradeoff. Transcoding allows web services to transmit variations of the same multimedia object at different sizes, allowing some control over the amount of bandwidth consumed in transmitting a page to a particular client. By some estimates (Ortega et al. [1997]), about 77% of the data bytes accessed across the web are from multimedia objects such as images, audio and video clips. Of these, 67% of the data are transferred for images. Hence, it is important to focus our attention on multimedia data.

Transcoding is a lossy operation; information is usually lost and the transformed object is of lower fidelity. Transcoding operation is frequently applied to fit an object to a certain device (e.g., transcoding an color image to a gray scale image can allow the original image to be viewed in an gray scale monitor). We focus our attention on transcoding to provide differentiated service. In this case, the system intentionally chooses a lower fidelity version, even though the original object can be displayed, albeit violating the service requirements. In the next section, we briefly describe some of the popular transcoding operations for typical web contents available in textual, images and streaming media formats.

1.2.1 Textual content

Textual transcoding can be performed either to fit an object to the current environment or more generally to meet certain service requirements. For example, transcoding can be applied to richly annotated textual formats such as HTML (Figure 1.1(a)) to produce a simpler textual representation



(a) Formatted text

All legislative Powers herein granted shall be vested in a Congress of the United States, which …

(b) ASCII text

FIGURE 1.1
Transcoding a formatted text to other variants (excerpts from the US constitution)

(Figure 1.1(b)). Such transformations can preserve the original content while sacrificing presentation aesthetics. Automatic language translators (e.g. babel fish translations from altavista.com) can also transform the original language of the article.

On the other hand, one can also imagine transformations that generate textual abstract of a longer report. Such operations can be performed using automatic information processing techniques or explicitly by the original content providers themselves. Such transformations are lossy; it is not possible to regenerate the original article from the transcoded version. The user is required to explicitly request the higher quality original.

1.2.2 Image content

Image content forms a rich part of web technologies. Typical images tend to be large. The very nature of image compression and representation formats allow for easier and productive transcoding to various alternative formats. Most image compression and transcoding algorithms are lossy. Popular image transcoding operations include reducing the color depth, reducing the image geometry (i.e. thumbnailing), cropping the irrelevant parts of the image, reducing the compression factor as well as recompressing the object to an alternative format. For example, the various transcodings of the original image of the Earth rising above the lunar horizon (Figure 1.2(a)) to a lower quality image compression factor (Figure 1.2(b)), cropped image (Figure 1.2(c)) as well as a thumbnail (Figure 1.2(d)) are illustrated in (Figure 1.2). Each such transformations offer varying levels of size savings. In the web, progressive format can allow clients to actively participate in the quality choosing process by terminating the transfer once the required image fidelity is received.

Note that transformations such as thumbnailing are easier to generate automatically than the correct cropping operation. The range of transcoded versions also depends on the image encoding formats. For example, color JPEG images can be encoded as a 24 bit TrueColor image or a 8 bit pseudo color format. Human visual acuity models can help direct the particular format for transcoding that shows the least visual quality loss. For example, human eyes are less sensitive to chrominance (color) values than luminance (brightness) values. Hence, it is possible to drop more chrominance components without noticeable loss in image quality. Also, image compression algorithms exploit color frequency distributions inherent in the original image (e.g. Huffman encoding). Without proper precautions, transcoding operations can perturb such distributions, leading to reduced compression ratios (i.e. increased image size) and lower quality images. Even if such transcoded output images are eventually discarded, valuable compute resources may still have been wasted. Note that it is not possible to always predict if a given transcoding can offer any space savings.

(a) Original JPEG image (56 KB) (b) Lower Quality Factor JPEG image (16 KB) (c) Cropped JPEG image (8 KB) (d) Thumbnail JPEG image (8 KB)

FIGURE 1.2
Transcoding an image to other variants (pictures courtesy: NASA)

1.2.3 Streaming media

As the number of clients with high bandwidth links to the last mile increase, streaming video and audio formats are increasingly becoming popular. By their very nature, streaming media tend to consume large amounts of data and also offer tremendous options for transcoding operations. Popular transcoding for streaming formats include clipping and highlighting, quality reduction (e.g. color depth, audio fidelity reduction) as well as transcoding to alternative formats (e.g. removing the video track completely). Streaming media allows such transcoding decisions to be dynamic, the stream quality can be dynamically improved or reduced based on the current network conditions. By contrast, transcoding decisions for images cannot usually be undone half way through transmission.

1.2.4 Content Adaptation of Composite Web Objects

Typical web documents can consist of a number of inlined images and other multimedia objects. It may be more appropriate to base transcoding decisions on the entire document group rather than on each individual components. Such analysis might expose the component intent; allowing more appropriate decisions. For example, document bullet icons can be easily identified by their corresponding HTML tags and appropriately replaced with a textual icon rather than transcoding each image to a lower fidelity version.

1.2.5 Quality Aware Transcoding

In general, for transcoding to provide the degree of control needed to deliver differentiated service, we need to understand its inherent tradeoff characteristics: the information quality loss, the computational overhead required in computing the transcoding and the potential benefits of reduced bandwidth requirements. Formal measurement of object quality loss can help in choosing the appropriate transformation from various transcodings illustrated in Figure 1.2. Without such characterization, transcoding policies do not have the ability to measure information quality loss of

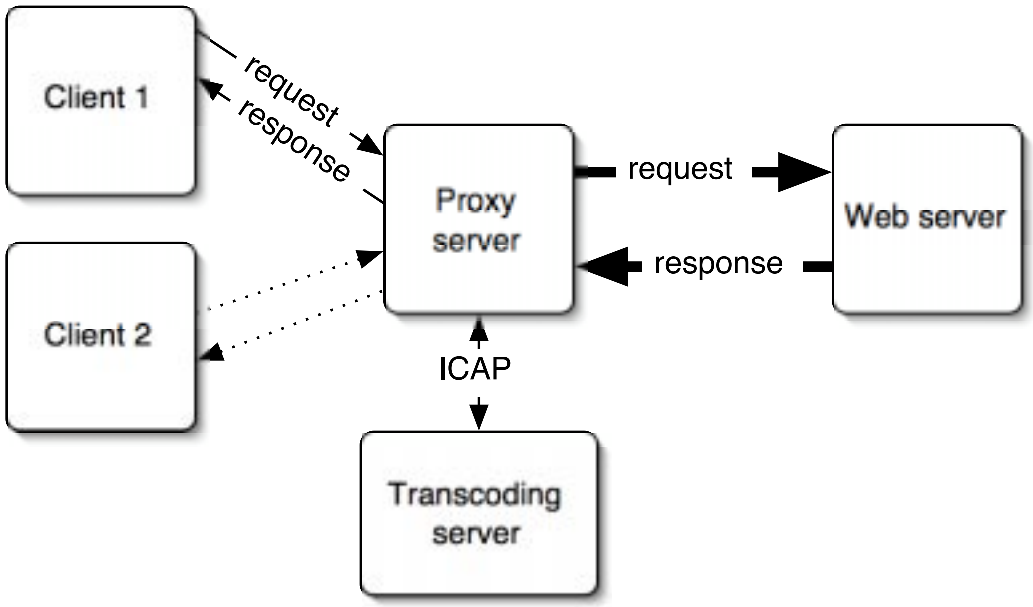


FIGURE 1.3

Web Content Adaptation Architecture

a transcoded image so it is impossible to ensure that preferred clients experience low degradation of quality. Without this characterization, systems that attempt to use transcoding to reduce bandwidth requirements can only transcode images to ad hoc Quality factor values, potentially leading to an increase in size for certain images. Systems have traditionally countered this by (unnecessarily) transcoding all images to a conservatively low Quality factor value.

To illustrate the benefits of such quantification for one specific case, Chandra and Ellis [1999] characterized the information quality tradeoffs, the computational requirements and the potential size reductions for transcodings that change the JPEG (Pennebaker and Mitchell [1993]) compression metric. It had been shown that the JPEG Quality factor parameter reflects a user's perception of image quality (Ford [1997]; Jacobson et al. [1997]). Chandra et al. analyzed typical web images (Chandra et al. [2001]) and developed techniques for measuring the initial Quality factor of an JPEG image as well as predict the computational cost as well as potential space benefits achieved by the transcoding. Such results are useful in any system that uses transcoding to reduce access latencies, increase effective storage space as well as reduce access costs.

1.3 Technologies that utilize transcoding operation

In the last section, we described some of the transcoding operations for various web object types. In this section, we describe various mechanisms on the web server that utilize transcoding to customize the object.

1.3.1 Web Content Adaptation Service Architecture

A typical web service architecture is illustrated in Figure 1.3. The system consists of a number of clients requesting web objects using varying network conditions. Web proxies are strategically placed in the network infrastructure to provide caching, traffic control and aggregation. Such proxies can be placed closer to the client, in the Internet back bone and at the servers themselves as reverse proxies. For simplicity, we illustrated these proxies as a single entity. These proxy servers can be expected to perform the content adaptation without modifying the origin servers themselves. These proxies can dynamically invoke transcoding operations using dedicated servers. The proxy servers can request transcoding services using the Internet Content Adaptation Protocol (ICAP) (Elson et al. [2000] - www.i-cap.org).

1.3.1.1 Internet Content Adaptation Protocol (ICAP)

ICAP is a protocol designed to off-load transformation and processing for web content to dedicated servers. ICAP allows ICAP clients to pass HTTP messages to ICAP servers for "adaptation". ICAP provides a light-weight RPC functionality for web services. Typical operations envisioned include language translation, virus checking, filtering and transcoding. The server executes its transformation service on messages and sends back responses to the client, usually with modified messages. The adapted messages may be either HTTP requests or HTTP responses. All ICAP transactions are based on standard HTTP.

1.3.2 Automatic transcoding by proxies and web servers

The simplest form of using transcoding is to automatically transcode objects to another form that is deemed useful to the end user. Typically, such transcodings are implemented in the Web proxy server which are closer to the end user. For example, America Online transcodes all JPEG images to a proprietary Johnson Grace ART format. Such transcodings are typically transparent to the end user. However, such transcoding operations run the risk of making inappropriate decisions. For example:

Lower quality object for end user: Automatic transcodings can make quality decisions that the user is not aware of; the user may actually want to pay higher cost to view higher quality objects because they are interested in them. Automatic transcoding is transparent and hides the fact that the images were transcoded.

Loss of content provider control: Even though transcodings can make web objects useful and applicable in many scenarios, content providers may want to exercise greater control, potentially denying service to certain clients. For example, a classical musician may not want users to listen to their master piece at a quality factor lower than some threshold (even if that meant denying service to a potential listener).

Copyright issues: Automatic transcoders may also not be aware of local copyright restrictions. Translating and derived work on copy righted work can have legal implications in certain jurisdictions. Depending on the location of the transcoding proxies, their operations may violate local tenets.

Some form of content provider and user interaction to influence the transcoding process is preferable.

1.3.3 Content producer and consumer involvement

Involving the content provider and the consumer is attractive for a number of reasons. The content provider can provide a-priori transcodings that are specifically customized to provide the objects in

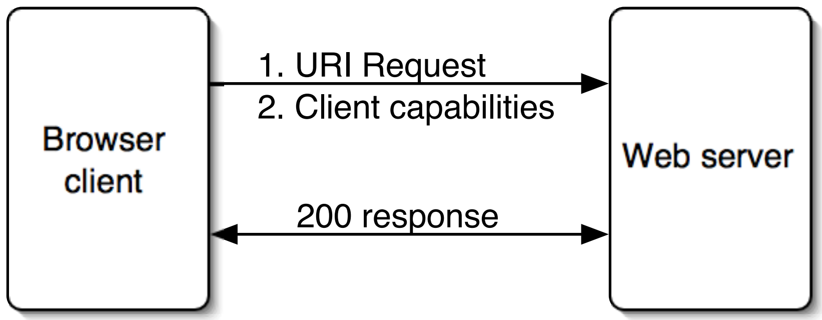


FIGURE 1.4

Server Driven Negotiation

a format that is acceptable. Also, the clients can better specify their tolerances and capabilities of their device in order to influence the object matching process. User would want to provide bounds on access latency and access cost. In general, we need a mechanism that addresses the following:

end users: allow end users to specify their tolerances (i.e. formats, size expressed as cost or latency tolerance)

proxy servers: negotiate variations of objects on behalf of a specific end user, while still serving other clients experiencing different requirements

content providers: maximum flexibility in controlling the content variations deemed acceptable for a particular situation

HTTP 1.1 (Fielding et al. [1999]; Holtman and Mutz [1998a,b]) provides such mechanisms to allow the server and client to negotiate contents. There are two kinds of content negotiation which are possible in HTTP 1.1: server-driven and agent-driven negotiation. These two kinds of negotiation are orthogonal and thus may be used separately or in combination. For example, proxy caches can use one such hybrid mechanism, referred to as transparent negotiation, utilizing agent-driven negotiation for information provided by the origin server in order to provide server-driven negotiation for subsequent client requests.

1.3.3.1 Server-driven Negotiation

The server-driven negotiation is illustrated in Figure 1.4. The server selects the appropriate variation based on client specified preferences. The client can specify their preferences using additional HTTP headers with the original request. Clients can use headers such as Accept, Accept-Language, Accept-Encoding etc. (Fielding et al. [1999]) to specify their preferred encoding, language etc. The efficacy of this scheme depends on the richness with which the client can specify its preference. However, adding such rich specification to every HTTP request can lead to inefficient network usage for all requests. Even if the browser specified all its capabilities, it still may not capture the users intent for a specific object. Users may want to change their preferred capabilities depending on the specific object. For example, users may want to view a lower quality version while generating a higher quality hard copy print out. Also, there are privacy concerns in transmitting the exact capabilities of the users browser to every web server (servers can keep track of the applications installed in the client). Such policies can also burden the server with the decision process for each request.

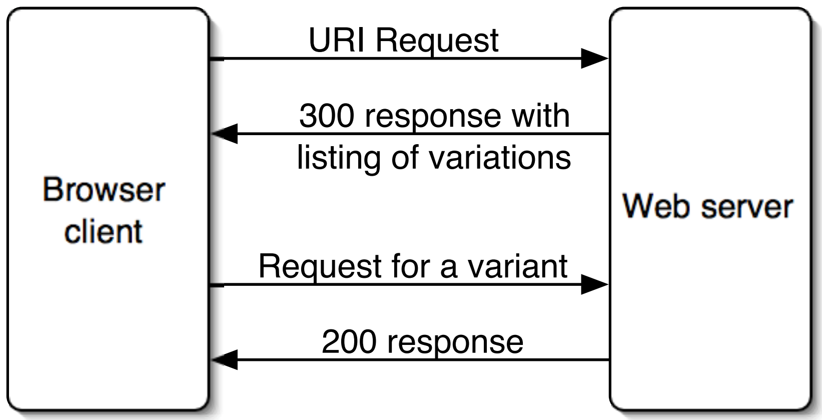


FIGURE 1.5

Agent Driven Negotiation

If the server was not able to negotiate a preferred object (either because it doesn't have the requests object or is unwilling to serve the requested object), it may use a Vary header to provide the alternatives that it is willing to provide to this user.

In summary, server-driven negotiation requires the clients to expose their capabilities and leverage processing power available on content servers to receive the appropriate content variation.

1.3.3.2 Agent-driven Negotiation

With agent-driven negotiation (Figure 1.5), the selection of the best representation is performed by the client. The server responds with all the available variations to the initial request. The server can generate this list using static, user provided variations or generate these variations dynamically. For example, Apache web servers utilize '.Var' files to allow users to specify these variations. The client browser chooses the acceptable format and specifically requests a (non-negotiated) URL. This selection can either happen automatically by the browser itself or via an explicit query to the end user.

In summary, agent-driven negotiation trades-off the extra request to the server for the necessity to expose its entire capabilities to the server. The decision overhead is paid by each client, rather than the origin server.

1.3.3.3 Transparent Negotiation

It is also possible to utilize a combination of server and agent driven negotiation in the form of transparent negotiation. Proxy caches can utilize information about the object variations available in the origin server to initiate server-driven negotiation with subsequent client requests. Transparent negotiation distributes the negotiation overhead with the proxy servers. However, transparent negotiation could potentially miss out on dynamic variations that become available on the server since the original agent-driven negotiation.

1.3.4 Systems that have utilized transcoding technologies

A number of systems and research prototypes have used transcoding to fit images to the current operating environment.

Han et al. (Han et al. [1998]) presented an analytical framework for determining whether to transcode and how much to transcode an image. Odyssey (Noble et al. [1997]) manipulated the JPEG Compression metric as a distillation technique for a web browser that adapts to changing network environments. The GloMop (Fox and Brewer [1996]; Fox et al. [1996]) project used transcoding to generate thumbnails on the fly to speed up image access from slow modems. The Soft Caching project (Ortega et al. [1997]; Kangasharju et al. [1998]) used progressive JPEG to produce lower quality images at the same spatial resolution. Caubweb (Mazer et al. [1998]) described a generic proxy service with content type specific application transducers that perform filtering operations such as transcoding. The Mowgli project (Liljeberg et al. [1995, 1996]) described content type specific lossless and lossy compression techniques to improve the web experience over wireless mobile links. Some of those systems do not use an informed transcoding technique that can quantify the information loss from the chosen transcoding operation. Chandra et al. (Chandra et al. [2001]) analyzed typical web images to develop a quality aware informed transcoding mechanism for these JPEG images (Chandra and Ellis [1999]). They utilized this technology in a proxy server to manage client network restrictions (Chandra et al. [1999]), in a web server to provide differentiated quality of service (Chandra et al. [2000a]) and in a mobile content creation device (digital camera) to manage the available storage and battery capacity (Chandra et al. [2000b]).

Commercial products such as WebExpress (Floyd et al. [1998]) from IBM, QuickWeb technology (intel) from Intel and Fastlane (Spectrum Information Technologies Inc.) from Spectrum Information technology have used various forms of compression and transcoding operations to improve web access from slow networks. The WebQoS (webqos) system from HP provides quality of service on the Web by using priority levels to determine admission priority and performance-level. WebQoS uses parameters such as source IP address, destination IP address, URL, port number, hostname and IP type-of-service to classify requests. The system uses these priorities in controlling the allocation of CPU and disk resources. Higher priority requests are sent to servers running in separate ports that operate under different system priorities. The system uses priorities to delay or deny service to lower priority clients. Though this policy leads to predictable service for the preferred clients, the lower priority clients can be turned away.

1.4 Challenges in the effective use of transcoding technologies

In this chapter, we described transcoding as a technique to adapt content to constrained environments. We summarize the important features discussed:

Transcoding is a lossy operation to generate variations of objects that are better suited to the current operating environment. Such variation allows web access from constrained client as well as at busy servers.

If care is not taken to understand the transcoding characteristics, such operations can yield objects that may not convey the intended information. With the emergence of new representation formats (e.g. JPEG 2000), it is imperative that informed transcoding decisions are made for these newer formats.

Transcoding operations that are appropriate for the clients usage patterns and that can also preserve the content providers intent without violating the providers rights and content control is a major challenge in enabling transcoding technologies.

Index

References

Tarek F. Abdelzaher and Nina Bhatti. Adaptive content delivery for web server qos. In *International Workshop on Quality of Service*, London, UK, June 1999a.

Tarek F. Abdelzaher and Nina Bhatti. Web content adaptation to improve server overload behavior. In *Eigth International World Wide Web Conf.*, Toronto, Canada, May 1999b.

All Things Web. Third State Of the Web Survey (SOWS III). http://www.pantos.org/atw/35654.html, May 1999.

Surendar Chandra and Carla Schlatter Ellis. JPEG Compression Metric as a Quality Aware Image Transcoding. In *Proceedings of the 2nd USENIX Symposium on Internet Technologies and Systems USITS-99*, pages 81–92, Boulder, CO, October 1999. USENIX Association.

Surendar Chandra, Carla Schlatter Ellis, and Amin Vahdat. Multimedia Web Services for Mobile Clients Using Quality Aware Transcoding. In *Proceedings of the Second ACM International Workshop on Wireless and Mobile Multimedia (WoWMoM'99)*, pages 99–108, Seattle, August 1999. ACM SIGMOBILE.

Surendar Chandra, Carla Schlatter Ellis, and Amin Vahdat. Differentiated multimedia web services using quality aware transcoding. In *INFOCOM - Nineteenth Annual Joint Conf. Of The IEEE Computer And Communications Societies*, pages 961–969, Tel Aviv, Israel, March 2000a. IEEE.

Surendar Chandra, Carla Schlatter Ellis, and Amin Vahdat. Managing the storage and battery resources in an image capture device (digital camera) using dynamic transcoding. In *Proceedings of the Third ACM International Workshop on Wireless and Mobile Multimedia (WoWMoM'00)*, pages 73–82, Boston, August 2000b. ACM SIGMOBILE.

Surendar Chandra, Ashish Gehani, Carla Schlatter Ellis, and Amin Vahdat. Transcoding characteristics of web images. In Martin Kienzle and Wu chi Feng, editors, *Multimedia Computing and Networking (MMCN'01)*, volume 4312, pages 135–149, San Jose, CA, January 2001. SPIE - The International Society of Optical Engineering.

Sukjae Cho. Power Management of iPAQ, February 2001.

Jeremy Elson, John Martin, Edward Sharp, John Schuster, Alberto Cerpa, Peter Danzig, Chuck Neerdaels, and Gary Tomlinson. Icap - the Internet content adaptation protocol. Technical report, ICAP forum, http://www.i-cap.org, March 2000.

Laura Marie Feeney and Martin Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proceedings IEEE INFOCOM 2001*, volume 3, pages 1548–1557, Anchorage, Alaska, April 2001.

R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. Hypertext transfer protocol – HTTP/1.1. Technical Report RFC 2616, Network Working Group, June 1999.

Rick Floyd, Barron Housel, and Carl Tait. Mobile Web Access using eNetwork Web Express. *IEEE Personal Communications*, 5(5), October 1998.

Adrian M. Ford. *Relations between Image Quality and Still Image Compression*. PhD thesis,

0-8493-0052-5/00/$0.00+$.50

University of Westminster, May 1997.

Armando Fox and Eric A. Brewer. Reducing www latency and bandwidth requirements via real-time distillation. In *Proceedings of the Fifth International World Wide Web Conf.*, pages 1445–1456, Paris, France, May 1996.

Armando Fox, Steven D. Gribble, Eric A. Brewer, and Elan Amir. Adapting to network and client variability via on-demand dynamic distillation. *ACM SIGPLAN Notices*, 31(9):160–170, September 1996. ISSN 0362-1340. URL "url–http://www.acm.org:80/pubs/citations/proceedings/asplos/237090/p16%0-fox/˝. Co-published as SIGOPS Operating Systems Review **30**(5), December 1996, and as SIGARCH Computer Architecture News, **24**(special issue), October 1996.

Richard Han, Pravin Bhagwat, Richard LaMaire, Todd Mummert, Veronique Perret, and Jim Rubas. Dynamic adaptation in an image transcoding proxy for mobile web browsing. *IEEE Personal Communications Magazine*, 5(6):8–17, December 1998.

Paul J. M. Havinga. *Mobile Multimedia Systems*. PhD thesis, Univ. of Twente, February 2000.

Koen Holtman and Andrew H. Mutz. Http remote variant selection algorithm – rvsa/1.0. RFC 2296, March 1998a.

Koen Holtman and Andrew H. Mutz. Transparent content negotiation in http. RFC 2295, March 1998b.

intel. Intel QuickWeb. http://www-us-east.intel.com/quickweb/.

R. E. Jacobson, A. M. Ford, and G. G. Attridge. Evaluation of the effects of compression on the quality of images on a soft display. In *Proceedings of the SPIE: Human Vision and Electronic Imaging II*, San Jose, CA, Feb 1997.

Jussi Kangasharju, Younggap Kwon, and Antonio Ortega. Design and implementation of a soft caching proxy. In *3rd Intl. WWW Caching Workshop*, Manchester, England, June 1998.

Tim Kindberg, John Barton, Jeff Morgan, Gene Becker, Ilja Bedner, Debbie Caswell, Philippe Debaty, Gita Gopal, Marcos Frid, Venky Krishnan, Howard Morris, Celine Pering, John Schettino, Bill Serra, and Mirjana Spasojevic. People, places, things: Web presence for the real world. In *Third IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, Monterey, CA, dec 2000. IEEE.

LAN/MAN Standards Committee of the IEEE Computer Society. *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE, 3 Park Avenue, New York, NY 10016, 1999.

M. Liljeberg, T. Alanko, M. Kojo, H. Laamanen, and K Raatikainen. Optimizing world-wide web for weakly connected mobile workstations: An indirect approach. In *Proceedings of the 2nd International Workshop on Services in Distributed and Networked Environments (SDNE'95)*, Whistler, Canada, June 1995.

Mika Liljeberg, Heikki Helin, Markku Kojo, and Kimmo Raatikainen. Mowgli www software: Improved usability of www in mobile wan environments. In *IEEE Global Internet 1996*, London, England, November 1996. IEEE Communications Society.

Murray S. Mazer, Charlie Brooks, John LoVerso, Louis Theran, Fredrick Hirsch, Stavros Macrakis, Steve Shapiro, and Dennis Rockwell. Distributed clients for enhanced usability, reliability, and adaptability in accessing the national information environment. Technical report, The Open Group Research Institute, 11 Cambridge Center, Cambridge MA 02142, 1998.

Jakob Nielsen. *Usability Engineering*. Academic Press, Boston, MA, 1993. ISBN 0-12-518405-0.

URL “url–http://www.useit.com/jakob/useengbook.html˝. (hardcover), 0-12-518406-9 (paperback).

Brian D. Noble, M. Satyanarayanan, Dushyanth Narayanan, J. Eric Tilton, Jason Flinn, and Kevin R. Walker. Application-aware adaptation for mobility. In *Proceedings of the 16th ACM Symposium on Operating Systems and Principles*, Saint-Malo, France, October 1997.

Antonio Ortega, Fabio Carignano, Serge Ayer, and Martin Vetterli. Soft Caching: Web Cache Management Techniques for Images. In *IEEE Signal Processing Society 1997 Workshop on Multimedia Signal Processing*, Princeton NJ, Jun 1997.

William B. Pennebaker and Joan L. Mitchell. *JPEG - Still Image Data Compression Standard*. Van Nostrand ReinHold, New York, 1993.

Spectrum Information Technologies Inc. Fastlane. http://www.spectruminfo.com/.

Mark Stemm, Paul Gauthier, Daishi Harada, and Randy H. Katz. Reducing power consumption of network interfaces in hand-held devices. In *Proceedings of the 3rd International Workshop on Mobile Multimedia Communications (MoMuc-3)*, Princeton, NJ, September 1996.

webqos. WebQoS Version 2. http://www.hp.com/go/webqos.

Mark Weiser. Ubiquitous computing. *IEEE Computer Hot Topics*, October 1993.