Adapting Multimedia Internet Content for Universal Access

Rakesh Mohan, *Member, IEEE*, John R. Smith, *Member, IEEE*, and Chung-Sheng Li, *Senior Member, IEEE*

***Abstract*—Content delivery over the Internet needs to address both the multimedia nature of the content and the capabilities of the diverse client platforms the content is being delivered to. We present a system that adapts multimedia Web documents to optimally match the capabilities of the client device requesting it. This system has two key components. 1) A representation scheme called the *InfoPyramid* that provides a multimodal, multiresolution representation hierarchy for multimedia. 2) A *customizer* that selects the best content representation to meet the client capabilities while delivering the most value.**

We model the selection process as a resource allocation problem in a generalized rate-distortion framework. In this framework, we address the issue of both multiple media types in a Web document and multiple resource types at the client. We extend this framework to allow prioritization on the content items in a Web document. We illustrate our content adaptation technique with a web server that adapts multimedia news stories to clients as diverse as workstations, PDA's and cellular phones.

***Index Terms*—Compression, content adaptation, Internet, multimedia, information appliances, rate-distortion, transcoding, universal access.**

I. Introduction

NETWORK appliances, or information appliances, are computing devices that are network enabled. They typically have fewer resources than personal computers and are geared toward a limited number of applications. Some current examples of network appliances are hand-held computers (HPC's), personal digital assistants (PDA's), set-top boxes, screen telephones, smart cellular phones and network computers. In "ubiquitous" or "pervasive" computing, consumers will use different network appliances to connect to the Internet for different applications, from entertainment to banking, from different settings, from living rooms to cars. Sources, such as *The Economist* [1] and International Data Corporation (IDC) [2], predict that the sales of network appliances will significantly outstrip that of personal computers after the year 2002. Therefore, within a decade, network appliances will replace personal computers as the client device of choice for viewing Web content.

Currently multimedia content is authored with the personal computer as the target client device. Web documents, which have rapidly become the largest deployed form of multimedia, are also authored specifically for personal computers with reasonable wired network connections. However, network appliances are very different from the typical PC on a modem or LAN. The network appliances vary widely in their features such as screen size, resolution, color depth, computing power, storage and software. They also use a variety of network connections ranging from cable to mobile, with different bandwidth, connection characteristics and costs [7]. The diversity of these devices will make it difficult and expensive to author multimedia content separately for each individual type of device. Therefore, technologies that can adapt multimedia content to diverse client devices will become critical in the coming pervasive computing era.

In this paper we present a system that **adapts** multimedia Web content to optimally match the resources and capabilities of diverse client devices. This system employs two key technologies.

1) A progressive data representation scheme called the **InfoPyramid** [25]. Content items on a Web page are transcoded into multiple resolution and modality versions so that they can be rendered on different devices. For example, a video item is transcoded in to a set of images so that it can be rendered on a device not capable of displaying video. The InfoPyramid provides a multimodal, multiresolution representation for the content items and their transcoded versions.
2) A **customizer** that selects the best versions of content items from the InfoPyramids to meet the client resources while delivering the most "value." The customizer allocates resources on the client among the items in the document. This resource allocation results in the selection of the appropriate resolution or modality of the content items. If the client has limited resources (such as a PDA or pager), some of the content items may not get any resources assigned and thus will not be delivered to the client. We propose a novel *value-resource* framework for the customizer. This value-resource framework allows us to design and analyze a number of content adaptation strategies.

We illustrate this content adaptation with a multimedia news delivery system that adapts to clients ranging from workstations to cellular phones.

A. Related Work

Much work (for a small sampling, see [3]–[6]) has been done on adapting video to bandwidth variations by selecting a suitable compression scheme. These systems consider only a single type of media, not composite multimedia documents.

Manuscript received September 9, 1998; revised December 9, 1998. The associate editor coordinating the review of this paper and approving it for publication was Dr. Thomas R. Gardos.

The authors are with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA.

Publisher Item Identifier S 1520-9210(99)01784-8.

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

Drastically different clients, such as those that cannot handle video, are not addressed.

Web content adaptation can be performed either at the server, at the client, at an intermediate proxy, or some combination of the three.

Some client devices adapt content at the device. For example, Windows-CE™ devices change color-depth (for example, from 24-bit color to 4-bit gray-level) of images. The drawbacks are that network appliances have low network bandwidth, which results in slow access to pages with rich multimedia, and they are restricted in their computational power, which makes content adaptation at the device slow, or even impossible.

Most content adaptation systems [7]–[16], [18] are http proxy-based. The proxy intercepts client device's requests for Web pages, fetches the requested content, adapts it, and sends the adapted version to the client. This content adaptation is often termed "*transcoding*."

In the TranSend project [7]–[10] a proxy transcodes Web content on the fly. The adaptation, which they term "distillation," is primarily limited to image compression and reduction of image size and color space. Video is also converted into different frame-rates and encodings using a video gateway [6]. Based on this work, a company, Proxinet [16], has been started that provides a proxy which customizes content for a special browser on the 3Com PalmPilot™ [17].

Bickmore and Schilit [11] also propose a proxy based mechanism. They use a number of heuristics and a planner to perform outlining and elision of the content to fit the Web page on the client's screen.

The Spyglass Prism™ [13], a commercial product, is another transcoding proxy. AvantGo [18] offers a solution similar to Proxinet.

Content adaptation upstream of the client results in a faster response time [7], [8]. Based on this observation, Intel launched the QuickWeb™ [12] service that compresses images at a proxy.

These transcoding proxies typically consider a few client devices and employ static, *ad-hoc*, content adaptation strategies. A common policy [7]–[13] is to scale all images by a fixed factor. Thus, these transcoding proxies fail to dynamically address the variation in the resource requirements of different Web documents. The set of client devices will also grow more diverse. Certain resources, such as effective network bandwidth, costs and patience of the users can be different for similar client devices. The static adaptation policies used by these systems do not handle well this variability in Web content and client resources.

None of the existing transcoding systems (with the possible exception of [11] and [14]) consider the requirements of the entire Web page or relationships between its various components in different media. Also, these systems only consider transcoding within the same modality.

In this paper, we propose a content adaptation framework that ***dynamically*** accounts for resource requirements of the complete Web page and its individual components. It selects from a number of different possible transcoded versions of the content, ones that provide the "best value" within the transcoding between modalities. We provide a theoretical framework in which various content adaptation policies can be formulated and analyzed.

One big benefit of the proxy approach is that it is totally transparent to the content providers; they do not have to change the way they author or serve content. However, there are a number of drawbacks to this approach:

1) content providers have no control over how their content will appear to different clients;
2) there may be legal issues arising from copyright that may preclude or severely limit the transcoding by proxies;
3) HTML tags mainly provide formatting information rather than semantic information;
4) on the fly transcoding is difficult to apply to many media types such as video and audio.

These factors limit both the quality and the amount of customization that proxies can provide.

In this paper we present an alternate solution that extends the Web server deployed by a content provider. In this system, the content author can lay the transcoding policies and control the adaptation process. Also, the content author can edit and replace the transcoded versions of content items generated by the system. This control of the customization overcomes problems of publisher control and copyright issues faced by transcoding proxies [7]–[18]. The content is authored in XML [23], allowing the author to provide more information to the transcoding and customization system than can be deduced from an HTML page. The key benefit of this server-based system is that due to the guidance provided by the author, significantly greater level of customization can be performed than is possible in transcoding proxies. The systems generates transcoded versions of the content items prior to any requests; thus, it can handle media items such as video and audio which are difficult to handle in proxies. This off-line transcoding also leads to lower response latencies than proxies. The server shares the benefit of transcoding proxies in speeding content delivery as the customized content is often much smaller than the original content.

B. Outline

We first present the overall architecture of the system. The InfoPyramid, a multimodal, multiresolution representation hierarchy for multimedia, content analysis, transcoding modules, content customization, and cache, is described in Section II.

In Sections III–V, we describe the customization process in detail. In Section VI, we present an implementation of the content adaptation system. We present a summary in Section VII.

II. System Architecture

The content adaptation system is an extension to a Web (http) server. An overview of the system architecture is shown in Fig. 1. The content source contains the multimedia content to be delivered by the Web server. First, content is analyzed to extract meta-data used in guiding subsequent transcoding and selection processes. Based on the capabilities

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

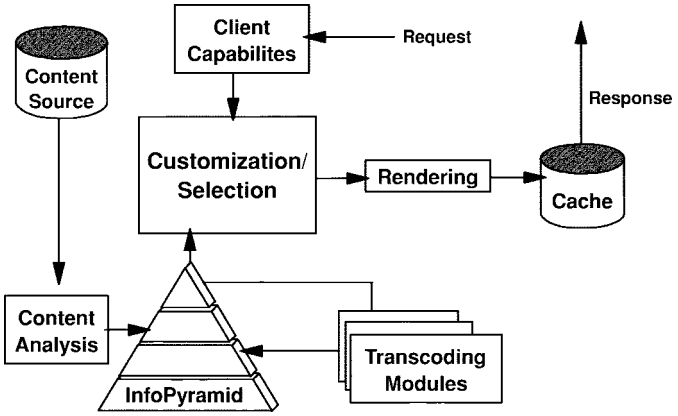

Fig. 1. Internet content adaptation system architecture.

are employed to generate versions of the content in different resolutions and modalities. A novel data representation, the InfoPyramid, is used to store the multiple resolutions and modalities of the transcoded content, along with any associated meta-data. This transcoding is done off-line, during content creation time. When the Web server receives a request, it first determines the capabilities of the requesting client device. A customization module then dynamically selects from the InfoPyramids, the resolutions or modalities that best meet the client capabilities. This selected content is then rendered in a suitable delivery format (for example, HTML) for delivery to the client. A cache that stores these client specific versions of content is used to improve response times. In the following sections, we describe these processes in detail.

A. Client Devices

The types of devices that can access the Internet are rapidly expanding beyond the workstation on LAN that most multimedia Internet content is authored for [1], [2], and [7]. One can now use personal digital assistants (PDA) such as the PalmPilot™ and Sharp Zaurus™, handheld personal computers (HPC) such as the Psion and numerous Windows-CE™ machines, various Internet capable phones such as the AT&T Smartphone™ (cellular) and Screenphone (wired), set-top boxes such as WebTV™ etc. to browse the Web. Even traditional computers such as workstations, laptops and PC may vary widely in their display and specially in their network bandwidth. The browsers designed to meet the special needs of handicapped people can be modeled as client devices with specific capabilities [19]. For example, a speech browser for the blind may be modeled as a device that only supports audio. Thus, we see that to fulfill the promise of universal access to the Internet, devices with very diverse capabilities need to be catered to.

Currently, the system considers the following client device characteristics.

1) *Screen* size i.e., width and height in pixels, color and bits/pixel.
2) Effective *Network bandwidth*.
3) *Payload* defined as the total amounts of bits that can document. For streaming media this includes only the initial buffer space required before the media starts playing, not the size of the media itself. The payload is defined as the product of the network bandwidth and the time the client is ready to wait (bandwidth* wait-time) before the complete Web page downloads. For storage constrained devices, the payload will be defined as the storage space.
4) *Capabilities* for displaying video/audio/image.

B. Content

We will restrict our discussion to Web pages. The content is authored in XML [21], which is converted to HTML prior to delivery. We are also working on an extension to HTML that allows authors to introduce more information for content customization using XML and also enables our content adaptation system to be deployed at proxies.

A multimedia Web ***document*** W is composed of a number of component ***items*** $A, W = \{A_i\}, i = 1, n$. Each item can be an atomic unit of media, such as an image or a video clip. An item can also be composed of other items, for example a document can have a number of stories as content items, and each story item may be composed of image items, text items, etc. For simplicity, we will first consider only atomic content items, and then, in Section V-B, deal with composite items.

C. Content Analysis

The authored content is analyzed to extract information that will be useful in transcoding and customization. Two types of content analysis are performed.

Each atomic item A of the document is analyzed to determine its resource requirements. The types of resources considered are those that may differentiate different client devices. We determine the following resource requirements.

1) Static content size in bits.
2) Display size such as height, width and area.
3) Streaming bit-rate.
4) Color requirements.
5) Compression formats.
6) Hardware requirements, such as display for images, support for audio and video.

The semantics of the content items are determined in the context of the entire document. We currently analyze images to determine their type and purpose [22], [23]. This analysis allows us to improve image transcoding by selecting policies according to image type and purpose [22].

D. InfoPyramid

The InfoPyramid [25] is a framework for aggregating the individual components of multimedia content with content-descriptions, and methods and rules for handling the content and content descriptions [24]. The InfoPyramid describes content in different modalities, at different resolutions and at multiple abstractions. In addition, it defines methods for manipulating, translating, transcoding, and generating the con-

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

modalities and resolutions so that it can be rendered on a variety of devices. Fig. 1 shows a simplified InfoPyramid for a video.

Multimodal: Multimedia content is usually not in a single media format, or modality. A video clip can contain raw data from video, audio in two or more languages, and closed captions. In the medical arena, MRI, CT, PET, and ultrasound can be captured for the same patient, resulting in multiple three-dimensional (3-D) scans of the same content.

For certain devices, the appropriate content modality may not be available. The required modality may be generated by transforming other modalities. For example, a video clip can be transformed into images showing keyframes [36], while text can be synthesized into speech.

Multiresolution: Each content component can also be described at multiple resolutions. Numerous resolution reduction techniques exist for image and video. Features and semantics at different resolutions can be obtained from raw data or transformed data at different resolutions, thus resulting in a feature or semantic pyramid.

Multiple-Abstraction Levels: The abstraction levels describe features and data in a hierarchical fashion. For example, one hierarchy could be features, semantics and object descriptions, and annotations and meta-data. For content adaption, we store meta-data such as size, color, bandwidth requirements, publisher preferences, etc., for each constituent element. This meta-data may be supplied by content analysis (Section II-C) and/or by the content author.

Methods and Rules: Methods generate content descriptors from the features of the data, or analyze, manipulate, provide modality translation, or process the data in various ways. In addition, the InfoPyramid may have rules to provide flexible application of the methods. Methods and rules provide linkage between different modalities, resolutions and abstractions. For content adaptation, we consider procedures and rules for translating and summarizing (transcoding) between modalities and resolutions.

The InfoPyramid concept can be further generalized by using other axes such as fault/loss tolerance, numerical complexity, interaction modality, etc. Rather than forcing a strong separation between the data and the content description meta-data, the InfoPyramid offers a continuum between the data, various abstractions of the data, and content description data.

Definitions: From each *original* item A_i, in the Web document $W = \{A_i\}$, an ***InfoPyramid*** $M_i = \{M_{ij}\}, j = 0, m_i$, is computed by transcoding A_i into j ***versions*** with different resolutions and modalities.[1] We will denote the original version by $M_{i0} = A_i$. We also introduce a null version, which corresponds to the item being deleted from the delivered content, by $M_{im_i} = \phi$.

E. Transcoding

Content transcoders populate the InfoPyramid structure with multiresolution, multimodal versions of the content. For example, in Fig. 2, the video is transformed to images by extracting a set of key frames [36]. Audio is also extracted from the video. Each of the modalities is then represented at different resolutions, bit-rates, color depth, etc. We have implemented a number of transcoding modules for handling video and images and imported others for text, images, video and audio. The system is designed to allow third-party content transcoders to be plugged in. The capabilities of the typical client devices and content analysis are used to guide the transcoding process. The transcoding is done off-line, unlike in previous proxy-based systems [7]–[18].

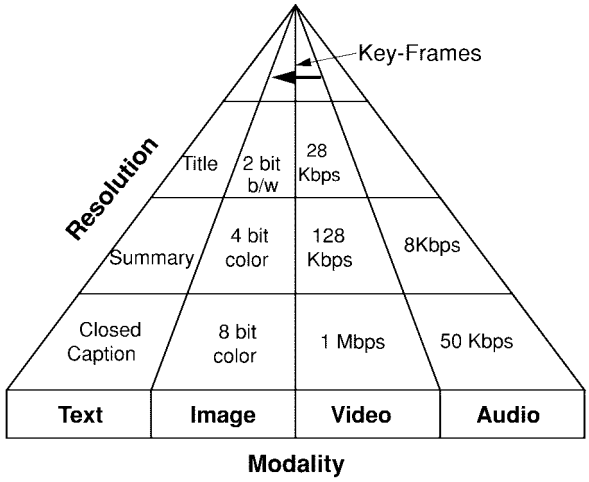

Fig. 2. An InfoPyramid for a video item.

F. Customization

The customization module uses the client device characteristics as constraints to pick the best content representation. The best representation is the one that maximizes content value for that client device. This customization process is detailed in Sections III–V.

The InfoPyramids represent the transcoded resolutions and modalities of the component multimedia items. From the InfoPyramids, the customization module selects the final ensemble such that it optimally satisfies all the client's resource constraints. This content selection is performed dynamically in response to a request. Thus, the customization is able to account for any time varying client resources such as effective bandwidth and storage.

The customization utilizes a value-resource framework, which is generalization of rate-distortion (Section III). We then solve the problem of generating a version of a Web document that provides the most "value" to a client within the client's resource constraints. In Section IV, we model the selection problem as one of optimal allocation of the resources on the client among the different versions of the multimedia items of the Web document. We show that different models for the relationship between the value and the resource requirements lead to different optimal resource allocation strategies. In Section V, we present extensions to the optimization process to 1) account for the importance of each item and 2) to jointly satisfy different class of resources, such as display area and

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

G. Cache

When a customized Web page is delivered to the client, it is also stored in a cache. When the system receives a request for a document, it first checks if a client with the same capabilities made the request previously, and if so, retrieves the corresponding customized. Temporal variations in resources on the client, such as bandwidth, CPU resources, storage, etc., will reduce the cache hit ratio. To effectively handle this, the cost of performing customization versus the variation in the resources will need to be considered. Our system currently performs customization again if the resources for the requesting client differ from the cached versions. Alternatively, one can group clients with very similar capabilities under the same client id. We will also explore the possibility of storing partial InfoPyramids based on customizations performed for clients, and using these to for subsequent customizations, thus reducing the search space for the customization.

III. Content Value

Image or video compression can be viewed as adapting the content to meet bit resource constraints. One framework for compressing to meet bit resource constraints [26], [28] has built on the rate-distortion $(R\text{-}D)$ theory due to Shannon [27]. Rate-distortion theory deals with the minimum bit-rate R needed to represent a source with desired distortion D, or alternately, given a bit-rate R determining the distortion D in the compressed version of the source. The rate-distortion framework is employed in many image and video compression systems, for example [26], [28]–[30], [33]. We generalize rate-distortion theory to a ***value-resource*** framework by considering different versions of a content item in an InfoPyramid as analogous to different compressions, and different client resources as analogous to the bit-rate.

Distortion is typically measured as the mean squared error (MSE) between the source and its compressed version. One problem with the MSE based distortion measure is that it may not correspond to the perceived loss of fidelity [31]. However, a bigger drawback is the difficulty of formulating a meaningful distortion measure when the adaptation is drastic. For example, it is difficult to measure the loss of fidelity when a video is transcoded to a set of key frames or transcoded into its textual transcript.

To overcome this problem, we introduce a subjective measure of fidelity which we call ***value***.

Definition: Value

$$V(M_{ij}) = \frac{\text{perceived value of transcoded version } M_{ij}}{\text{perceived value of original } M_{i0}}.$$

$$V \in [0,1], \quad V = \begin{cases} 1 & \text{for original item } M_{i0} \\ 0 & \text{when the item is excluded } M_{im_i}. \end{cases}$$

The benefit of V is that we have a measure for fidelity that is applicable to transcodings of media at multiple resolutions and multiple modalities. This also allows us to compare document items that were in different media types. However, the drawback is that we still do not have a computational assigned by the author for each transcoding, or we can assume some arbitrary functional relation between V and R, the resource utilized. In the special case where we can measure the distortion D of all the versions, and the distortion for the null version is assumed to be infinite, we have $V = (1/1+D)$.

The value/distortion is neither an easily estimated metric, nor is it uniform across different people with diverse interests. In general, it will also be difficult to manually assign values to different transcodings. The content value is a useful construct that helps us analyze various dynamic content adaptation policies in a theoretical rate-distortion based framework and draw parallels with compression.

IV. Resource Allocation

We can then model the content adaptation as the following resource allocation problem:

$$\max \left\{ \sum_i V_i \right\} \text{ such that } \sum_i R_i \le R_{\text{client}} \qquad (1)$$

where $V_i \in \{V(M_{ij})\}$ and $R_i \in \{R(M_{ij})\}$ are the values and resources used by the ith item M_i of the multimedia document. While V_i and R_i are discrete, we will first consider them to be continuous, and then deal with the discrete case. R_{client} is the maximum resource available at the client.

Let the value V_i be some function of the resource, R_i, i.e. $V_i = f_i(R_i)$. We convert the above *constrained optimization* problem to an *unconstrained* optimization problem by considering the Lagrangian [32]:

$$L(R,\lambda) = \left\{ \sum_i V_i + \lambda \left(R_{\text{client}} - \sum_i R_i \right) \right\} \text{ with } \lambda \ge 0.$$

Then if R_0 is an optimal solution, there exists a λ_0 such that $\Delta L(R,\lambda) = 0$. Given that the items, and thus their values, are independent of each other, we get $(\partial L/\partial R_i) = (df_i(R_i)/dR_i) - \lambda$. Therefore, the candidate solutions to (1) are given by

$$\frac{df_1(R_1)}{dR_1} = \frac{df_2(R_2)}{dR_2} = \cdots = \frac{df_i(R_i)}{dR_i} = \cdots = \lambda. \qquad (2)$$

A. Analytic Functions

Content value, as an alternative to distortion, makes it possible for authors or users to specify value judgements about various transcoded versions of the content. However, manually assigning the values is not a practical proposition in most scenarios. To mitigate this problem, we introduced functional mappings between content value and resource utilization. This is not to suggest that there actually exist such a simple mechanism for assigning value (or distortion). Computing distortion, even in specific modalities such as images, that is meaningful perceptually over all images and people is not easy [31]. Our framework allows one to design fast adaptation policies for a combinatorial resource allocation problem, by assuming a particular functional mapping that captures the general trend of reduction in value with resource

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.