Adapting Content for Mobile Devices in Heterogeneous Collaboration Environments

Sangmi Lee[1, 2], Sunghoon Ko[1], Geoffrey Fox [1, 3]
Community Grid Computing Labs, Indiana University [1]
Department of Computer Science, Florida State University[2]
Department of Computer Science, Indiana University [3]
slee@csit.fsu.edu, {suko, gcf}@indiana.edu

Abstract

The portability of new miniaturized devices, together with their ability to connect conveniently to networks in different places, makes *mobile computing* possible. Recent advances that have led to increasing bandwidths, for wireless communications, enable mobile users to utilize a vast range of sophisticated services such as real-time collaborative services. *Garnet Collaborative system* is a universal accessible collaborative environment supporting synchronous communication and universal accessibility from a wide range of devices. *Garnet Message System Micro Edition* (GMSME) is developed to integrate heterogeneous mobile devices into the Garnet collaborative system. Garnet integrates desktop in traditional LAN environments, wireless LAN based mobile devices, and pervasive 3G communication technology seamlessly. In this paper, we describe the process of adapting content in GMSME. GMSME provides application-aware transcoding, to tune the shared resources, for different collaborative models. Also, it optimizes shared resources for each heterogeneous client environments.

Key words: Content Adaptation, Transcoding, Collaboration, Mobile Computing

1. Introduction

The availability of miniaturized devices and wireless communications has made mobile computing feasible. An ever more mobile workforce, and the computerization of inherently mobile activities is driving a need for applications to be integrated with traditional distributed systems. Mobile cellular telephones are widely available these days. A handheld computer is integrated with such a telephone is called as a smartphone. Currently wireless network service vendors have introduced a wide bandwidth telephone network, 3G communication [2], and it enhances the possibility of adapting a smartphone as a client in traditional distributed systems. Existing wireless LAN technology is already equipped to *Personal Digital Assistants* (PDAs) and laptops, and enables *pervasive computing* with its wide-bandwidth network communication [3]. Pervasive or ubiquitous computing is a new emerging computing style which adapts various computing devices throughout our living and working spaces. These devices include PDAs, smartphone, traditional desktop, wearable computers and so on. These devices coordinate with each other and network services seamlessly.

We address the issue of pervasive computing for our collaborative system, *Garnet* [4]. We have been investigating design of an adaptive collaborative system supporting multiple heterogeneous devices. Moreover, we generalize it to universal collaboration – the capability of multiple users to link together over the web with disparate access modes. Despite innovative advances in computing capability and wireless communication services, mobile systems are typically slow, unreliable, and have unpredictable temporal characteristics. Further, the user interface is clearly limited. The design of distributed mobile applications needs to identify the practicalities, reliability, and possibilities of continuous interaction and integrate synchronous and asynchronous collaboration.

While adapting new devices, one should consider various factors: network communication mechanisms, computing capabilities, display capability, etc. We developed the *Garnet Message System Micro Edition* (GMSME) [20] to provide adaptability to the Garnet system. The major functionalities of GMSME are – support for heterogeneous wireless network communications, the manipulation and optimization of content, support for collaborative environment, and finally the integration of the 3rd party applications for mobile devices. In this paper, we focus on how we tune the shared content for the limited capable mobile devices according to the characteristics of specific collaborative features.

Originally, transcoding technology is considered for converting multimedia from one format to another format preferred by specific devices [12]. We utilize various transcoding technologies to adapt shared content based on the type of collaboration. Here transcoding technology includes image resizing, converting image formats, compressing data, and transform technology such as the use of stylesheets. There have been a number of approaches to accommodate mobile devices by transcoding technologies [25-29]. However, different collaborative features may need different transcoding technologies according to their unique functionality. In this paper, we will explain our approach based on two different and popular resource sharing model, shared display and shared export.

In shared display, one shares the bitmap display and the state is maintained between the clients by transmitting the changes in the display. Meanwhile, the shared export model filters the output of each application to one of a set of common formats and builds a custom shared event viewer for these formats. This allows a single collaborative viewer to be leveraged among several different applications. Document formats such as W3C's Scalable Vector Graphics (SVG) [15] or Adobe, Inc.'s Portable Document Format (PDF) [10] are particularly interesting and support of collaborative viewers is a major advantage of Garnet System. With the scalability of vector graphics, and separation of user presentation from the master's content, shared export provides more flexibility for scientific visualization or geographical information systems. The constraints of "real-time synchronous collaboration" for both methods, implies a delay of 10-100 millisecond for each participant [1]. In shared display method, the time constraint is the most important factor to perform its illusion of collaboration. Basically shared export also requires this "real-time" constraint; however the graphical quality has higher priority within the range of time delay for synchronous collaboration. We investigate technology for adapting content that can maximize the collaborative features and optimize network bandwidth.

The rest of this paper is organized as follows: Section 2 reviews other related works in the area of pervasive computing and collaboration systems. In section 3, we describe the data flow in Garnet collaborative system. We also discuss utilizing transcoding technologies in GMSME to fit the shared content for limited display system in section 4. We also describe application-aware transcoding used in GMSME, with our resultant data. The conclusion and our future directions are discussed in section 5.

2. Related Work

Adapting new devices and emerging technologies innovatively to collaborative services are described in [5-7]. Rendezvous [8], GroupKit [9] and several groupware toolkits utilize the model-view so that services can generate different views for different users. However, since the situation is greatly simplified by using a centralized architecture, they were not able to support the diversity in devices accessing the service. One of the early efforts to adapt PDAs to work with conventional desktop computers is Carnegie Mellon University's Pittsburg Pebbles PDA Project [6]. Pebbles is designed to communicate with PDAs through the communication server, PebblesPC, and every message is conforms to the Pebbles protocol defined in their header files.

Transcoding is one of the most popular ways to tune content from a service provider. Transcoding is the transformation that converts the multimedia object from one form to another, frequently trading object fidelity for size [25], and is used to convert image or video formats (reduction resolution or compress data). A classical approach from ORL AT&T Lab's VNC [13] allows user to customize encoding methods. As extended versions of VNC for mobile devices, there are Harakan Software's PalmVNC [7], Nokia's Java VNC viewer [14]. These approaches implemented the client side of the VNC protocol for PDAs and wireless phone. This They enabled mobile device to be a thin client of a traditional desktop coupled with the sharing of master's view and the execution of user events.

Transcoding technology is also used to fit document and graphics files to the unique constraints of mobile devices and other Web-enabled products. A number of distributed services use transcoding technology to generate documents for their heterogeneous clients [25-29]. The Apache Cocoon [32] project allows automatic generation of HTML, PDF, and WML files through the processing of statically or dynamically generated XML files using XSL [33] and XSLT [11]. The idea of transcoding is also adapted to Web service architectures. Duke University's Quality Aware Transcoding project [30] investigates differentiated Web services, which enables the Web services and Web servers to manage their available bandwidth with its quality aware transcoding. IBM introduces WebSphere Everyplace Access [12], which supports developers and administrators to utilize the transcoding technology by accessing portlet which performs transcoding. Transcoding technology is developed as a component of WebSphere, which is IBM's Web service infrastructure [24]. WebSphere Everyplace Access is designed to perform its transcoding process in their individual portlet. This approach is different from existing distributed systems supporting the transcoding process for heterogeneous devices, because it enables the transcoding process to be separated from the proxy architecture. It also enables content providers to provide high-quality transcoding technologies on the server side.

3. Dataflow in the Garnet Collaborative System

Garnet system enables synchronous/asynchronous communications and provides a collaboration environment for heterogeneous clients. The Garnet system deploys a conventional shared event model for collaboration. Garnet ensures that objects are shared properly between collaborating clients with messaging to propagate changes to the object's state. Hand-held devices, mobile phones, as well as conventional desktops can join in one collaboration session of Garnet system. Every message delivered to mobile devices by GMSME. GMSME is a proxy-based service unit for mobile devices [20].

As depicted in Figure 1, shared graphical content from Garnet is processed to fit within the display space for each mobile device based on the user and device profile. There are four major data processing stages to tune the graphical content: using stylesheets, resizing the graphic, graphic format converting, data compression. Each processing stage can operate alone or can be cascaded to facilitate processing for a specific device. There is also an image renderer for generating image from graphical resources described in graphical markup languages, such as SVG document.

Note that this dataflow is deficient in the sense that there must be a clear flow not only from the Garnet Collaborative Service to mobile user, but also back again to process the user interactive event. A collaborative event in a specific device should interoperate with different types of devices correctly. This can be quite complicated and it is not clear how this is achieved in general with various transformations. For this reversibility problem, the original document is kept in transcoding unit itself, which allows us to access the document from each processing stage for generating the corresponding interactive event for different devices.

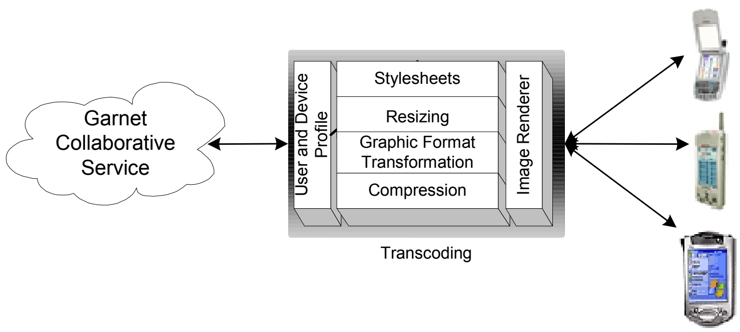


Figure 1. Dataflow for adapting mobile devices

4. Application -aware Transcoding in Garnet

Here we discuss transcoding technology utilized in Garnet to support mobile devices with limited capability. Resource sharing in real-time is performed with events recording state changes transmitted from a "master" instantiation to replicas on other clients in the same session. The resources include visualizations, web pages and Microsoft Word documents and are shared with *shared export* or *shared display* mechanisms on the PDA.

Shared Display is the simplest method for sharing documents with the frame-buffer, corresponding to either a window or the entire desktop, replicated among the collaborating clients. Shared display does not allow significant flexibility; for instance, different clients cannot examine separate viewpoints of a scientific visualization. More flexible sharing is possible by sharing object state updates among clients with each client being able to choose how to realize the update on its version of the object, a process known as the shared export mechanism [1].

The transcoding technology should consider the characteristics of the collaborative feature. Shared display requires fast and efficient propagation upon the master's change of display. Thus reducing the size of data over the wireless network is critical. On the other hand, shared export provides more flexible visualization and needs to ensure quality of the graphic and user interactivity to provide a user's preferred view.

The transcoding process in GMSME inherits Handheld Message Service's (HHMS) [20] transparent application-awareness. When the master of the session changes his or her collaborative feature, the mobile users do not have to initiate new features or fix their profile to adapt new resource. Also the underlying transcoding technology is performed according to the changed features. In this section, we will describe the transcoding technologies used in our collaboration prototypes, shared display and shared export in detail.

The Garnet system is developed with pure JAVA solution. The client application running on mobile devices are also based on JAVA techniques, SUN's J2ME platform [21].

4.1 Transcoding in Shared Display

Since Shared Display is sharing documents with the frame-buffer, modest client dependence is possible with mobile devices, for example receiving a reduced size image. Some collaboration systems support remote manipulation with user interactions on one machine holding a replica frame-buffer transmitted to the instance holding the original object. This is an important capability in help desk or consulting applications, similar to situations that occur frequently in the debugging of code. As this works for all applications without modifying them, this is the basic shared document mechanism in Garnet System [4].

The Garnet system processes image captured from the master's framebuffer in three ways. First, it resizes the images based on the device profile. Table 1 shows a resultant data from a test session with Microsoft Power point. The resolution is customizable on the client side. Second, Garnet system supports graphic format transformation for specific environment, for example *Mobile Information Device Profile (MIDP)* from Sun supports only raw bitmap data and the PNG (Portable Network Graphics) [34] graphic format. To deal with this case, Garnet system generates PNG image for MIDP equipped smartphone client. Finally, it compresses image data to save network bandwidth. Garnet compresses image data with Huffman [35] and LZ77 [36] algorithm. For raw bitmap data, comprising of 8 bit of RGB and alpha, Garnet eliminates alpha bits which represents transparency of image, and reduces the communicating data size to 25%. Table 1 presents the actual data size transformed via a wireless LAN network or 3G communication service as well.

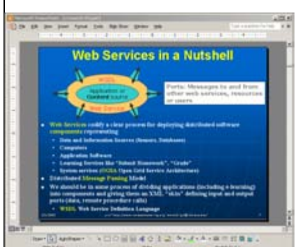
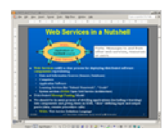
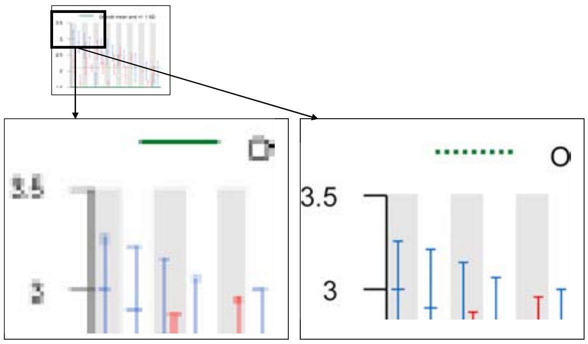
Hardware Specification	Desktop PC 1280 x 1024 LAN SUN J2SE	PDAs (iPaq 3900) 240 x 320 Wireless LAN 802.1b SUN PersonalJAVA	Smartphones (Treo300) 160 x 160 3GWirelessCommunication SUN MIDP
Image			
Image Size (pixels)	1106 x 930	553 x 465	120 x 100
Transmitted data size (KBytes)	4017.9 KB	50.9KB	1.7 KB
Image Format	Bitmap image	Bitmap image	PNG image

Table 1. The resultant data from test session of Garnet shared display

4.2 Transcoding in Shared Export

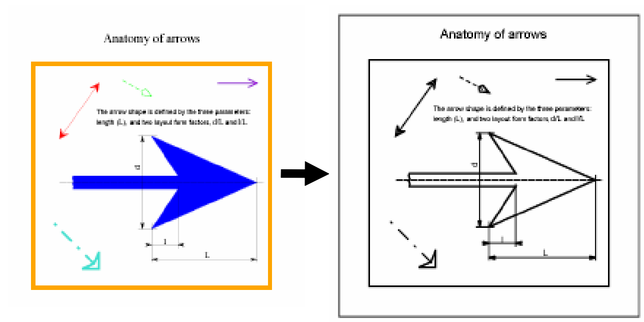
Since shared export allows the users to access more abstract stages of an object, it provides significant flexibility with view points based on user preferences [16]. For instance, different users can browse a display data that is generated from identical data sources but rendered in different way. This is very time consuming to develop if one must do this separately for each shared application. The shared export model filters the output of each application to one of a set of common formats and builds a custom shared event viewer for these formats. This allows a single collaborative viewer to be leveraged among several different applications.

One of major advantage of Garnet System is its supporti for a collaborative viewer based on W3C's SVG format. Scalability implies that each client can resize and scroll while preserving correct positions of pointers and annotations for their various resolutions. As depicted in figure 2, the benefit of the vector graphics format is its scaling/resizing ability. For mobile devices which have limited display size, this ensures flexible resource access based on user preferences. SVG is useful as it is already available for Adobe Illustrator [14] and both PowerPoint and Macromedia Flash are exportable to this syntax. Currently there is a Flash (which is a binary 2D vector graphics format) to SVG converter [37] from the University of Nottingham while OpenOffice.org's *OpenOffice* [38] exports PowerPoint to SVG.



(a) Bitmap Image (b)Vector Graphics

Figure 2. 400 % Zoom In Images



(a) (b)

Figure 3. Styling with CSS for Black and White PDAs

technology such as the use of CSS syntax and properties or XSL. Figure 3 shows the different output images rendered with different stylesheets. The advantage of styling with stylesheet is flexibility in reformatting images. In addition to color adjustment, figure3(b) is transformed by CSS stylesheet, which redefines the width of a brush stroke to be wider and the text style to more recognizable.

To provide individual presentation to each user, Garnet allows every user to own their object instance in shared export. Figure 4 is the object flow of the SVG shared export for each user. Image rendering and transformation are performed in GMSME to reduce the workload in mobile devices. Eventually the ready-to-use image is delivered to PDAs or smartphones. There are several graphic formats supported in mobile devices. SUN's MIDP supports only PNG and raw bitmap graphics, while SUN's Personal Java runtime environment supports JPEG and raw bitmap graphics. Thus, we provide format conversion for specific mobile environments. Some of the graphic formats include data compression mechanisms; however, raw bitmap image data needs additional data compression to better utilize wireless network bandwidth. Data compression capabilities can be fine tuned by individual users based on their needs.

Since shared export is designed for more flexible and high quality resource sharing, stringent timing constraints are not the overriding factors. With the scalability of vector graphics, shared export provides maximized resource sharing to users. The users are able to browse the best quality of image supported by their devices.

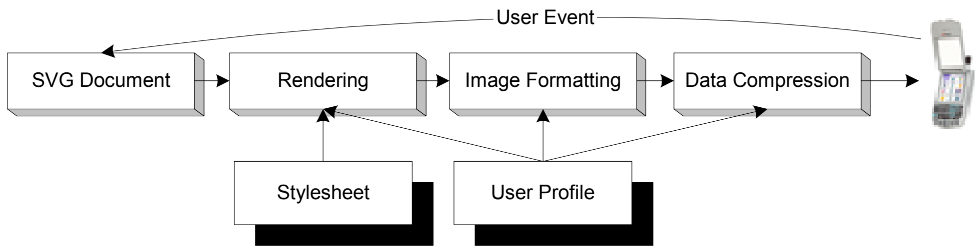


Figure 4. Object flow in SVG Shared Export in Garnet

5. Future work and Conclusion

For future work, the process of adapting content will be developed as a component of a Web service. Performing transcoding in proxy-based middleware may have disadvantages in a large user group. As we presented in Table 1, the actual data delivered to mobile devices are much smaller than the original data from master client. However, the transcoding functions should download whole amount of data to process it, which can incur significant network congestion in proxy-based middleware. WebSphere Everyplace Access is designed to perform its transcoding process in individual WebSphere portlets. This approach enables the transcoding process to be separated from the proxy-like middleware. We also plan to address the separation of transcoding and extend the scheme of message-based collaboration as a Web service.

GMSME provides the accessibility to heterogeneous mobile devices. GMSME adapts shared resources to different hardware profiles of mobile devices and user preferences. Also, the transcoding technologies in GMSME are application-aware. We explained different transcoding strategies used in the two major object sharing methods in GMSME, shared display and shared export.

Finally, the final version of this paper will incorporate results from our ongoing performance measurements. Specifically we will report results pertaining to the performance of our strategies, outlined in this paper, with the communicating entities residing in a heterogeneous networked environment comprising a traditional LAN, a wireless LAN (802.1b) and a 3G wireless communication network.

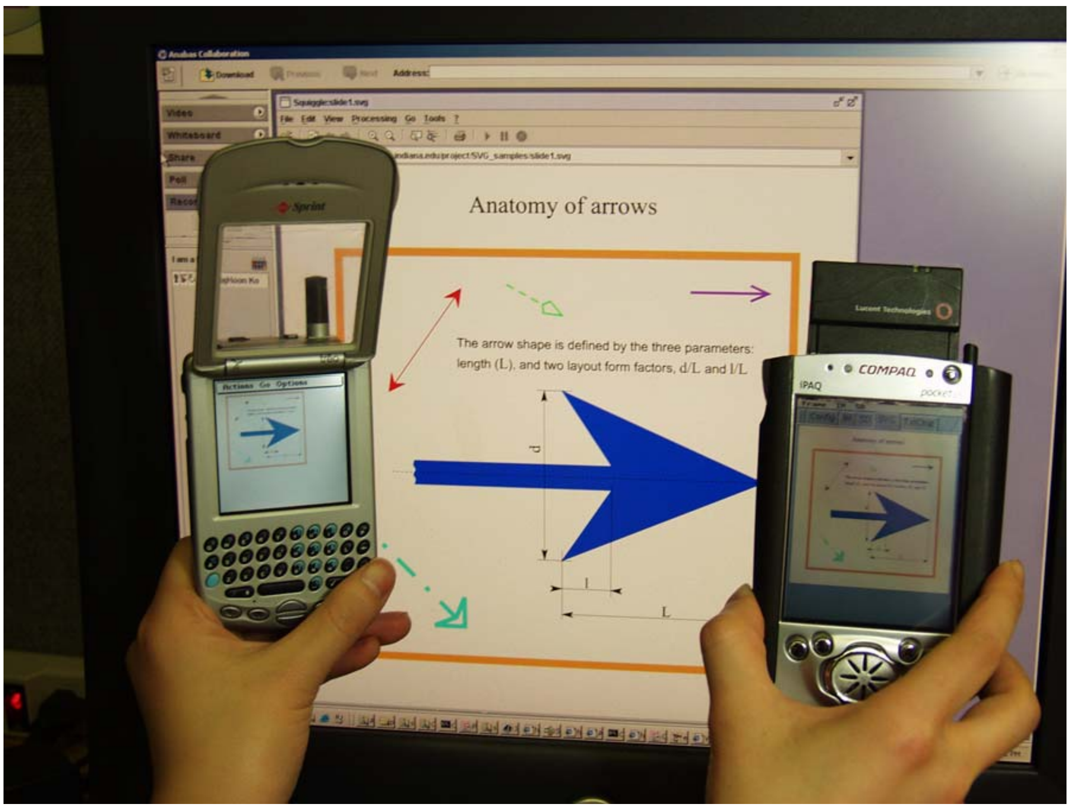


Figure 5. Snapshot of mobile collaboration with Shared Export

References

[1] Geoffrey Fox, Dennis Gannon, Sung-Hoon Ko, Sangmi Lee, Shrideep Pallickara, Marlon Pierce, Xiaohong Qiu, Xi Rao, Ahmet Uyar, Minjun Wang, Wenjun Wu Book chapter on Peer-to-Peer Grids http://grids.ucs.indiana.edu/ptliupages/publications/p2pgridbook.pdf
[2] Sprint, “PCS Vision Style Guide for J2ME”, June, 2002, http://developer.sprintpcs.com/adp/resources.do?redirect=resources&bgcolor=1
[3] IEEE, “International Standard ISO/IEC 15802-2 : 1995 ANSI/IEEE Std 802.1B, 1995 edition”,

http://standards.ieee.org/reading/ieee/std_public/description/lanman/15802-2-1995_desc.html
[4] G. C. Fox. Architecture and Implementation of a Collaborative Computing and EducationPortal. ERDC Technical report, May 2001
[5] I, Marsic, "An Architecture for Heterogeneous Groupware Applications," Proc. of the 23rd International Conference on Software Engineering, pp.475-484, IEEE Computer Society Press, 2001.
[6] Brad A. Myers. "Using Hand-Held Devices and PCs Together," Communications of the ACM. Volume 44, Issue 11. November, 2001. pp. 34 - 41.
[7] Harakan Software, PalmVNC,
http://www.btinternet.com/~harakan/PalmVNC/
[8] R.D. Hill, T. Brinck, S.L. Rohall, J.F. Patterson, and W. Wilner, "The Rendezvous architecture and language for constructing multiuser applications", ACM Transactions on Computer-Human Interaction, 1(2):81-125, June 1994.
[9] M. Roseman, and S. Greenberg, "Building real-time groupware with GrouKit, a groupware toolkit.", ACM Transactions on Computer-Human Interaction, 3(1):66-106, March 1996.
[10] Adobe Systems, Inc. *Portable Document Format*. http://www.adobe.com/products/acrobat/adobepdf.html
[11] W3C, "XSL Transformations (XSLT) Version 1.0", W3C Recommentation, Nov, 1999. http://www.w3.org/TR/xslt
[12] IBM Raleigh Lab., "Transcoding Technology in WebSphere Everyplace Access: Using Transcoding Technology to Expand your Pervasive Portal", IBM WebSphere Developer Technical Journal, Sept, 2002.
http://www7b.boulder.ibm.com/wsdd/techjournal/0209_thrasher/thrasher.html
[13] AT&T Laboratories, *VNC - Virtual Network Computing from AT&T Laboratories Cambridge*,
http://www.uk.research.att.com/vnc/
[14] Nokia, Java VNC viewer, http://www.mgroeber.de/nokia.htm
[15] W3C, Scalable Vector Graphics (SVG), http://www.w3.org/Graphics/SVG/Overview.htm8
[16] Sangmi Lee, Geoffrey Fox , Sunghoon Ko, Minjun Wang, Xiaohong Qiu "Ubiquitous Access for Collaborative Information System Using SVG" SVG OPEN 2002 July 2002 Zurich, Switzerland
[17] Adobe Systems, Inc. *Adobe Illustrator*. http://www.adobe.com/products/illustrator/main.html
[18] SVG and the Electronic Research Group, *Flash to SVG*, http://broadway.cs.nott.ac.uk/projects/SVG/#project
[19] OpenOffice.org , *OpenOffice.* http://www.openoffice.org
[20] Geoffrey Fox, Sung-Hoon Ko, Kangseok Kim, Sangyoon Oh, Sangmi Lee, "Integration of Hand-Held Devices into Collaborative Environments "; proceedings of the 2002 International Conference on Internet Computing (IC-02) , June 24-27 Las Vegas. http://grids.ucs.indiana.edu/ptliupages/publications/PDA_IC2002.pdf
[21] Sun Micro Systems, J2ME, http://java.sun.com/j2me/
[22] Cascading Style Sheets, http://www.w3.org/Style/CSS
[23] Sun Micro Systems, "Mobile Information Device Profile (MIDP)", http://java.sun.com/products/midp/
[24] IBM, WebSpheres, http://www-3.ibm.com/software/info1/websphere/index.jsp
[25] S. Chandra, A. Gehani, C. Ellis and A. Vahdat, Transcoding Characteristics of Web Images, Multimedia Computing and Networking, Jan, 2001, San Jose, CA.
[26] Intel Inc., Intel quickweb, http://www-us-east.intel.com/quickweb/
[27] Oracle Inc., "Oracle portal-to-go; any service to any device", white paper, Oct,1999, http://www.oracle.com/mobile/portaltogo/
[28] IBM, "Websphere transcoding publisher.", 2001, http://www.ibm.com/software/webservers/transcoding.
[29]J.R. Smith, R. Mohan, and C.Li, "Scalable multimedia delivery for pervasive computing." In Proc. of ACM Multimedia, Oct. 1999, pp.131-140, ACM Multimedia, Orlando, Florida
[30] S. Chandra, C. Ellis and A. Vahdat, "Differentiated Multimedia Web Services using Quality Aware Transcoding", InfoCom, 9th Annual Joint Conference Of The {IEEE} Computer And Communications Societies, 2000
[31] Sun Microsystems Inc., "Personal Java 1.0 Specification" , http://java.sun.com/products/midp/
[32] The Apache XML project, COCOON, http://xml.apache.org/cocoon/
[33] W3C, The Extensible Stylesheet Language (XSL), http://www.w3.org/Style/XSL/
[34] W3C, Portable Network Graphics (PNG), http://www.w3.org/Graphics/PNG/
[35] Huffmann, D.," A Method for the Construction of Minimum Redundancy Codes," Proceedings of the IRE 40(9): 1098-1101, 1952
[36] J. Storer, and T.G. Szymanski, "Data Compression via Textual Substitution", Journal of the ACM 29:928-951, 1982
[37] SVG and the Electronic Research Group, *Flash to SVG*, http://broadway.cs.nott.ac.uk/projects/SVG/#project
[38] OpenOffice.org , *OpenOffice.* http://www.openoffice.org