Wireless Internet for The Mobile Enterprise Consortium http://winmec.ucla.edu/

TECHNICAL REPORT/WHITE PAPER
UCLA-WINMEC-2005-110-CONTENT

SUBMITTED TO WIRELESSCOM2005

Hierarchy Content Organization and Delivery for Mobile Engineering Computing

Xiaoyong Su, B.S. Prabhu, Chi-Cheng Chu, Rajit Gadh
UCLA – Wireless Internet for Mobile Enterprise Consortium (WINMEC)
420 Westwood Plaza, Los Angeles, CA 90095
x.y.su, bsp, cchu, gadh@wireless.ucla.edu

Abstract

In this paper, we propose a novel architecture to enable engineering content exchange in mobile computing environment by using a language which is extended from eXtensible Markup Language (XML). To solve the challenges that we are facing in mobile computing environment where network resource and computational resource are limited, we use a hierarchy content representation and on-demand progressive transmission mechanism for an engineering field service automation application, which features with rich graphical engineering content including drawing, schematics, plans, circuits, maps, etc. Since content is delivered and represented in a progressive fashion based on user request, both problems of limited network resource and computing resource are being minimized.

Keywords: Mobile computing, multi-level division, graphics decomposition and composition, on-demand content delivery, engineering graphics.

1. Introduction

Mobile computing and wireless communication technologies are changing the traditional business operations and practices in enterprises. As more and more device vendors support powerful processors such as XScale/wireless MMX chips [1] and Open Multimedia Application Platform [2], latest mobile devices such as PDA, cell phones, smart phones, and handhelds, can support rich multimedia content and enable both enterprises and consumers applications and services.

Among obvious technology advances in mobile computing, there are still a number of issues which need to improve. Foremost issue being the conflict arising out of the need to support ubiquitous mobility and better performance, on account of usually small screens, limited power and computation capability, and constraints on user-input methods. On the other hand, wireless communications is fraught with problems of low bandwidth, intermittent connection, and signal interference, etc. All these factors pose a formidable challenge to mobile engineering computing (MEC) in general and more so when multimedia content is part of the mix [3] or large mount data is presented.

For instance, problems generally don't present when transmitting small size 2D vector graphics files. They are emerged when transmitting large size files, say for example a 2MB DXF file to a GPRS connected cell phone. Ideally, the maximum bandwidth of the GPRS connection is 171kbps. At this rate the total download time Td will be 2000*8/171 = 93.6s. However, the actual download time is far more than theoretical value because usually the bandwidth available is 30-60 kbps based on the device and the implementation of the service by the carrier as well as connection condition. This situation potentially will lead to service interruptions and erroneous downloads leading to higher costs. Table 1 and Table 2 show the downloading and computing comparison between mobile computing and stationary computing.

Table 1. Downloading time comparison (Bandwidths of GPRS and 802.11b are assumed consider various factors)

	20KB	200KB	2MB
GPRS (36Kbps)	4.4s	44s	440s
802.11b (2MBps)	0.08s	0.8s	8s
100M Fast Ethernet	~0	~0	0.16s

Table 2. Time required comparison of rendering JPEG file on mobile device and desktop system

	17KB	165KB	622KB
Mobile Device (iPaq5450, XScale 400MHz, 64M RAM)	<1s	~3s	~9s
Desktop computer (Compaq Evo, 2.4GHz, 512MB RAM)	~0	~0	<1s

Thus for successful mobile engineering content exchange, the current framework will have to take a comprehensive recourse to hierarchy content representation, progressive delivery.

In this paper we report our work on engineering field service automation applications where rich-content is required. Our attempt is to create a Multi-Level modeling and representation concept, and an on-demand progressive delivery method based on XML framework. We use 2D engineering graphics organization and delivery as an example. Same concept can be easily extended to 3D and multimedia content.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 explains the system architecture of our layered 2D vector graphics and delivery concept. Details of layered 2D vector graphics representation scheme forms section 4. Section 5 describes the details of content delivery and interactive operation. Our prototype system and its working are explained in section 6. Finally, we draw conclusions on the research in section 7.

2. Related work

2.1 Multimedia for enterprise

Multimedia would enable strong first impression, mitigating ambiguity and influencing better decision making [4, 5]. With intelligent organization of multimedia content it is possible to add syntax and semantics which can be represented as metadata of the content, which in turn aids content management and location [6, 7]. Compare to plain txt or binary content, multimedia content has more information which may involved in audio, video, voice, 2D/3D graphics, picture, text, animation, etc. To facilitate these rich content in a heterogonous computing environment, which is always a feature of enterprise computing, researchers are trying to find out better way to represent and deliver these rich-content. Chung-Sheng Li, et al develop Infopyramid [8, 9] to represent multimedia content for adaptive delivery in heterogonous computing environment. They present very detail on how to organize general multimedia content and delivery based profile of client device. Andrzej Dabkowski and Anna Maria Jankowska [24] propose a comprehensive framework and multi-tier architecture to solve ubiquitous computing in ERP system. they use a efficient way to access information in ERP database and present data in a way of device context-aware fashion. Tayeb Lemlouma and Nabil Layaïda [25] develop a dynamic content generation and adaptation framework based on NAC [25]. This technology enables customized representation on client.

Since engineering content is data emphasized which is different from general multimedia content, we can not simply make use of these great ideas. Indeed, our work subscribes to these ideas - the choice of content representation standards; the proposed hierarchic representation schema, intelligent delivery and interaction reflect this thinking, and demonstrates a better quality of experience to mobile engineering users.

2.2. Engineering Data Organization and visualization

There are few research works reported related to handling of big 2D graphics files in a MEC. Most of these are for visualizing large data sets in a stationary environment, where there are no significant restrictions. Usually, hierarchical representation [10, 17], server based computing [11, 18] and tree data structure [13, 14, 15] are most commonly used methods.

Abello J., *et al,* [10] present a hierarchic representation for massive multi-digraphs. They propose a method to transform an arbitrary graph into hierarchies by building a hierarchy of multi-digraph layers on top of the input multi-graph. Each layer consists of sets of vertices of previous level. By using the hierarchy structure, they can control the display space efficiently. Jason Leigh and Stuart Bailey [11] propose a methodology for supporting collaborative exploratory analysis of massive data set in tele-immersive environments. In this methodology, a remote rendering server will render the viewpoints as a sequence of stereoscopic images or animations. Then the images and animations will be compressed and streamed to the client. Paula Frederick *et al,* [12] present their study on improving the performance of visualization of large figures. They use a persistent structure that is composed of an R-tree [13, 15] and V-Tree [14] for storing and retrieving 2D spatial data. Bin Pham and On Wong [27] have analyzed the capabilities and limitations of current handheld devices and discuss the important issues to be considered such as data organization, management, communication, input methods and user interfaces.

Among these methods, hierarchic representation has big advantage over the other two methods. But most research on this issue don't mention about how to transmit the content progressively. Server based solutions have to maintain a state for each session and need lots of computational resource on the server side. It is not suitable for a massively distributed computing environment. The tree structure representation method solves the storing and retrieving issue in massive data sets, but it is still hard for mobile devices to consume large data in these cases.

3. Architecture

As mentioned earlier 2D graphics usually is represented in a single file containing lots of information. As a result, to use this content, no matter what format it is in, it is required to be transmitted and read as a single file into the device memory, subject to the bandwidth and computational capability of the MCE. We present a multi-level 2D graphics model in this paper in an attempt to solve these problems. For this we do not intend to create another new format but use existing standard data exchange formats only with a new representation framework. The underlying key concept of our work is to represent content into intelligent granular levels and to deliver the same on demand.

3.1. Graphics decomposition

Based on the properties or syntax of 2D graphics content, we can extract the primitives that have same or similar properties and separate them into files. Here we introduce a three level graphics representation mechanism. Firstly common properties and attributes such as name space description, coordinate system, etc., are extracted and they constitute the top level graphics domain description file or the 'Root'. At the second step, graphics views, profiles and graphics objects are grouped forming Level 1. Groups of low-level primitives form the high-level graphics object in this level. Level 2 constitutes the individual graphic primitives. This process is called as multi-levels division.

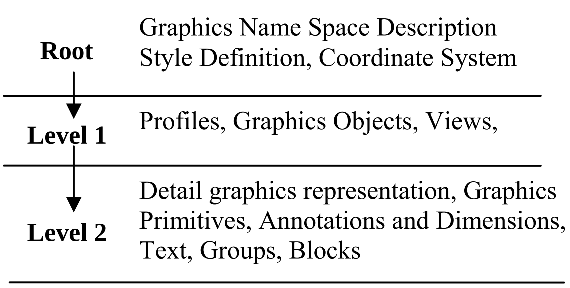

Fig.1. two level graphics decomposition

Figure 1 shows a typical multi-levels division. Root, a single description file, is the entry point of graphics content. This file contains information including graphics description (Vendor, Reference, Name Space, etc) and style definition that will be used for all the primitives in this name space. Level 1 consists of profiles, graphics objects (such as image, symbol, marker and graphics components etc), Views (Multiple views are possible to be used to describe a graphics domain). In this level multiple files will be used to represent all the information. Level 2 consists of detail graphics primitives, Annotations, Dimensions, Groups, Blocks, and Text. Criteria for subdivision can be based on primitive type, region of interest, pre-defined templates, and even user selection.

When the graphics content is broken into different levels, it should be done in such a way that the level of detail achieved is manageable as well as the granularity will suit the MCE. Too much detail will cause too many small size files which would need extra bandwidth. This would provide obvious benefits such as reducing the problems attributed to service interruptions, facilitate short duration transmissions, and only deliver content of interest.

3.2. Tree structure representation

In this section, we will discuss how to decompose single 2D graphics content in detail.

If GN represents a graphics domain, $LN1_i$ represents an object in Level1, $LN2_{ij}$ represents the sub-components of object $LN1_i$ and symbol $\oplus$ represents content merging, we can establish the following relationship:

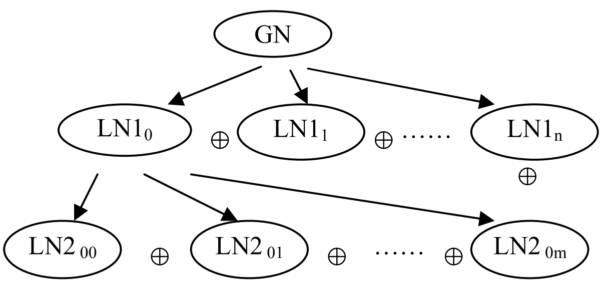

Fig.2. Tree structure of the multi-level division

$GN = \sum LN1_i = LN1_0 \oplus LN1_1 \oplus \ldots\ldots \oplus LN1_n$, where $i \in (0, n)$

$LN1_i = \sum LN2_{ij} = LN2_{i0} \oplus LN2_{i1} \oplus \ldots\ldots \oplus LN2_{im}$, where $j \in (0, m)$

If we represent this relationship in a tree structure, it can be shown as in figure 2. In this tree structure, each node represents a single file or directory. We use a graphics domain which has multi-level tree structure to organize these files. The graphics domain looks like name space in JAVA. For example, if the file name of Level1 node $LN1_0$ is 'PartProfile.udml' (profile of a part in one of the views) and Level2 node $LN2_{01}$ is 'hatch.udml' (hatching of the profile of the part), then the full lookup index of this branch of the tree will be *YOUR_ORGANIZATION.GN.*$LN1_0$*[PartProfile.udml].*$LN2_{01}$*[hatch.udml]*. These nodes can be dispersed over a number of machines in an enterprise network. By using such a method, graphics content can be easily located in a distributed environment.

3.3. An instance of graphics decomposition

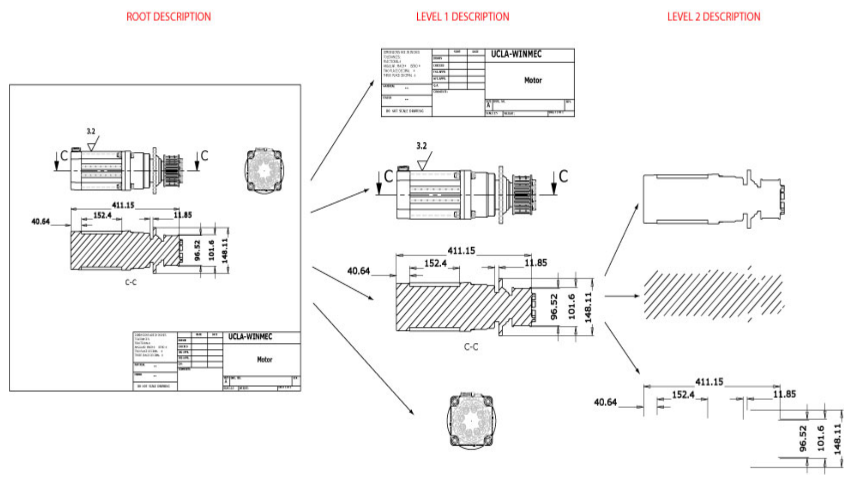

Fig.3. An instance of sub-division sequence and layout

Figure 3 shows an engineering drawing containing Multi-Level information. It has three views, textual annotation, drawing information, dimensions, etc., and each of these can be separated into different nodes (files) containing semantically related information.

Firstly, the common properties such as line style, line width, font face and size, etc., are extracted. At the same time, the first level division will be

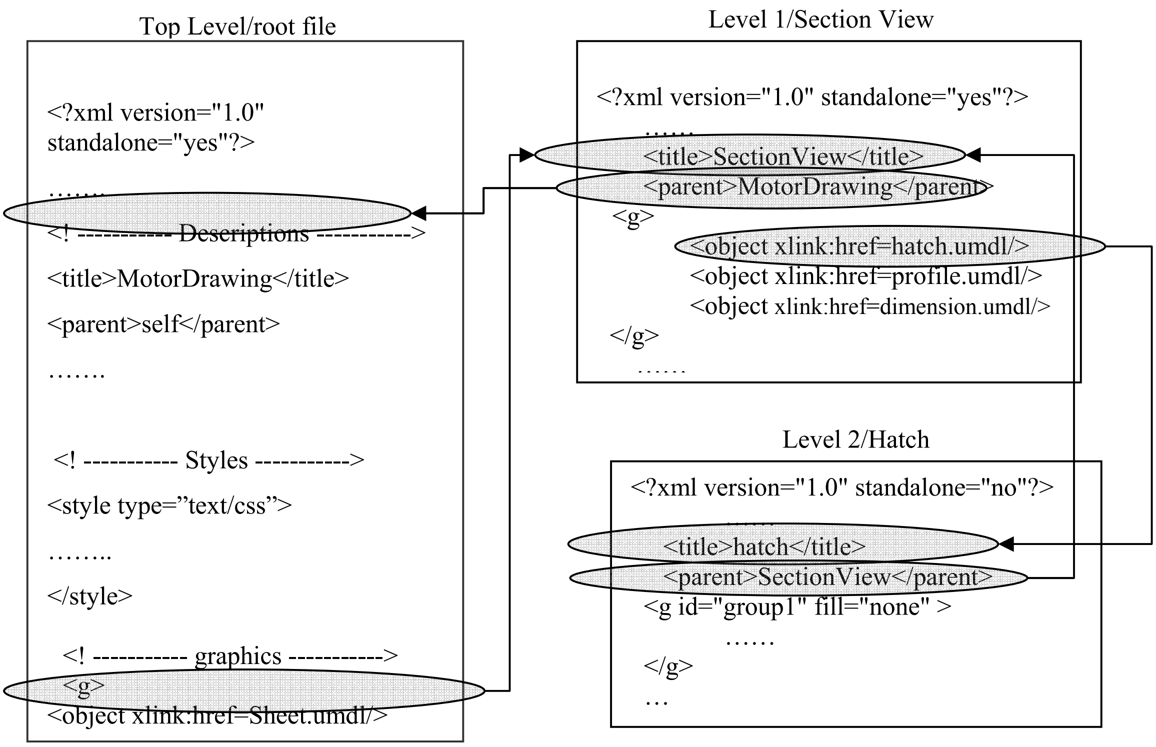

Fig.4. Two-way relationship between content files

conducted. The graphics content will be divided into sheet, main view, left view and top view.

In the process, common properties, along with the general description of this drawing such as vendor, language etc, will be used to form the root and it will constitute the top level description file. Starting from this file, it is possible to transverse the entire graphics content.

In the second stage, the second level division is carried out. In the above example, top view has been divided into profile, hatch and dimension.

If we reverse the whole process, we can use the divided content to regenerate the original drawing.

3.4. Multi-level 2d graphics representation

As we can see from this example, besides the graphics description sentences, there are some other descriptions at the beginning of the file. These sentences are necessary to describe document type, styles and other useful information, but they increase the file size. So, in our design, we try to minimize these descriptions to only the essential in the graphics domain without violating the requirements of the standard thereby providing maximum portability. We use Unified Media Description Language (UMDL) [25], to represent subdivided content.

Figure 4 shows the hierarchical content representation in UDML. As we can see, a two way relationship is been kept to form the hierarchy structure.

4 Content delivery and interaction

To take advantage of this hierarchy content representation method, we propose an on-demand content delivery and interaction architecture. Two steps of session will be used to represent conversation and interaction between client and content server. First step is called system session and second step is called instance session.

System session starts at the request for any graphic content and begins with the downloading and parsing of the Root of the graphics domain. This information will be stored on the client until the client closes the session and allows the user to navigate to the other levels of the content. Instance session is established when a particular graphics content mapped to links in the root of the graphics domain is requested by the client. This session is maintained only during the file transmission. Session ends when the downloading of graphic content is completed.

Content can be delivered in two ways. The first one is 'sequential delivery'. Once the root is loaded into the memory, system will automatically use the links in the root to discover and download the contents of next level. The second method is 'on demand delivery', where the root is downloaded first, but the system won't download the content of the next level. In turn, the system provides a list of all the content of the

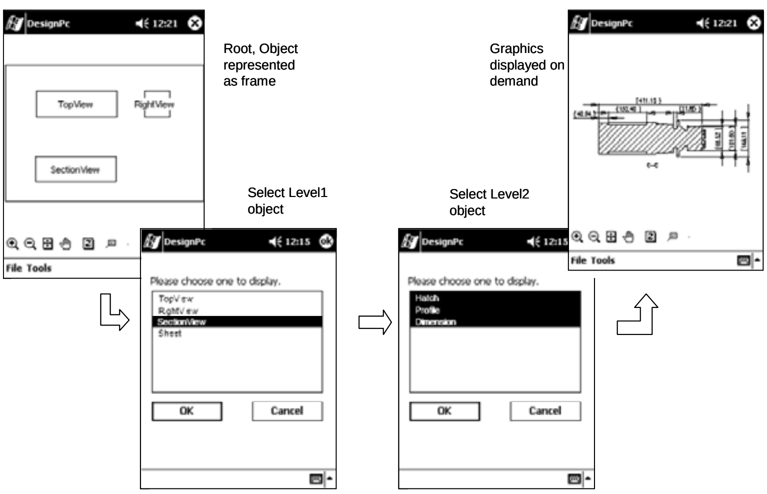

Fig.6. Prototype of 2D graphics representation and rendering

next level which can be selected as per the user need. The type of delivery can be chosen by the client as required. During the delivery process, the client's profile will be used to decide which level of content, in what detail and in which format should it be delivered.

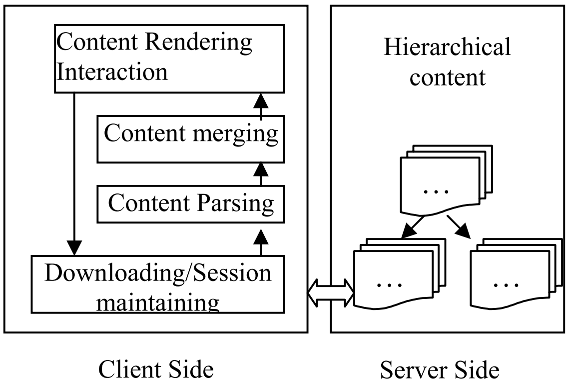

Fig.5. Graphics content delivery and process

Figure 5 shows the schematic of the delivery framework. In this framework, content delivery is initiated by the client request. As mentioned before, by subdividing large content into interconnected multiple small files, content server doesn't need to maintain the state of whole interaction/transaction. Content adaptation and rendering is finished at the client side. The process of delivery is stateless. Advantage of this delivery method is that the server can support more clients at the same time.

Server maintains the graphics domain and listens to client request. Client has four major components: Downloading/Session Maintenance, Content Parsing, Content Merging, and Content Rendering/Interaction. Downloading/Session Maintaining module maintains the conversation and sends content request to server and handles the downloaded content. Content Parsing module will read the content into memory and build a DOM structure for each individual file. Content Merging module merges the different content together. Two of the Content Merging methods will be discussed later. Content Rendering and Interaction module enables content visualization and interaction on the client. It directly controls the Downloading and Session Maintaining module. For example, user can start or end a session, request a portion of content, etc.

5 Prototype

Based on the framework presented above, a prototype is being implemented, whose overall functionality is shown in Figure 6.

6 Conclusion

The purpose of this research is enable engineering content exchange in mobile computing environment. In this attempt we have proposed a XML based intelligent, distributed, hierarchic and concise 2D graphics content representation and progressive delivery architecture.

The proposed architecture employs a multi-level division method to dissect the graphics content into multiple levels using a variety of rules and strategies. Each level can be delivered on demand as against the delivery of entire monolithic file. Currently, we have implemented the framework. The system has performed satisfactorily in representing different engineering graphics content (drawings, schematics, layouts), which is quite encouraging for field service applications - one of the main focus areas of enterprise application research at WINMEC.

A prototype based on the framework is being developed wherein modules for real-time device based content adaptation, personalized user interactivity, persistent session management and migration, and content synchronization modules will be designed and implemented.

Acknowledgements

We are pleased to acknowledge the support of Intel Corporation, and Wireless Internet for the Mobile Enterprise Consortium (WINMEC), UCLA towards partial funding for this research.

References

[1] PCA Overview, Intel, http://www.intel.com/pca/developernetwork/overview/index.htm
[2] OMAP for 2G and 3G: Overview, http://focus.ti.com/docs/apps/catalog/overview/overview.jhtml?templateId=1106&path=templatedata/cm/level1/data/wire_omap_ovw
[3] R. Rosenbaum, H. Schumann, and Ch. Tominski, Presenting Large Graphical Contents on Mobile Devices – Performance Issues, in: P.C. Deans (ed): E-Commerce and M-Commerce Technologies, IRM Press, Hershey, London, 2004.
[4] Kai H. Lim, Izak Benbasat, and Lawrence M. Ward, The Role of Multimedia in Changing First Impression Bias, Information Systems Research, Vol. 11, No. 2, June 2000, pp. 115–136.
[5] *William I. Grosky, Ramesh Jain, Rajiv Mehrotra*, The Handbook of Multimedia Information Management, Prentice Hall, New Jersey, USA, 1997.
[6] Ramesh Jain, Semantics in Multimedia Systems, The 9th International Conference on Multi-Media Modeling (MMM '03), January 7-10, 2003, Taipei, Taiwan.
[7] A. Gunasekaran, and P.E.D. Love, Current and future directions of multimedia technology in business, International Journal of Information Management 19, 2, April 1999, pp. 105-120.
[8] Rakesh Mohan, John R. Smith, Chung-Sheng Li, Adapting Multimedia Internet Content for Universal Access, IEEE Transaction On Multimedia, Vol. 1, No. 1, March 1999
[9] C-S. Li, R. Mohan, and J. R. Smith, Multimedia content description in the InfoPyramid, Proc. ICASSP'98, Special Session on Signal Processing in Modern Multimedia Standards, Seattle, WA, May 1998
[10] James Abello; Jeffrey L. Korn; MGV: A System for Visualizing Massive Multidigraphs. IEEE Transactions on Visualization and Computer Graphics 8(1): 21-38 (2002)
[11] Leigh, J.; Johnson, A.E.; DeFanti, T.A.; Bailey, S.; Grossman, R.; A methodology for supporting collaborative exploratory analysis of massive data sets in tele-immersive environments, Proc. of The Eighth International Symposium on High Performance Distributed Computing, Aug 3-6, 1999
[12] *Frederick, P.; Gattass, M.; Mediano, M.R.*. Efficient visualization of graphical objects, Proc. of XII Brazilian Symposium on Computer Graphics and Image Processing, Oct 17-20. 1999.
[13] Timos Sellis, Nick Roussopoulos, and Christos Faloutsos. The R+-Tree: A Dynamic Index for Multi-Dimensional Objects. In *Proceedings of the 13th VLDB Conference*, pages 507–518, September 1987.
[14] Maur´ıcio Riguette Mediano, Marco Antonio Casanova, and Marcelo Dreux. V-tree - a storage method for long vector data. In *Proceedings of the 20th VLDB Conference*, 1994.
[15] A. Guttman. R-trees: A dynamic index structure for spatial searching. In *Proceedings of the ACM SIGMOD Conference on Data Engineering*, pages 47–56, 1984.
[16] B. S. Prabhu; S. S. Pande. Intelligent interpretation of CADD drawings. Computers & Graphics 23, pages 25-44, 1999
[17] Leila De Floriani, Enrico Puppo. Hierarchical triangulation for multiresolution surface description, ACM Transactions on Graphics (TOG), Volume 14 Issue 4, 1995
[18] Jin Jing, Abdelsalam Sumi Helal, Ahmed Elmagarmid, Client-server computing in mobile environments, ACM Computing Surveys (CSUR), Volume 31 Issue 2, 1999
[19] Andres Baravalle, Marco Gribaudo, Vitaveska Lanfranchi, Using SVG and XSLT for graphics representation, Proceeding of Open SVG 2003, 2003
[20] Chang, Yi-Hong & Chuang, Tyng-Ruey . *Online Aggregation and Visualization of Census Data: Population Mapping with SVG, XML, and Free Software.* In: Proceedings of SVGOpen 2002, Zurich, Switzerland.
[21] Lewis, C.T., Karcz, S., Sharpe, A., Parki, I.A.P.*BioViz: Genome Viewer. Development of an SVG GUI for the visualization of genome data.* In: Proceedings of SVGOpen 2002, Zurich, Switzerland.
[22] *Chaitan Baru, Amit Behere, Charles Cowart* , Representation and Display of Geospatial Information: A Comparison of ArcXML and SVG, Second International Conference on Web Information Systems Engineering (WISE'01) Volume 2, December 03 - 06, 2001
[23] Arei Kobayashi, Satoru Takagi, Naomi Inoue, Extensions of SVG for human navigation by cellular phone, Proceedings of the SIGGRAPH 2003 conference on Web graphics: in conjunction with the 30th annual conference on Computer graphics and interactive techniques , 2003
[24] Jankowska, A.M. (with Dabkowski, A.): Comprehensive Framework for Mobile ERP System; in: Proceedings of the 14th International Workshop on Database and Expert Systems Applications (DEXA03), 2003, Prague, pp. 890-895.
[25] Lemlouma, T., Layaida, N., Content Adaptation and Generation Principles for Heterogonous Clients, OPERA Project, INRIA Rhone Alpes, Position Paper for the W3C Workshop on Device Independent Authoring Techniques, July 2002
[26] Xiaoyong Su, B.S. Prabhu, Chi-Cheng Chu, Rajit Gadh, Middleware for Multimedia Mobile Collaborative System, Third annual wireless telecommunications symposium (WTS 2004), May 14-15, 2004, CalPoly Pomona, Pomona, California, USA
[27] Binh Pham and On Wong, Handheld Devices for Applications Using Dynamic Multimedia Data, Proc. of the 2nd Intl. Conf. on Computer graphics and interactive techniques in Australasia and South East Asia, June 15-18, 2004, Singapore, pp 123-130.