January 2004

Dynamic multimedia adaptation and updating of media streams with MPEG-21

L. Rong
University of Wollongong

I. Burnett
University of Wollongong, ianb@uow.edu.au

Dynamic multimedia adaptation and updating of media streams with MPEG-21

Abstract

The paper discusses media streaming using dynamic resource adaptation and update as a means of facilitating universal multimedia access (UMA): the concept of accessing multimedia content through a variety of possible schemes (Bormans, J. et al., IEEE Sig. Process. Magazine, 2003). As background, the paper summarizes the most common content negotiation approaches and addresses their facets and problems. MPEG-21, the multimedia framework and its relationship to UMA are then explained. Our primary focus is an end-to-end approach to content adaptation which takes advantage of MPEG-21 to facilitate the UMA concept in a media streaming environment. The concept is validated using a media streaming test-bed which provides for wide adaptation according to broad usage descriptions.

Keywords

multimedia communication, MPEG-21, UMA, content negotiation, dynamic multimedia adaptation, dynamic resource adaptation, dynamic updating, media stream updating, universal multimedia access

Disciplines

Physical Sciences and Mathematics

Publication Details

This article was originally published as: Rong, L & Burnett, I, Dynamic multimedia adaptation and updating of media streams with MPEG-21, Proceedings of the First IEEE Conference on Consumer Communications and Networking (CCNC 2004), January 2004, 436-441. Copyright IEEE 2004.

Dynamic multimedia adaptation and updating of media streams with MPEG-21

Letian Rong and Ian Burnett
University of Wollongong, CRC SIT (Smart Internet Technology)
lr98@uow.edu.au, i.burnett@elec.uow.edu.au

Abstract

This paper discusses media streaming using dynamic resource adaptation and update as a means of facilitating Universal Multimedia Access (UMA) [1]: the concept of accessing multimedia content through a variety of possible schemes. As background to our work, the paper summarizes the most common content negotiation approaches and addresses their facets and problems. MPEG-21 the multimedia framework and its relationship to UMA is then explained. The primary focus of the work is an end-to-end approach to content adaptation which takes advantage of MPEG-21 to facilitate the UMA concept in a media streaming environment. The concept is validated using a media streaming test-bed which provides for wide adaptation according to broad usage descriptions.

1. Introduction

Today, wireless users increasingly expect to have access to the Internet anywhere, at anytime through a ubiquitous mobile computing environment. The inevitable outcome is that there will be numerous devices each with their own set of distinct features. This will allow users to access information in various forms. Resources will need to be customized and dynamically adapted to both the current usage environment and user preferences so as to provide them with seamless access. This is especially important in a mobile environment where users can change their environment (e.g., locations, devices etc) dynamically.

MPEG-21 addresses the requirements of UMA by providing a normative open framework for multimedia delivery and consumption. It is used by all the players in the delivery and consumption chain [2].

This paper focuses on the Digital Item Adaptation (DIA) Tools part of MPEG-21 which specifies metadata and mechanisms related to the usage environment and to media resource adaptability. These tools are the basis for the proposed test-bed. Since the scope of MPEG-21 is very broad, interested readers are referred to references for information on the other areas of MPEG-21 [1, 2, 3].

2. Related work

In terms of dynamic media resource adaptation and content negotiation, many approaches have been proposed as solutions to problems in the field. The most well-known of these are now discussed.

2.1. HTTP 1.1 content negotiation mechanism

HTTP 1.1 specifies that there are three possible kinds of content negotiation mechanisms in HTTP. They are server-driven, agent-driven and transparent negotiations [4]. A server-driven approach is based on the theory that the server is in charge of selecting the best representation according to an algorithm located at the server, while the agent-driven approach performs the selections on the client's side. Transparent negotiation is a combination of the two types of negotiations above [5]. Apache [6] has implemented a negotiation algorithm based on the server-driven approach by using a simple weighted-sum mechanism to perform multiple tags matching. Other similar approaches have also been implemented and verified [7].

2.2. Composite Capability/Preference Profiles (CC/PP) and User Agent Profile (UAProf)

CC/PP[8] and UAProf[9] are two standards that have been created for describing delivery context based on the Resource Description Framework (RDF) [10]: CC/PP was created by the W3C and UAProf was created by the WAP Forum. CC/PP is intended to be used to describe device capabilities and user preferences for guiding the adaptation of content presented by a device, while UAProf is an earlier implementation of RDF which was specifically designed to describe device capabilities of wireless devices.

0-7803-8145-9/04/$17.00 ©2004 IEEE.

2.3. 3GPP Packet Switched Streaming service (PSS)

3GPP PSS is a standard which defines a multimedia streaming service within 3G wireless networks. It covers the protocols and codecs which are necessary for the delivery of streaming content to multimedia terminals in Third Generation (3G) wireless networks. It specifies that how the server and terminal must behave in order to enable the streaming service [11]. Release 5 of PSS which contains the capability exchange ability has not yet been released.

2.4. Analysis and motivation

All the technologies mentioned in section 2.3 are related to the idea of UMA, yet, they each only cover one or a few specific domain(s) of the concept. For example, of HTTP 1.1, the content negotiation specification is tied to the HTTP protocol [4] and its implementation usually only supports a small range of attributes for the negotiation process (eg., mime type and language) [6].

CC/PP was designed to solve this limitation by introducing more complex descriptors for describing device capabilities and user preference. Therefore it has a tight coupling with HTTP, even though it is protocol independent. However there is no normative procedure in the standard to describe how these descriptors can be adapted for the negotiation and adaptation process. For example, it does not describe how clients and servers can perform adaptations based on their capability information and how to relate resource adaptation with its usage environment. This information is crucial to the resource adaptation process. There have been several proposals that attempt to address some of the issues in the framework [12, 13, 14, 15]. Yet none of them cover the whole UMA scope.

UAProf and 3GPP PSS are two standards that are specifically designed for wireless devices and applications, and they cover a limited scope within only the wireless domain.

Furthermore, currently most of the content negotiation and adaptation approaches involve complex algorithms or classifications for matching usage environment attributes with resources [16, 17, 18]. Thus, a more comprehensive framework is required to "fill the gaps" and fulfill all the necessary requirements of the UMA concept. The following work is motivated by trying to address those missing links within the MPEG-21 framework. In addition, the end-to-end approach that we propose here will demonstrate how content adaptation can be performed neatly under MPEG-21.

3. MPEG-21 multimedia framework

3.1. MPEG-21 and Digital Item (DI)

The fundamental unit of distribution and transaction in the MPEG-21 framework is the Digital Item (DI). It can be considered as a structured digital object which consists of resource(s) (e.g., a photo album, a web page) and related information on the manipulation of the resource(s) (e.g., terminal capabilities, intellectual properties). DI is represented as a Digital Item Declaration (DID) in MPEG-21 through the Digtial Item Declaration Language (DIDL) [19] which conforms to the XML standard. Therefore it is important to clarify that DI as the basic conceptual unit under MPEG-21 and DID as the XML representation of it.

Currently, a DI usually consists of resource(s), a list of choices that correspond to the various adaptation aspects of those resources and Digital Item Adaptation (DIA) information which steers the adaptation process. The DIA concept is explained in the next section.

It is also possible however for a DI to be valid without including any resources. The reason for this is that the DI can function as a wrapper for the various other parts of MPEG-21 which may not relate to particular resource(s). This ability enables these parts to be combined or exchanged in an interoperable manner under the same multimedia framework. The various parts include elements that conform to Intellectual Property Management and Protection (IPMP), Rights Expression Language (REL), Rights Data Dictionary (RDD) and Digital Item Adaptation (DIA) tools. An example would be a Digital Item which only includes terminal capability information of a device.

3.2. Digital Item Adaptation (DIA)

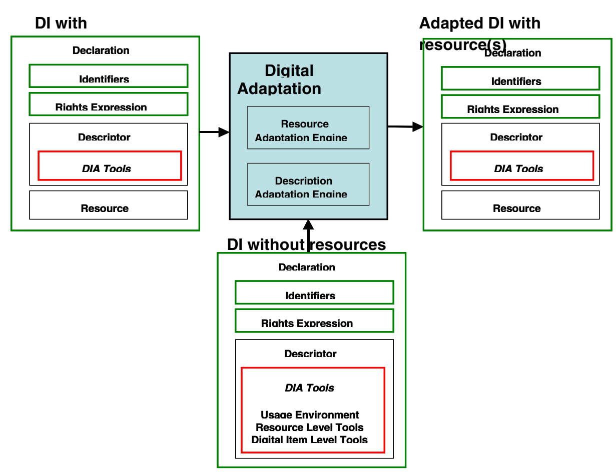

Figure 1: Digital Item Adaptation architecture [20]

One of the main goals of MPEG-21 is to provide solutions for universal multimedia access (UMA). That led to the creation of a new part in MPEG-21: called Digital Item Adaptation (DIA).

Currently, DIA is the most active part of the MPEG-21 standardization. It aims to standardise tools for adapting resources on the basis of usage environment descriptions to produce a modified Digital Item. Figure 1 illustrates the relationship of DIA to the other MPEG-21 components and the overall scope of DIA.

3.3. Digital Item Adaptation tools

DIA Tools is a collection of descriptions and format-independent mechanisms that provide support for Digital Item Adaptation. The descriptors are represented in XML and can be either wrapped in a DID or be used independently. The DIA Tools are further clustered into three major categories as illustrated below [20]:

Usage Environment Description Tools

- User Characteristics
- Terminal Capabilities
- Network Characteristics
- Natural Environment Characteristics

Digital Item Resource Adaptation Tools

- Bitstream Syntax Description (BSD)
- Adaptation QoS (AdaptationQoS)
- Metadata Adaptability

Digital Item Declaration Adaptation Tools

- Session Mobility (SM)
- DIA Configuration (DIAC)

Figure 2: DIA tools architecture [20]

The *Usage Environment Description Tools* includes descriptors to describe various dimensions of the usage environment: namely user characteristics, terminal capabilities, network characteristics and natural environment characteristics.

The *Digital Item Declaration Adaptation Tools* describes how Digital Item Declarations can be adapted as a whole. The 2 main tools currently are: Session Mobility (SM), DIA Configuration (DIAC). SM describes the configuration state information that pertains to the consumption of a Digital Item in real time, so that a Digital Item can be consumed continuously when it is transferred from one device to another. DIAC specifies how and where the related usage environment information can be used for the adaptation of a DI.

In contrast to Digital Item Declaration Adaptation Tools, *Digital Item Resource Adaptation Tools* contains tools that are useful for the adaptation of resources contained within a Digital Item. The two main tools defined are the Bitstream Syntax Description (BSD) language and Adaptation QoS (AdaptationQoS) schema. BSD language describes the high level structure of bitstreams so that bitstreams can be modified according to their BSD information. Furthermore, BSD is independent of specific media resource formats which allows resource adaptation engines to manipulate bitstreams without requiring any additional information. AdaptationQoS schema describes the relationship between QoS constraints (e.g., on network bandwidth or a terminal's computational capabilities), feasible adaptation operations satisfying these constraints and associated media qualities that result from adaptation. It provides the means to associate QoS parameters with their constraints and their values. BSD can be used in conjunction with AdaptationQoS to enable terminal and network Quality of Service by transforming bitstream resources as the former describes how bitstreams can be modified and the latter can be used to steering the modification through its QoS parameters.

3.4. Digital Item Adaptation engine

As shown in Figure 1, the Digital Item Adaptation Engine takes both a normal DI and a DI with metadata as inputs and produce an adapted DI as the output. The engine operates on the input DIs according to the DIA tools may carried in the normal DI itself, as well as the DIA tools carried in the metadata DI. It is further divided into two sub-engines: Description Adaptation Engine and Resource Adaptation Engine.

The Description Adaptation Engine is responsible for adapting DIA descriptors within DIs, while the Resource Adaptation Engine performs adaptation on the corresponding resources. The DIA tools provide information and drive the adaptation for both adaptation engines. The result is an adapted DI which may contain original DI contents, as well as adapted DIA elements, Resources, DID Identifiers and IPMP/REL elements.

4. MPEG-21 test-bed architecture

The architecture for our media streaming test-bed is built based on the DIA concept explained in section 2. The three parts within the DIA tools architecture collaborate to enable dynamic adaptation and updating of media streaming under the framework of MPEG-21. It also functions as a simulator for validating some of the current and future possibilities in the framework. The architecture is illustrated as below:

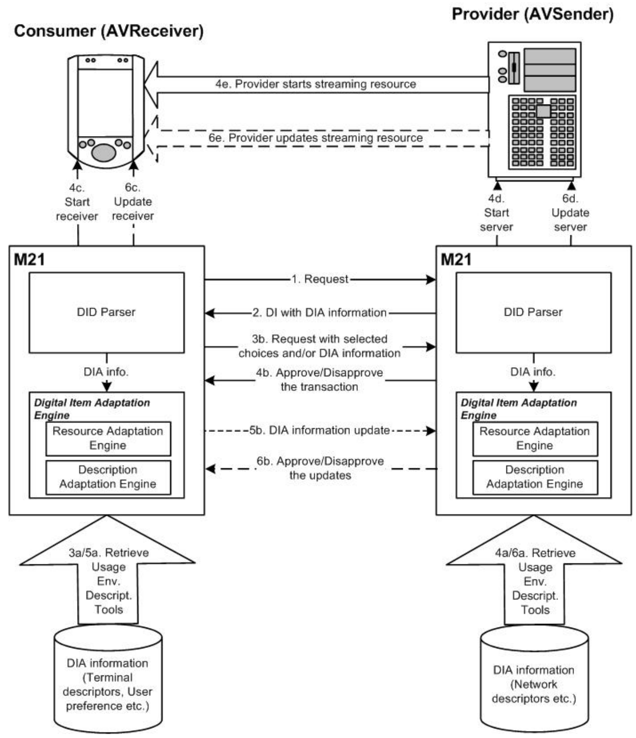

Figure 3: multimedia streaming test-bed architecture

The core part of this test-bed is the middleware (M21) which is inserted below multimedia applications. This has the advantage of allowing many types of multimedia applications to be used without actually modifying them individually to our architecture. M21 communicates with those applications through APIs.

M21 is consists of a DID parser and a DIA adaptation engine and all of its components are written in Java. The Xerces XML parser is used for parsing of XML documents under the Java implementation. This middleware is used on both the consumer and provider side to reflect the peer-to-peer nature and ubiquitous User concept of MPEG-21; a User is any entity that interacts within the MPEG-21 environment and/or makes use of digital items [1].

The RTPTransmitter and RTPReceiver are new media streaming applications for transmitting and viewing streamed media under RTP/RTCP. Both are constructed on top of the Java Media Framework (JMF). The current version of the programs supports multiple synchronized/unsynchronized media streaming between multiple client and server pairs with dynamic bit-rate change and frame positioning.

M21 performs the underlying dynamic content adaptation process through its usage of DIs and configures and launches the RTPTransmitter and RTPReceiver automatically when the process is completed. Also, M21 notifies these applications when the receiving platform usage environment changes during the transmission.

4.1. Steps involved in the media streaming process

There are 6 steps involved in this dynamic media adaptation and streaming process. These are shown diagrammatically in Figure 3 and will be given here in greater detail:

Step1. The consumer sends a request to the provider for a Digital Item representation of a streaming media resource (e.g., a movie). If a similar request was issued in the past, the required data might be stored in local cache. This might enable the consumer's request to jump straight to step 3. The request is sent in the form of a Digital Item, however other possible approaches are to deliver them through pre-existing file-delivery protocols (HTTP, FTP etc., e.g., http://www.abc.com/movie.DI).

Step2. The provider sends the requested DI to the consumer. The DI should contain a link or links to multiple pre-existing variations resources and a list of choices which correspond to the resources or which can be selected so as to configure those resources further. The choices can be selected by a user manually or by a user agent automatically. Furthermore, information related to DID Adaptation tools (e.g., DIAC, DIDCP) are also encapsulated in the DI to provide further guidelines on the adaptation process. The guidelines specify information on the locations of adaptations (at consumer, intermediate nodes or provider), the types of descriptors required for adaptations and how the choices should be selected (by the device automatically or by the user manually). The depth of complexity of the contents of a DI is dependent on the author.

Step3. The consumer device retrieves metadata DIs from a local database which contains Usage Environment descriptors such as terminal descriptors, natural environment descriptors and user preferences. The device then performs adaptation related to choice selections (DID configurations) based on all the resource adaptation related information (e.g., DIA tools) in the received DI and the metadata DI (*Step3a*).

The consumer device requests a resource from the provider device. This may involve the transfer of required DIA information to the provider prior to or as part of the resource request (*Step3b*).

Step4. The provider device receives the request and retrieves Usage Environment descriptors giving information regarding network descriptors (e.g., current network bandwidth) from the local database. Again, this information is represented in the form of a metadata DI. The device then determines the appropriate resource adaptation based on the request from the consumer, the

0-7803-8145-9/04/$17.00 ©2004 IEEE.

metadata DI and contents of the DI sent to the consumer in *step1* (*Step4a*).

The provider sends a metadata DI which contains acknowledgement information back to the consumer through any underlying application level protocols (e.g., HTTP, FTP or SOAP) (*Step4b*).

If the acknowledgement includes approval of the request, the consumer launches the corresponding program (e.g., RTPReceiver). Otherwise, the transaction terminates here (*Step4c*).

The provider notifies the streaming server to stream the adapted resource (*Step4d*).

The streaming server begins streaming the adapted resource to the consumer (*Step4e*).

Step5a and 5b. The consumer transfers an updated request with updated Usage Environment descriptors and/or new adaptation related choice selections (DID configuration information) to the provider according to information from the DIAC tool (e.g., locations of adaptations, etc) held in the received DI in *Step2*. Furthermore, the Session Mobility tools can be used to transfer the current session state of the media streaming to a new device if the consumer changes his/her viewing device.

Step6. The sub-processes of step 6 are similar to *Step4*. The provider determines the appropriate resource adaptation based on the updated information in the DIA tools. It sends an acknowledgement back to the consumer and the currently streaming resource is modified.

4.2. Experiment settings and validation

This new end-to-end approach for media stream adaptation on the basis of contextual descriptors is verified by installing RTPTransmitter and RTPReceiver on two separated machines which both have M21 installed as their underlying support middleware. The two machines are connected through a 100Mbps half-duplex local area network.

Users on the consumer side (e.g., RTPReceiver installed) can open a DID through the M21 middleware containing a sample movie menu and a User may then choose which movie they would like to watch before sending a request to the provider (e.g., RTPTransmitter installed) (*step1*).

When the provider receives the request, it sends back a DID which contains the URL to the resource, a list of choices related to the adaptation of the resource and some additional information of DIA tools as specified above. The current choice selections include: VideoQuality, Coarse Language, Supported Format and Language (*step2*).

When the consumer receives the DID, M21 prompts the user to make selections on VideoQuality and Coarse Language, while Supported Format and Language are selected by M21 automatically according to the terminal descriptors of the consumer device. This selection of choices demonstrates that a variety of usage environment descriptors can be incorporated into the middleware and this choice list can be easily extended. The consumer then sends the second request to the provider after the choices on VideoQuality and CoarseLanguage are made (*step3*).

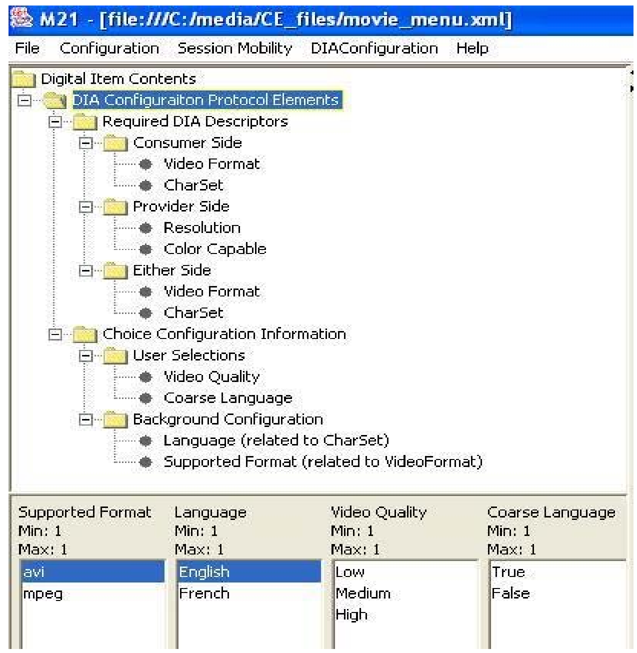

Figure 4: M21 screenshot during the choice selection process in step3.

When the provider receives the second request, it sends a positive acknowledgement back to the consumer if the requested movie resource can be adapted and then it lauches RTPTransmitter to transmit the movie. If the requested movie resource can not be adapted, then a negative acknowledgement is sent back and no streaming applications will be launched. When the consumer receives a positive acknowledgement, they launch the RTPReceiver application to start receiving the streamed movie (step4).

Dynamic resource updates currently allow the user to change the VideoQuality choice during the streaming of a movie. This single choice can easily be extended to include other choices and/or usage environment descriptors for dynamic resource update (step5 and step6).

5. Conclusion

Our proposed end-to-end approach to contextual adaptation of media streaming demonstrates how media streaming with dynamic adaptation and updating can be performed under MPEG-21. This approach utilizes the

Digital Item Declaration and Digital Item Adaptation Tools to support the UMA concept for multimedia streaming through heterogeneous terminal devices. Using this approach, streamed media can be easily adapted to its current usage environment and consumers have the ability to automatically and/or manually update the media stream when their usage environment changes. Furthermore, this approach can be simply extended to cater for media resources (e.g., image, graphics and web pages etc.) in other application domains. In contrast to other approaches, it includes adaptation of real-time streams on the basis of device and environmental descriptors without complex content matching algorithms.

In future versions, it is planned to expand the test-bed into a consumer-gateway/proxy-provider architecture to verify current elements and exploit new adaptation strategies. Also, the test-bed will be expanded to include security features from the Digital Right Management (DRM) and (Intellectual Property Management and Protection (IPMP) sections as they become more mature.

In conclusion, MPEG-21 offers a broad and complete multimedia framework that enables the delivery and consumption of multimedia content in an interoperable manner. It overcomes some of the critical issues faced by other standards by providing all the necessary "ingredients" for a highly automated, usage environment aware, secure multimedia transaction. Furthermore, MPEG-21 enables content creators and service providers to customize multimedia content and provides consumers with a large variety of media choices in an interoperable way.

6. References

[1] Jan Bormans, Jean Gelissen, and Andrew Perkis, "MPEG-21: The 21st Century Multimedia Framework", *IEEE Signal Processing Magazine*, March 2003.

[2] Andrew Perkis, Yousri Abdeljaoued, Charilaos Christopoulos, Touradj Ebrahimi, "Universal Multimedia Access from Wired and Wireless systems", *Birkhauser Boston, transactions on Circuits, Systems and Signal Processing; Special issue on Multimedia Communications*, Vol. 20., No.3, 2001, pp. 387-402.

[3] Jan Bormans, "MPEG-21 Requirements v 1.4", *ISO/IEC JTC1/SC29/WG11/N5333*, Awaji Island, Japan, December 2002.

[4] The Internet Society, "RFC 2616: HTTP 1.1 Content Negotiation", ftp://ftp.isi.edu/in-notes/rfc2616.txt, June, 1999, pp. 70-73.

[5] K. Holtman and A. Mutz, "Transparent Content Negotiation in HTTP", *IETF RFC 2295*, 1998.

[6] Apache, "Apache HTTP Server: Content Negotiation", http://httpd.apache.org/docs/content-negotiation.html.

[7] S. Seshan, M. Stemm, R. H. Katz, "Benefits of transparent content negotiation in Http", *Global Internet Mini Conferenceat Globecom 98*, Nov, 1998.

[8] World Wide Web Consortium, "Composite Capabilities/Preferences Profile", http://www.w3.org/Mobile/CCPP/.

[9] Open Mobile Alliance, "WAP Forum Specifications", http://www.wapforum.org/what/technical.htm.

[10] World Wid Web Consortium, "Resource Description Framework", http://www.w3.org/RDF/.

[11] Packet Switch Streaming Activity Group, http://www.imtc.org/act_pss.htm.

[12] Vladimir Korolev and Anupam Joshi, "An End-End Approach to Wireless Web Access", *IEEE Distributed Computing System Workshop*, 2001 International Conference, April, 2001.

[13] Mark H. Bulter, "Using capability classes to classify and match CC/PP and UAProf profiles", http://www-uk.hpl.hp.com/people/marbut/capClass.htm, February, 2002.

[14] Erich Izdepski, "Ultimate Device Support with CC/PP", *XML-JOURNAL.com*, July, 2002.

[15] Hjelm, J., Nilsson, M., "Position-Dependent Services Using Metadata Profile Matching", *INET*, 2000.

[16] B. Krishnamurthy, C. E. Wills, "Improving Web Performance by Client Characterization Driven Server Adaptation", *WWW2002*, May, 2002.

[17] D. Chalmers, M. Sloman, N. Dulay, "Map Adaptation for Users of Mobile Systems", *WWW10*, May, 2001.

[18] J. R. Smith, R. Moha, C. Li, "Scalable Multimedia Delivery for Pervasive Computing", *ACM Multimedia 99*, October, 1999.

[19] Vaughn Iverson, Todd Schwartz, Young-Won Song, Rik Van de Walle, Ernesto Santos and Doim Chang, "MPEG-21 Digital Item Declaration FDIS", *ISO/IEC JTC1/SC29/WG11/N4813*, May, 2002.

[20] Anthony Vetro, Andrew Perkis and Christian Timmerer, "Study of ISO/IEC 21000-7 CD – Part 7: Digital Item Adaptation", *ISO/IEC JTC1/SC29/WG11/N5612*, March 2003, Pattaya, Thailand.

7. Acknowledgements

The authors would like to acknowledge the support of the Smart Internet Technology CRC.