

# Universal Profiling for Content Negotiation and Adaptation in Heterogeneous Environments

Tayeb Lemlouma and Nabil Layaida

OPERA Project, INRIA Rhône Alpes  
Zirst - 655 avenue de l'Europe - Montbonnot - 38334 Saint Ismier Cedex - France  
E-Mail: [Tayeb.Lemlouma@inrialpes.fr](mailto:Tayeb.Lemlouma@inrialpes.fr), [Nabil.Layaida@inrialpes.fr](mailto:Nabil.Layaida@inrialpes.fr)

This position paper focuses on the content negotiation and adaptation of services in heterogeneous environment and this using the concept of universal profiling. We discuss our vision to achieve an advanced content negotiation. For this, we define a profiling schema that includes the utile description of the entire component that can participate in the final deliverance of services. The defined profiling schema is opened, extensible and doesn't depend to a particular kind of devices, which make it useful in the context of adapted services deliverance.

## 1. Content adaptation and negotiation

Nowadays multimedia systems become more and more heterogeneous. Thanks to the explosive progress in computing technologies, modern systems include a wide diversity of client devices which are heterogeneous with respect to their capabilities and preferences. In such environments, a device can vary from a workstation, to a WAP phone, a PDA or any other limited terminal. The problem in these environments is to deliver a content understandable by the end user. Since clients don't share the same characteristics and use different formats of documents, services deliverance must depend to the user context and to the set of constraints that characterize the use of the wanted service.

One can think to put multiple versions of the same original document for different devices, and thus, to deliver the corresponding variant to the appropriate kind of user. This solution is heavy to do, because of the huge quantity of the original documents that can exist on the server side. Furthermore, the complete descriptions of a client (which we call the client profile) can vary dynamically in time and so it may be unknown at the authoring step. In addition to this, it's very difficult to take a census of all the kinds of users that exist in the environment and to prevent all the diversity of clients that may exist in the future.

To respond efficiently to the heterogeneity of the environment, multimedia systems must dispose of advantage mechanisms that allow adapting services according to users' interests and capabilities. These mechanisms must be capable to negotiate the best document variant if it exists but also to negotiate the best adaptation to apply on the original document in order to meet the client requirements. In this context, a good description of: the user (its preferences and the used device capabilities), the server (its content and adaptation capabilities) and the network plays a major role to make the final delivered service well adapted to the end user.

## 2. The adaptation of services

The adaptation of services represents the core of the architecture of an adaptable multimedia system. An adaptation method allows transforming a document from an initial state to a final state. An adaptation method can be a program (ex. a JAVA program that transforms BMP images to WBMP images), an XSLT [10] style sheet (ex. a style sheet that transforms XHTML documents to WML documents) or any other adapting procedure. The adaptation is applied when the final state is more understandable by the user according to its context. A document is understandable when it respects as well as possible the constraints of the client service's use. In the context of services negotiation, constraints describe what kinds of service are supported by the client and the characteristics of these services and the included media resources. For example, a constraint of the form: 'OnlySupportedService = wml' expresses that the user supports only WML documents. A constraint like 'MaxSize = 3000 Bytes' (which means that the size of a wml card must not be greater than 3000 bytes) can be a sub-constraint of a WML card description.

Since adaptation processors will be applied on original content, a particular attention must be paid during the authoring step. Content creation must be developed following device independent principles, which makes easy the adaptation of the content to a particular device context. To reach this goal, XML-based authoring represents a good solution thanks to XML [9] advantages such as the platform independence, the clear structuring, the well supporting, the modularity, etc.

### 3. Content negotiation and the universal profiling

An adaptation method is applied on an original content, at the end step of some messages exchange between the client device and the original server. Selecting the document to deliver or the adaptation method to apply depends on many parameters:

- The server content (versions, original documents)
- The content of related servers that represent a part of the same global multimedia system and may be exploited by the original server.
- The server capabilities in terms of transformation and adaptation methods.
- The client profile
- The network profile

In the step, called the content negotiation, which precedes the final document deliverance, the server uses a negotiation strategy that allows to do the best effort to match all the environment parameters constraint with the client profile. In this step the acquisition of the client profile is important, the profile can be sent by the client but also be stored initially on the server side. In our negotiation and adaptation architecture - that we will describe in section 5 - profiles and profiles changes are sent by clients, and this to make the environment opened to a large diversity of devices and also to support dynamic changes of profiles content.

The negotiation strategy that we have adopted in our architecture takes into account the limitation of devices capabilities, this why negotiation messages are exchanged between clients and devices only

- When this is needed: i.e. when the deliverance of a service needs the current client profile; and
- Transports only utile information: i.e. information necessary to mach the server content and capabilities with the user demand.

So to respond to the requirement of an advanced negotiation strategy, environment description must be available in a clear and complete manner. Furthermore, eventual profiles acquisition and exchanges must be reduced and concern only utile information which has a direct relation with the content negotiation need and services deliverance.

To reach that objective, we define a universal profiling schema for content negotiation in

heterogeneous environments. Schemas are designed on an RDF [4] basis and its vocabulary shares the same format as the one used in the CC/PP model [2]. Profiles definition is done in a manner that makes it extensible and independent from any particular kind of users or devices. For example, the *neg:DeviceName* element of the client profile can represent a WAP phone, a Pocket PC, or any other class of users, which is not the case of some schemas developed for particular users, such as UAProf [8] for only mobile phones. Our profile schemas are accessible from [6]; examples of profiles that follow these schemas can be found in [7].

## 4. Profiling schemas

The schema model that we define is designed to be universal, which means that every component (server, client, document, etc.) susceptible to play a role in the content negotiation is described in the form of a profile. Actually our model includes following kinds of profiles:

### A) Client profiles

Profiles concerning the client description are given as follows:

**A.1 Client Profile:** The client profile describes the general characteristics of the client, i.e. its hardware and software platform. The schema includes the following components:

- Hardware platform
- Software platform
- Browser user agent

The hardware platform component describes the physical characteristics of the device such as the device name, displaying capabilities etc. The information included in this component can have a direct relation with the final deliverance of services, for example if the device is a mobile phone, the server can use the phone number information to send data to the phone. The software platform component describes the software side of the device such as the operating system used, its version, etc.

The browser user agent component describes players or tools used by the client to access to the network services. The component can include for example the player name, the player version, the used accessing protocol etc. This component plays a major role in the content negotiation. It includes a collection of information about the use of the network services. This collection is organized in three categories: 1) only supported resources, 2) preferred resources, and 3) non supported resources. Categories 1 and 3 are represented in the form of *rdf:bag* element. Category 2 is represented by an *rdf:seq* element and includes resources which can be ordered using the value of the *neg:priorityLevel* element.

To minimize the size of the client profile and allow the use of the utile information only when this is needed, we use only a link to the location of the corresponding client resource profile. Server will request the client resource profile if it contain information necessary in the services deliverance.

**A.2 Client Resource Profile:** A client resource profile includes one component called 'content requirement' which gives a detailed description about the constraints of the client use of a resource. The component can also use 'OnlySupportedResources', 'NonSupportedResources' and 'PreferredSupportedResources' elements in order to specify more constraints about a particular resource.

### B) Server Profiles

The description of the server is given in the form of the following profiles:

**B.1 Document Instance Profile:** In this profile, the document instance that exists on the server side is considered as a resource and a detailed description about it is given. Here the resource is described independently to the other included resources (images, videos, etc.). The profile includes three components:

- Document instance description
- Multimedia content
- Adaptable resources description

The last component gives the list of existing variants and adaptation methods that can be used to adapt the document instance. Each adaptable resource (variant or method) can be associated with a value, *neg:ResourceAdaptabilityValue*, that indicates its adaptability level and which can be used in the content negotiation strategy applied by the server. The detailed description of an adaptable resource, i.e. the profile, is cited in the document instance profile as a link. Note that if many documents share the same instance profile, the instance profile can be stored once and in this case we speak about a document profile and not a document instance profile.

**B.2 Resource Profile:** This profile is similar to the client resource profile. It completes the description of the document instance profile, for a single or many documents. The profile includes two components:

- Media resource description
- Adaptable resources description

The last component plays exactly the same role as the adaptable resources description component of the document instance profile. The entity considered here is the resource used by the document instance and not the document itself.

### **B.3 Adaptation Method Profile:**

This profile includes two components:

- Adaptation resource description
- Adaptation method description

The last component describes the capabilities of the adaptation method in terms of input requirements and output description.

Our schema model is still extensible to add new elements and information in order to offer a complete description about the useful information that can be used in the content negotiation. Among the future steps of this work, will be to include the definition of the network profile and to define the manner of processing it in the negotiation strategy used by the server.

## **5 An implemented Negotiation and Adaptation Architecture**

We have implemented "NAC" a Negotiation and Adaptation Core for heterogeneous multimedia systems [3]. Actually we are extending the core to support the handling of different profiles that belong to the defined schema.

NAC includes different entities:

- 1) **A Negotiation module**: responsible to determine the best variant to deliver or the kind of the adaptation which must be applied on the original requested document.
- 2) **An adaptation module**: responsible to apply a well determined adaptation method and can be enriched at any moment by new methods. Among the adaptation methods that we use, we find XSLT style sheets (ex. XHTML to WML, XML to Latex), transformation programs (ex. text to speech) and direct adaptation methods (ex. SMS message senders).
- 3) **The UCM**: The User Context Module is responsible to interact with the server in order to communicate the client capabilities and preferences. Since clients in the heterogeneous environment have limited capabilities, the module is designed to achieve the minimal set of required operations. Actually we have developed a variant of the UCM module for Pocket PCs. A complete architecture must include such module in all the devices that exist in the environment.
- 4) **Players and users contexts listeners**: this component of the architecture interacts directly with client players and UCM modules to process their requests. Listeners can be executed on the server side or on an intermediate proxy.

The implemented architecture must be ameliorated to support a large set of constraints that concerns the use of services. An advanced vocabulary of negotiation- related constraints must be well defined in the profiling schema.

## References

1. Tayeb Lemlouma, Nabil Layaïda, A Framework for Media Resource Manipulation in an Adaptation and Negotiation Architecture, OPERA Project, INRIA Rhône Alpes, August 2001  
<http://www.inrialpes.fr/opera/people/Tayeb.Lemlouma/Papers/RSRS.pdf>
2. W3C, Composite Capabilities / Preferences Profiles, <http://www.w3.org/TR/CCPP-struct-vocab/>
3. Tayeb Lemlouma, Nabil Layaïda, NAC: A Basic Core for the Adaptation and Negotiation of Multimedia Services, OPERA Project, INRIA Rhône Alpes, September 2001  
<http://www.inrialpes.fr/opera/people/Tayeb.Lemlouma/Papers/ANegoP.pdf>
4. W3C, Resources Description Framework, <http://www.w3.org/RDF>
5. Tayeb Lemlouma, Nabil Layaïda, The Negotiation of Multimedia Content Services in Heterogeneous Environments, 8th International Conference on Multimedia Modeling. Amsterdam, 5-7 November 2001, <http://www.inrialpes.fr/opera/people/Tayeb.Lemlouma/Papers/MMM2001.pdf>
6. Tayeb Lemlouma, Nabil Layaïda, Universal Profiling Schemas  
<http://www.inrialpes.fr/opera/people/Tayeb.Lemlouma/NegotiationSchema/index.htm>
7. Tayeb Lemlouma, Nabil Layaïda, Universal Profiling Schema for Content Negotiation, OPERA Project, INRIA Rhône Alpes, January 2002  
<http://www.inrialpes.fr/opera/people/Tayeb.Lemlouma/Papers/UniversalProfilingSchema.pdf>
8. WAP Forum, WAG UAPProf version 20-Oct-2001,  
<http://www1.wapforum.org/tech/documents/WAP-248-UAPProf-20011020-a.pdf>
9. W3C, Extensible Markup Language (XML) 1.0, W3C Recommendation:  
<http://www.w3.org/TR/1998/REC-xml-19980210>, 10 February 1998.
10. W3C, XSL Transformations (XSLT), Version 1.0, <http://www.w3.org/TR/xslt> W3C Recommendation November 16th, 1999

