A Data Specification Model for Multimedia QoS Negotiation

Lea Skorin-Kapov
Ericsson Nikola Tesla d.d.
Krapinska 45
HR-10000 Zagreb, Croatia
+385 1 365 3965
lea.skorin-kapov@ericsson.com

Maja Matijasevic
University of Zagreb, FER
Unska 3
HR-10000 Zagreb, Croatia
+385 1 6129 757
maja.matijasevic@fer.hr

ABSTRACT

An important issue for multimedia quality of service (QoS) negotiation is the specification of the parameters/profiles that need to be coordinated in order to reach an agreement among involved entities regarding service configuration, resource allocation, and the dynamics of service adaptation. The contribution of this paper is a generic Data Specification Model, including specification of both input and output data for the QoS negotiation process. The applicability of the proposed model is illustrated for an example multimedia service scenario.

Keywords

Multimedia, QoS, Negotiation, Adaptation, Data Specification.

1. INTRODUCTION

The efficient delivery of multimedia services over QoS-enabled heterogeneous 3G/4G networks requires a common view and understanding of quality of service (QoS) among all parties in the service delivery chain. The overall purpose of multimedia QoS negotiation is to reach an agreement among involved entities regarding the service configuration, resource allocation, and the dynamics of service adaptation in response to changes in the service environment. Parameters/profiles to be considered when coordinating and negotiating end-to-end (E2E) QoS include:

- End-user related parameters (e.g., capabilities, preferences, subscription data, context);
- Service requirements (e.g., system resources, network resources) and adaptation capabilities;
- Network capabilities (e.g. specification of available QoS classes, available bandwidth);
- Operator policies and pricing/charging data.

While a number of approaches in standards and literature deal with specifying various data sets that are relevant for negotiating QoS [1], what is missing is a high level data model specifying all relevant data in a comprehensive way. The model should be generic in the sense that it is independent of a particular network and service scenario. In this work we aim to provide a model of the data impacting the multimedia QoS negotiation process.

The goal of the proposed Data Specification Model is to identify the parameters specified by various actors (e.g., end user, service/application provider, network operator) that impact the service-level QoS negotiation process, i.e., the parameters that serve as input and the parameters that comprise the output.

The paper is organized as follows. In Section 2 we present an overview of related work dealing with the specification of data relevant for multimedia QoS negotiation. Section 3 provides a high level view of the QoS negotiation process. In Section 4 we present the proposed Data Specification Model, including specification of both input and output data for the QoS negotiation process. A case study presenting applicability of the proposed Data Specification Model for an example service scenario is given in Section 5. Section 6 concludes the paper.

2. RELATED WORK

A number of approaches in standards and literature deal with specification of QoS parameters at the service level and specification of adaptation capabilities. The IETF has specified the Session Description Protocol (SDP) [2], commonly used together with the Session Initiation Protocol (SIP) for multimedia sessions. SDP allows participants to declare their receiving capabilities and preferences on media components and their characteristics. The shortcomings of SDP led to the proposal of XML-based SDPng (new generation), which at the time of this writing remains only as a proposal and has not advanced to RFC status. SDPng has been used as a basis for a data format in End-to-End Negotiation Protocol (E2ENP) [3] for negotiating and coordinating application layer QoS parameters and capabilities.

A key issue in negotiating and adapting service parameters is the specification of user requiremens as well as service requirements and adaptation capabilities. In the case of an adaptive service, this includes specification of alternative service operating parameters (e.g., different combinations of media components, different codecs, display sizes, frame rates, etc.) and adaptation rules indicating when (under what conditions) and how it would be best to adapt the service. A technique which has been often adopted to specify the relationship between achievable quality and constraints (e.g., network resource availability) is based on the

MOBIMEDIA 2007, August 27-29, Nafpaktos, Greece

DOI 10.4108/ICST.MOBIMEDIA2007.1689

specification of utility functions (UF) [4]. UFs reflect the adaptation capabilities of an application, relating the change in utility to a change in available resources. For example, while occasional frame loss or jitter may yield acceptable utility for a web camera application streaming a view of the city, such performance would have unacceptable utility for a high quality streaming movie. Hence, the shape of a UF, in general, is not just media-specific, but also application-specific. Based on UF specification and dynamic availability of network resources, a suitable application adaptation technique may be invoked. The notion of a Utility-Based Adaptation Framework is discussed in [5] with the authors identifying the major concepts involved in adaptation processes as the specification of a mapping between application adaptation space, resource constraints, and achieved utility. The specification of application adaptation capabilities has also been addressed by standards groups such as the Moving Picture Experts Group (MPEG), a working group of ISO. The MPEG-21 Digital Item Adaptation (DIA) standard [6] focuses on supporting meta-data driven adaptation, whereby Digital Items (a fundamental unit of distribution and transaction) contain information about adaptation capabilities. Such information may be used by end systems and/or network based content adaptation nodes for media adaptation on the fly, based on parameters such as current resource availability and user preferences.

Standardization efforts are also active in the area of specifying client capabilities and user-related data. Significant initiatives include the Composite Capabilities/Preferences Profile (CC/PP) by the WWW Consortium (http://www.w3.org/Mobile/CCPP/), and the User Agent Profile (UAProf) by the Open Mobile Alliance (http://www.openmobilealliance.org/tech/profiles). The Third Generation Partnership Project (3GPP) has specified a Generic User Profile (GUP) as a collection of user related data aimed to enable harmonized usage of user-related information located in different entities [7]. The goal is to provide an architecture, general data description model, and interfaces with mechanisms to handle the data.

3. MULTIMEDIA QOS NEGOTIATION

We describe the process of multimedia E2E QoS negotiation as comprising eight steps, or phases, as illustrated in Figure 1.

The QoS negotiation process is initiated by a service request. The first phase is the collection of relevant input data specified by involved actors (phase I). Following data collection, a matching process (phase II) uses input parameters to determine a set of feasible service parameters based on evaluation of the following conditions: (1) a user's terminal capabilities can support the minimal processing requirements for that service; (2) a common set of service operating parameters (e.g., codecs, resolution) can be found for each required media component (parameters must be in line with operator policies); (3) the user's access network can support the minimum requirements for all required media flows; and (4) the user's requirements (e.g., in terms of acceptable cost, media components and timing constraints) can be met. We name this parameter set Feasible Service Profile (FSP).

The matching process is followed by a negotiation process (phase III) in which the FSP is signaled to relevant parties in order to achieve an E2E agreement. An end user may accept, refuse, or modify the offered parameters. Network entities may further authorize resources to include limits on data rates and traffic classes for uplink/downlink flows, and also to apply QoS policy and admission control mechanisms.

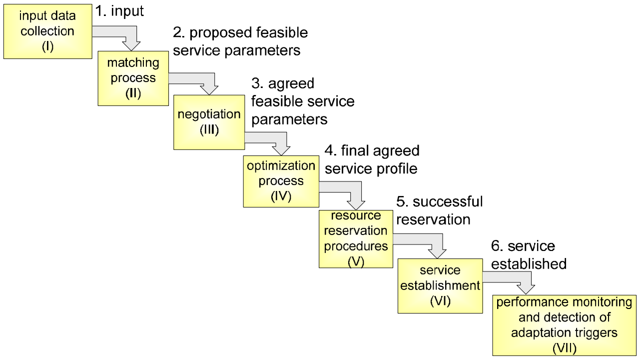

Figure 1. Phases in the QoS negotiation process

Based on negotiated and authorized parameters, the final FSP is determined. The FSP then serves as input to an optimization process (phase IV) which calculates the optimal service configuration and respective resource allocation across all media flows according to an established objective. The optimization objective may be formulated dynamically, e.g., based on user preferences such as: (1) maximum possible service quality; or (2) minimum cost while maintaining acceptable service quality; or (3) the best “value for money” service.

The optimal service configuration, and a number of alternative (suboptimal but feasible) service configurations are then ordered by descending utility and signaled to involved entities in the form of an Agreed Service Profile (ASP). The ASP contains a so-called media degradation path (MDP), defined as an ordered list of alternative service configurations and corresponding requirements (Figure 2). MDP also contains a “recipe” on when and how to degrade a service. The question of “when” is related to the notion of *adaptation-triggering events*, which represent a change in the input parameters or a violation of agreed parameters, hence leading to the need for service adaptation.

Having determined the ASP, the necessary resources are reserved (phase V), and, following the successful reservation, the service is established (phase VI). The QoS performance (according to agreement) is continuously monitored by network entities to

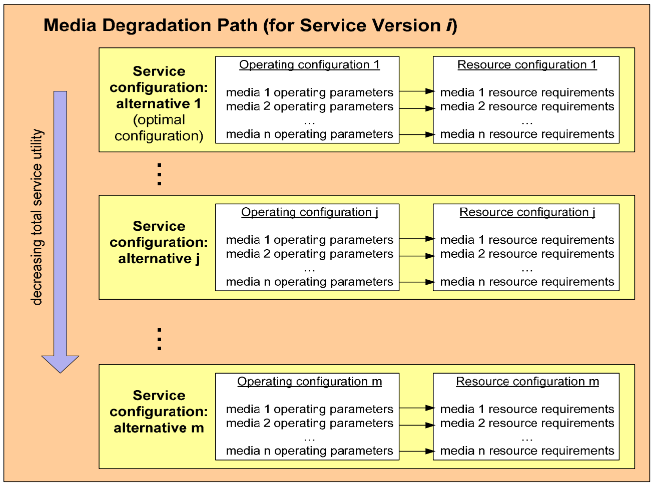

Figure 2. Media Degradation Path

detect adaptation-triggering events (phase VII). If adaptation becomes necessary, it occurs along the previously signaled MDP.

4. DATA SPECIFICATION MODEL

Figure 3 shows a generic model of the data that serves as input to the QoS negotiation process. It consists of associations, which encompass data logically related by a particular role in the QoS negotiation process. Examples of associations include "User Data", "Service Provider Data", "Service/Application Data", and "Network Provider Data". A particular association links together one or more components that represent logically related data within the association. For example, "user preferences", "access network capabilities", "terminal capabilities", and "subscription data" are considered components which are linked together to form the association "User Data". The roles taken on by actors and determined by the business model may impact the components within a certain association; e.g., if the service provider and network provider roles are taken on by two different actors, then a component corresponding to an SLA will exist between them.

A number of different types of components have been identified. The component in the lowest level contains one or more parameter sets, each composed of one or more parameter-value relations. For example, we can have an association "User data" which contains a component "user preferences", which in turn contains a parameter set "video preferences", which in turn contains the parameters "codec=MPEG", "frame rate=25", and "desired quality = HIGH". It is important to note that QoS parameters are specified at different abstraction levels (e.g., user perceived QoS, application level QoS, network QoS) and need to be mapped all the way down to resource requirements (e.g., bandwidth determined based on codec output rate and frame rate). Values may be of any type (discrete, continuous, string, Boolean, etc.). Each parameter has associated with it a Parameter Description to specify parameter semantics in the negotiation process. A component may also correspond to a policy specification, where the policy is composed of one or more rules. We consider rules as specifying actions to be performed in response to defined conditions. Conditions and actions are both specified in relation to certain parameters or parameter sets.

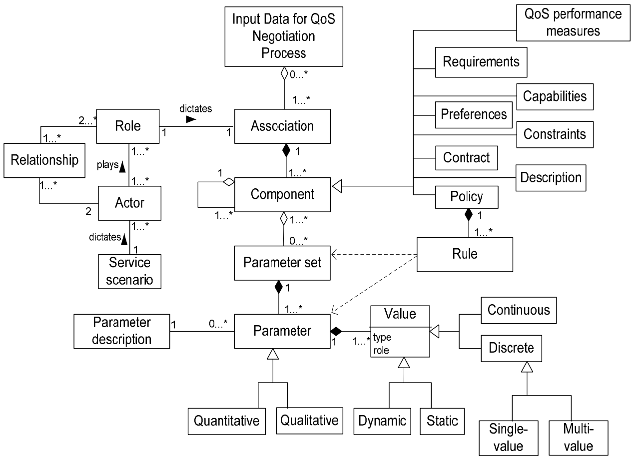

Figure 3. Input data specification for QoS negotiation

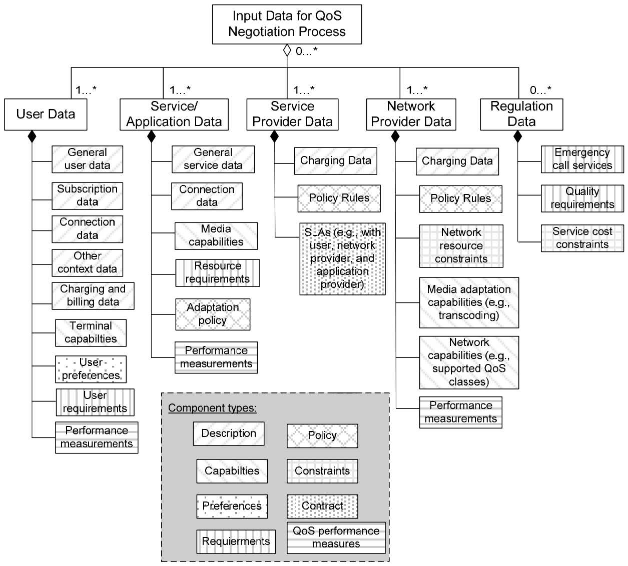

Figure 4. Example data associations and components

To demonstrate the model applicability, Figure 4 shows an input data example for a typical set of identified roles and corresponding associations. The following data associations are identified: (1) User Data: a collection of user-related data. In the case that multiple users are involved in QoS negotiation, multiple "User Data" associations will need to be considered; (2) Service/Application Data: data used to describe the characteristics and requirements of the service to be delivered to the end user(s). This data is specified by the actor that is providing the end user service/application; (3) Service Provider Data: data related to the policies, capabilities, and contracts of the Service Provider which are relevant for QoS negotiation. Multiple service provider domains may be involved along the E2E path; (4) Network Provider Data: data related to the policies, capabilities, and contracts of the Network Provider which are relevant for QoS negotiation. Multiple network provider domains may be involved along the E2E path; (5) Regulation Data: data specified by a regulative authority which is relevant for QoS negotiation and hence may impose certain requirements and/or constraints.

4.1 Modeling User Data

The user-related data relevant in negotiating and adapting QoS is modeled as a generic User Profile, shown in Figure 5. The User Profile represents a logical grouping of data into a number of components. Components of the proposed User Profile may be stored at distributed locations in the network (e.g., end-user terminal, network repository), with different entities having access to request, retrieve and modify profile information.

The "General user data" component provides general, non-service specific data about the end user, such as the user's name and postal address. The "Subscription data" component is a collection of data related to a user's subscription, including user ID(s), authorized services/media types, subscribed services/media types, data for authentication and authorization at registration, user priority level (e.g., high priority user, low priority user), and subscribed QoS profile (e.g., a "low bandwidth" subscription).

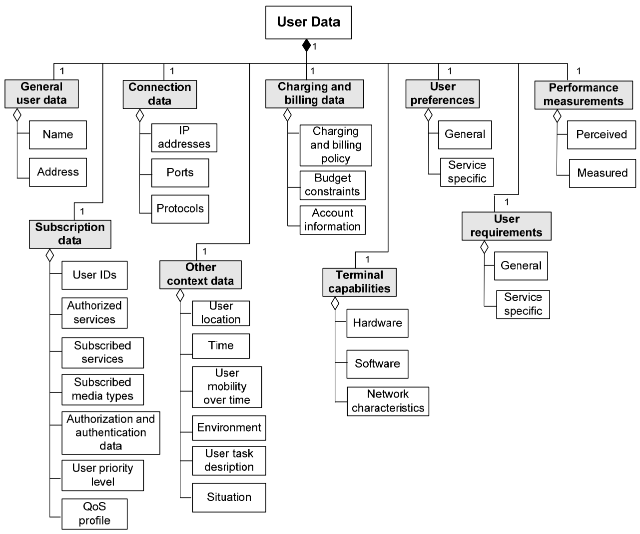

Figure 5. Generic User Profile

The "Connection data" component specifies the user's IP addresses (multiple addresses if multiple terminals are involved), ports for receiving data, and protocols that are to be used.

Context may be defined as any information that can be used to characterize the situation of an entity, including user requirements and preferences, location information, ambient conditions (spatial, physiological, and environmental information), access network and terminal capabilities, etc. For this reason, the complete user context may be derived based on all user data components. The component labeled "Other context data" specifies those context parameters that are not included in other components, such as user location, time, environment, and situation (e.g., in a meeting, at lunch), etc.

The "Charging and billing data" component specifies information related to charging and billing a user. With regards to QoS negotiation, it is necessary to consider a user's budget to determine what kind of service the user can afford. A user may also impose certain restrictions on the amount of money that he/she wishes to spend for a service, over a period of time, etc. Furthermore, different charging models may have been agreed to by the user (e.g., time-based charging, volume-based charging, service-based charging, event-based charging, flat rate, etc.).

The "Terminal capabilities" component specifies data related to the capabilities of a user's terminal(s). A user may have/use multiple terminals in which case sub-components and parameter sets will be specified for each of those terminals. The hardware description specifies characteristics such as display size, processor type, memory, user interface capabilities, etc. The software description specifies parameters related to the terminal's application environment, such as operating system, browser information, supported software, and supported media types. Media types include a list of MIME types and encoding/decoding capabilities, important for the customization of media formats that are delivered to the terminal. Parameters describing the terminal access network environment are a part of the "network characteristics". These parameters specify the bearer on which the session was opened, bearers that are supported by the device, and the downlink and uplink QoS communication characteristics.

The following two components specify user "Preferences" and "Requirements". An important consideration is the way in which a user expresses requirements and preferences, and the mechanism of collecting such data. In general, the rule is that from the point of view of the user, expressing requirements and preferences should be as simple as possible. Commonly, users will provide only perceptive media quality descriptions when specifying requirements and preferences, rather than technical descriptions. More advanced matching and "fine tuning" should remain transparent to the user. For example, rather than offering the user a number of options, a provider would offer only a high/low quality service version, and with regards to network QoS, the offered options would be "gold", "silver", or "bronze" service.

The last component included in the generic User Profile is "Performance measurements". Performance measurements may be provided by a user during or after a service lifetime to indicate the actual achieved QoS, and may be based on qualitative (e.g., user perceived quality) and quantitative metrics (e.g., measured loss).

4.2 Modeling Service Data

The proposed generic Service Profile is shown in Figure 6. The "General service data" component includes parameters specifying service name, designation, and description. The "Connection data" component specifies the IP address, ports for receiving data, and protocols to be used for the media flows. The "Media capabilities" specify the types of media components included as part of the service, including a list of MIME types and encoding/decoding capabilities, codec parameters, and formats.

4.2.1 Service versions

We assume that a service may come in one or more different versions to meet the heterogeneous capabilities of end users and access networks. Assuming that a multimedia service is comprised of one or more media components, we specify service versions as differing in the included media components (e.g., Version-1 with streaming media and Version-2 without streaming media). Consequently, versions will also differ in network requirements. A particular service version may further be configured in different ways (e.g., Version-2 with streaming media may choose between codecs C1 and C2). We assume the Service Profile to include a specification of all available service versions, their possible operating parameters, and corresponding requirements. The optimal service version and corresponding configuration of operating parameters will then be determined as a result of the QoS negotiation process. For each service version, the Service Profile specifies minimum network and processing requirements needed to support all (potentially) active media flows. These requirements are considered in determining if a particular service version is feasible. Furthermore, requirements related to associations of media flows (e.g., audio-video synchronization requirements) are also specified. The last component under "Service versions" specifies the resource requirements and adaptation capabilities per media flow.

4.2.2 Service adaptation

We base our method of specifying service requirements and adaptation capabilities on Utility Functions (UF), and the modified concept of specifying a mapping between adaptation utility, and resource spaces proposed in [5]. Our modification restricts the entire adaptation space to an Operating Space *O*, the

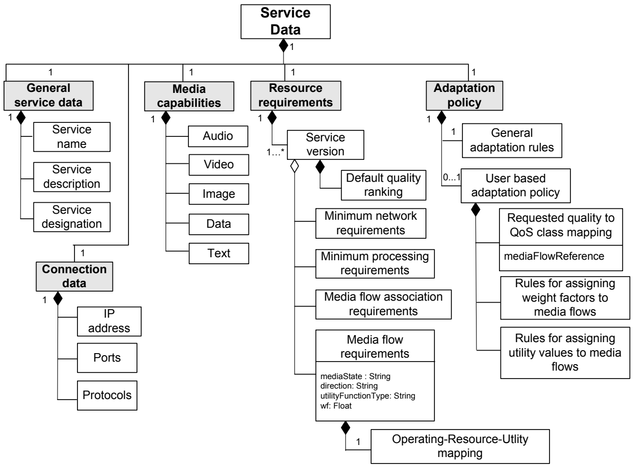

Figure 6. Generic Service Profile

dimensions of which constitute those application-level QoS parameters that are being agreed on E2E (e.g., codec, frame size, frame rate etc.) for the purpose of service configuration and resource allocation. The modified concept is referred to as Operating–Resource–Utility (*O–R–U*) mapping. Figure 7 illustrates the proposed mapping. We assume a mapping of feasible media operating points in *O* to a Resource Space *R* indicating the resources required to support that operating point. The resource space may consist of a number of constrained resources, such as network resources (e.g., network bandwidth) and system resources (e.g., processor cycles, memory). A point in *O* is also mapped to a Utility Space *U*. Since the choice of a particular operating parameter may allow an application to adapt to a range of resource availability values, a point in *O* may be mapped to multiple points in *U* and *R*. (For example, the same codec may operate at different output rate and thus require different bandwidth for each.) Within certain resource dimensions (e.g., bandwidth), media operating points (e.g., MPEG-4 codec chosen) may be mapped to continuous ranges or discrete resource values, resulting in continuous or discrete utility values. In the case of a continuous range, the Service Profile specifies a set of discrete points from that range and allows for the remaining points to be interpolated. Furthermore, multiple operating points may theoretically have the same resource requirements and different values in the utility space. For our focus regarding optimal resource distribution among multiple flows (discussed in more detail in the following section), we are interested in specification of the maximum utility achievable for a particular resource allocation. The maximum overall utility is specified as a function of multiple utility dimensions.

4.2.3 Media Flow requirements

In addition to the *O–R–U* mapping specified for each media flow, a number of other parameters are specified as part of the "Media flow requirements" component as shown in Figure 6. The "media state" parameter indicates whether a media flow is currently active or inactive. This parameter may be modified during the course of a service lifetime (e.g., a user starts/ends a video stream). The "media direction" parameter indicates whether a flow is transmitted uplink, downlink, or both. The "utility function type" indicates whether a utility function is continuous or discrete. The parameter "weight factor" (WF) is assigned to the user perceived utility for each media flow to indicate its relative importance with regards to other flows. This WF is taken into consideration when calculating optimal distribution of resources among media flows.

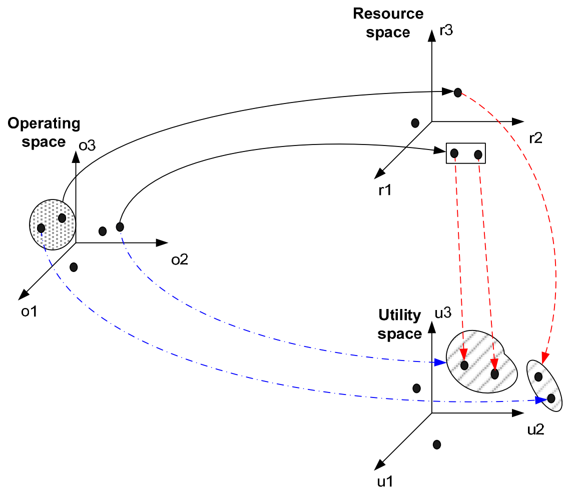

Figure 7. Operating–Resource–Utility mapping

While adaptation capabilities (in terms of different ways to configure the service) are specified in the *O–R–U* mapping, a Service Profile may specify general "Adaptation policy" referring to the conditions under which adaptation capabilities should be applied.

4.3 Specification of Agreed Service Profile

An Agreed Service Profile (ASP) represents the output of a successful QoS negotiation process. It specifies the optimal service configuration and resource allocation, as well as a degradation path for media flow degradation in case of a decrease in resources. The ASP structure is different than the original Service Profile in that certain components have been added, while other components, which are not needed to specify the final agreement, have been removed.

A generic ASP structure is shown in Figure 8. The "General service data" contains the same parameters as the original Service Profile. The "Connection data" component specifies those IP addresses, ports, and protocols that will be used for the agreed service configuration. The "Media capabilities" component specifies those media types that are included in the service configuration (e.g., encoding/decoding, codec parametrization, MIME type). These media specifications may then referenced by the service "Configuration and requirements" component. The "Configuration and requirements" component specifies the optimal service configuration, a number of alternative service configurations to be used in case of a decrease in available resources, and corresponding requirements.

It was previously stated that we assume a service to be offered in one or more versions differing in the included media components (e.g., Version-1 with streaming media and Version-2 without streaming media). We define a Version Degradation Path as being comprised of different (feasible) service versions ordered according to decreasing user perceived value. The highest quality service version that is feasible is labeled as the "chosen version". A service version may then further be configured in different

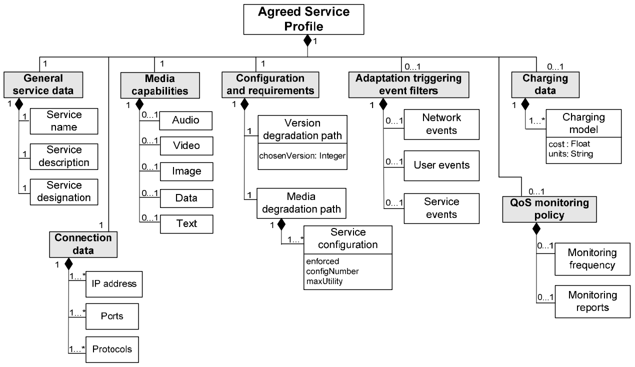

Figure 8. Agreed Service Profile

ways, where each service configuration specifies how to set up active media flows and what resources to assign to each flow.

For a chosen service version, the MDP is specified and included in the ASP. Switching to a different service version would result in the need to update the ASP with a new MDP for the newly chosen version. At a given point in the service lifetime, only one of the configurations is labeled as "enforced=YES" while others are labeled as "enforced=NO". In the case that none of the configurations in current MDP can be supported, the service is degraded to a lower quality version and a new MDP is specified.

It should be noted that by specifying the MDP we are considering the effect of degradation (or upgrading) simultaneously on all involved flows, rather than only for a single flow. For example, if an audio stream may be configured in three different ways (A1, A2, A3), and a video stream in two different ways (V1, V2), then we can specify a degradation path (in order of decreasing utility) as: {A1, V1}, {A1, V2}, {A2, V1}, {A2, V2}, {A3, V1}, {A3, V2}. Such an approach can allow for optimal degradation (or upgrading) in light of changing resource availability. Without the specified degradation path, it would not be clear in which order to degrade (upgrade) multiple media components so as to provide maximum achievable utility.

While the MDP specifies the logic of how a service should be adapted, there remains a question of "when" this is done. We define an adaptation-triggering event as a signal indicating that. An adaptation-triggering event notifies the service logic that a violation has occurred with regards to agreed network/system performance (e.g., a decrease in the network bandwidth allocation past a given threshold), or a change has occurred in the input parameters necessary for the QoS negotiation process (e.g., regarding access network/terminal capabilities, user preferences, costs, service requirements, etc.).

5. CASE STUDY

In this section we illustrate model applicability for an example service scenario. The service we developed is a Web based application called *Virtual Automobile Gallery* (VAG) (Figure 9), allowing a user to navigate through a 3D virtual gallery and view images of different automobiles. The gallery is created using the Virtual Reality Modeling Language (VRML) and viewed using the Cortona VRML plug-in. Throughout the world are stands that a user can click to view video clips, streamed across the network and displayed using a Java Media Framework based player.

We start by identifying the different data associations that will provide input for QoS negotiation. We assume a business model composed of roles as shown in the example in Figure 4, whereby the VAG is hosted by a third-party service provider (providing a service profile specifying service requirements), and offered to the user via an operator taking on the role of both a primary service provider and a network provider.

A user profile has been specified in accordance with the generic user profile (Figure 5). The "Terminal Capabilities" component specifies the hardware and software capabilities of two available terminals: a laptop PC and a handheld iPAQ. The available access networks are specified as UMTS and WLAN. Within the "User Preferences" component the user specifies that (s)he prefers high quality audio and video, and a service optimization objective that will maximize service quality. The "User Requirements" component specifies a maximum user budget that the user wishes to spend for the service, and a maximum allowed download time of 20 s for any requested file download.

The VAG service profile has been specified in accordance with the generic service profile (Figure 6). The application was developed in three different versions differing in display size and availability of media streaming (version 1 offers audio and video, version 2 only audio, and version 3 has no integrated streaming). The following media flows have been identified: virtual scene description (versions 1, 2, and 3), audio stream downlink (versions 1 and 2), and video stream downlink (version 1 only). For each stream, an operating-resource-utility mapping is used to specify requirements and adaptation capabilities (Table I). Resource requirements are defined as bandwidth requirements for fixed values of delay and loss corresponding to different network QoS classes (for audio and video: delay = 200 ms and loss = 2%; for scene description: delay = 500 ms and loss = 0%). Utility is defined on a scale of [0, 1]. Values are hypothetical and specified for test purposes only.

We assume the scenario where the user is using a PC and sends a service invitation indicating that the UMTS access network connection is currently active. Following successful negotiation and optimization of service parameters taking into account all user, service, and network constraints, an ASP is determined.

Assuming that ASP contains all three versions, since all of them are feasible, a version degradation path is specified as version1→version2→version3, with version 1 being the highest quality and "chosen" version. At service establishment, the only active media flow is the scene description download.

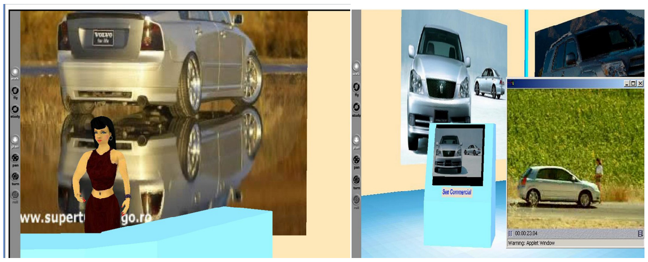

Figure 9. Virtual Automobile Gallery

Table 1. Operating points for the VAG application

Media flow	Operating parameter	Resources req. [kB]	Utility
Audio	codec: GSM	3.65	0.25
	codec: PCM	9.77	0.70
	codec: MPEG	12.25	1.00
Video	codec: H263	41.00	0.70
	codec: MPEG	56.00	1.00
Scene Description	fileSize: 300 kB	10.00	0.10
	fileSize: 300 kB	200.00	1.00

Consequently, the MDP included in the ASP contains only one media flow. Following this download, a user may click to start a video clip. The MDP is updated to include audio and video. The following is an example of MDP specification containing two possible service configurations:

```
<mdp:MediaDegradationPath>
   <mdp:config number="1" util="2" enforced="yes">
            <mdp:media ref="audio" direction="downlink">
                        <mdp:operating codec="mpeg "/>
                        <mdp:network minB="12.25" maxB="12.25" />
            </mdp:media>
            <mdp:media ref="video" direction="downlink">
                        <mdp:operating codec="mpeg"/>
                        <mdp:network minB="56" maxB="56" />
            </mdp:media>
   </mdp:config>
   <mdp:config number="2" util="1.7" enforced="no">
            <mdp:media ref="audio" direction="downlink">
                        <mdp:operating codec="pcm"/>
                        <mdp:network minB="9.8" maxB="9.8" />
            </mdp:media>
            <mdp:media ref="video" direction="downlink">
                        <mdp:operating codec="mpeg"/>
                        <mdp:network minB="56" maxB="56" />
            </mdp:media>
   </mdp:config>
</mdp:MediaDegradationPath>
```

Configuration 1 is the optimal configuration and is labeled as "enforced". Should a decrease occur in available network resources, the network will attempt to reserve resources according to Configuration 2, with necessary signaling messages sent to relevant entities along the service path to indicate degradation. Should the user decide to switch from PC to iPAQ, an update will be signaled and the MDP may again be updated to reflect feasible service configuration alternatives taking into account the new terminal restrictions (e.g., the iPAQ may not support video).

6. CONCLUSION

The generic Data Specification Model for multimedia QoS negotiation proposed in this paper may be considered as a reference model to be used by multiple actors involved in the QoS negotiation process for the purpose of providing a common understanding of specified parameters and corresponding semantics. Application providers are offered flexibility in specifying the requirements and adaptation capabilities of offered services; users can specify parameters such as preferences in order to customize service delivery; and service/network operators can specify their own policies and resource allocation schemes that are to be applied.

In order for proposed profiles to comply with existing standards regarding session signaling and description formats (namely the IETF SIP and SDP protocols, adopted also by 3GPP and ETSI/TISPAN standards bodies as protocols for session establishment, modification, and termination), enhancements to standards are needed allowing for more complex and extensible descriptions to be included. While SDPng is a move in this direction, SDPng is at the time of this writing only a proposal and has not yet been accepted as a standard. The key enhancements proposed in this work include: (1) the specification of more complex user parameters, such as user preferences/requirements and context allowing for advanced service customization; (2) the specification of an operating-resource-utility (O-R-U) mapping describing service requirements and adaptation capabilities; (3) the specification of an MDP to aid in service renegotiation and adaptation; and (4) the specification of adaptation triggering event filters to indicate to involved entities when adaptation-triggering events should be sent.

We have demonstrated model applicability using an example multimedia service scenario. Our current work is focused on further applying the proposed model in the scope of a broader QoS negotiation framework encompassing parameter matching, service optimization, and signaling for dynamic QoS adaptation.

7. ACKNOWLEDGMENTS

The authors acknowledge the support of Ericsson Nikola Tesla research project SESAME, and research projects 036-0362027-1639 and 071-362027-2329 funded by the Ministry of Science, Education and Sports of the Republic of Croatia.

8. REFERENCES

[1] Jin, J., and Nahrstedt, K. QoS Specification Languages for Distributed Multimedia Applications: a Survey and Taxonomy. *IEEE Multimedia*, July-Sept. 2004, 74-87.

[2] Handley, M., and Jacobson, V. SDP: Session Description Protocol. IETF RFC 2327, April 1998.

[3] Guenkova-Luy, T., Kassler, A. J., and Mandato, D. End-to-End Quality-of-Service Coordination for Mobile Multimedia Applications. *IEEE Journal on Selected Areas in Communications*, *22, 5* (June 2004), 889-903.

[4] Wang, X., and Schulzrinne, H. An Integrated Resource Negotiation, Pricing, and QoS adaptation framework for multimedia applications. *IEEE Journal on Selected Areas in Communications, 18, 12* (Dec. 2000), 2514-2529.

[5] Chang, S.-F. Optimal Video Adaptation and Skimming using a Utility-Based Framework. In Proc. of the *Tyrrhenian International Workshop on Digital Communications*, Capri Islands, Italy, Sept. 2002.

[6] –, Information Technology – Multimedia framework (MPEG-21) – Part 7: Digital Item Adaptation, ISO/IEC 21000-7, 2004.

[7] 3GPP TS 23.240: 3GPP Generic User Profile (GUP); Architecture (Stage 2). Release 6, 2005-03.