An OWL-S based architecture for self-optimizing multimedia over IP services

David López[1], Jorge E. López de Vergara[2], Francisco González[3], and Alfonso Sánchez-Macián[3]

[1] Universidad de León, Campus de Vegazana, 24071 León, Spain,
`david.lopez@unileon.es`,
[2] Universidad Autónoma de Madrid, Escuela Politécnica Superior,
Ciudad Universitaria de Cantoblanco, 28049 Madrid, Spain
`jorge.lopez_vergara@uam.es`,
[3] Universidad Politécnica de Madrid, DIT,
Paraninfo de la Ciudad Universitaria 28040 Madrid, Spain
`vidal,aasmp`@dit.upm.es

Abstract. This article presents an architecture to manage a multimedia over IP delivery service. It considers user's expectations as the main metric that defines the quality of the service thus assuring an efficient resource and effort investment by network operators. The proposed architecture integrates into a single platform both required elements as well as involved processes in a multimedia over IP delivery scenario. In order to achieve a scalable solution, a distributed proposal is made according to proven modeling techniques, with maximum decoupling and composition. Web Services have turned out to be a valid approach to specify involved components. It is necessary, however, to extend the capabilities that web services provide with alternative technologies such as OWL-S to integrate both components and processes into a single unified specification of the management architecture.

1 Introduction

The ever growing amount of consumers embracing the so called new technologies is modifying the way we live, consume and spend our leisure time. Nowadays many digital devices like personal computers, MP3 players or mobile phones enable us the access to any multimedia content be it by the Internet, satellite, mobile networks or yet unforeseen access networks. Incumbent network operators, usually focused on the provision of traditional voice as well as data services, are compelled to modify their business strategies in order to adapt themselves to new consumer patterns. As a result of the progressive reduction in the fixed voice services revenue, network operators are investing a large amount of resources in high capacity networks as well as adopting state of the art technology to cope with the last mile access bottleneck. Being able to deliver high quality, bandwidth intensive, digital content to users will provide telcos new ways to increase their average revenue per user. Not only new services like Video on

Demand, Voice over IP, Video conference and gaming could be provided by the same medium access but at the same time new business models based on content personalization will be possible. Contrary to existing multimedia delivery platforms like broadband, cable or satellite, IP networks constitute a competitive advantage for network operators based on interactive user-tailored services.

Coping with this new paradigm of dynamic, user oriented services is a challenging issue for network operators which must adapt its service management procedures in order to remain efficient while handling a growing complexity. This situation calls for the development of new architectures capable of automating operational procedures in a consistent and unified manner, consistent in the sense that they provide up-to-date information regarding the status of the service and unified as it integrates disparate sources of information into a single entity. A promising solution to this scenario, as identified in [1], is the so called autonomic computing in which software entities show self-configuring, self-healing self-optimizing, self-protecting behavior.

In this paper a proposal to manage quality of multimedia content delivered over IP is given. The proposed management architecture, defined according to formal procedures, is able to control relevant resources in order to ensure an optimum quality response. In the first place, related work to adaptive concepts and management architectures is presented. Later, involved elements and processes are identified. In the third place an integration into a single, formally specified, entity by means of semantic annotation techniques is provided. This approach enables a flexible integration of disparate technologies to provide a framework capable of evolving according to the specifics of the service. Finally some performance results and conclusions are presented.

2 Related work

The management of telecommunication services and its related platforms is a research area still under further development in which several initiatives are currently working, for instance [2], [3], [4]. In a similar way to other domains, like distributed computing, a platform to manage multimedia service delivery must face not only the heterogeneity of involved resources -several technologies, many devices- but at the same time the evolving nature of new on demand services that enable users to personalize the service in many ways [3], [5]. The premises to ensure a scalable management platform shall be similar to [6], that is:

1. Maximum decoupling among components.
2. Maximum abstraction of the inner complexities of every component by means of well defined interfaces.
3. Existence of mediation mechanisms between components.

Whereas the first and second premises contribute to overcome the inherent complexity of a management platform, only the third one allows for adaptive solutions capable of dealing with the heterogeneity and dynamic conditions in which the services take place. Several research studies, as in [7], have shown that the concept of adaptability is closely related to the capacity of providing new functionalities based on the composition of individual elements.

Service adaptability is, as a matter of fact, a premise for self-optimizing architectures. If a service is to be self-optimizing then it has to be able to adapt to the dynamic environment in which it exists. For instance a self-optimizing multimedia over IP delivery service would ensure certain quality level no matter the network conditions and user concurrency might be. To do so the service would need to adapt the coding scheme in order to provide a graceful service degradation. Examples of how this adaptation is accomplished can be found in [8] and [9].

2.1 Adaptive Service Architecture

In [10] a proposal defining those requirements a service needs to fulfill in order to exhibit adaptive behavior is given. To do so it defines the concept of Adaptive Service Element (ASE) as a service abstraction which, relying on available context information, controls the state of involved resources.

An ASE is characterized by a service description, a model of the state, a description of the services it uses as well as a rule-based model for describing and restricting its behavior.

Several research efforts [11], [12] are supporting the validity of ASE as a conceptual reference to develop novel service management architectures that rely on the enhanced description logic that ontologies provide in order to formally specify both the description and the state model.

With regards to the third element defined in ASE, that is the rule-based model to determine service behavior, there are several alternatives to specify the goals to be accomplished: event-condition-actions, policies or utility functions.

2.2 Utility based service behavior

Quality of experience (QoE) has been defined as an extension of the traditional quality of service (QoS) in the sense that QoE provides information regarding the delivered services from an end-user point of view [13]. As a matter of fact the idea of QoE is closely related to the traditional concept of utility functions as high-level forms of requirement specification. Both QoE and utility functions allow to set degrees of desirability for some given levels of delivered QoS, this flexibility has turned out to be of practical use in self-optimizing management architectures [14].

The advantage of constraining the admissible behavior of any service to those that optimize utility, that is maximize QoE, is clear: by doing so the service behaves in a way consistent to high level business policies, that is user satisfaction.

Within the multimedia scope of this paper, QoE is defined as the cause-effect relationship among network metrics and quality aspects like video artifacts, audio distortions or whichever parameter that might affect users perception, that is for some set of quality characteristics $k = 1, \ldots, n$:

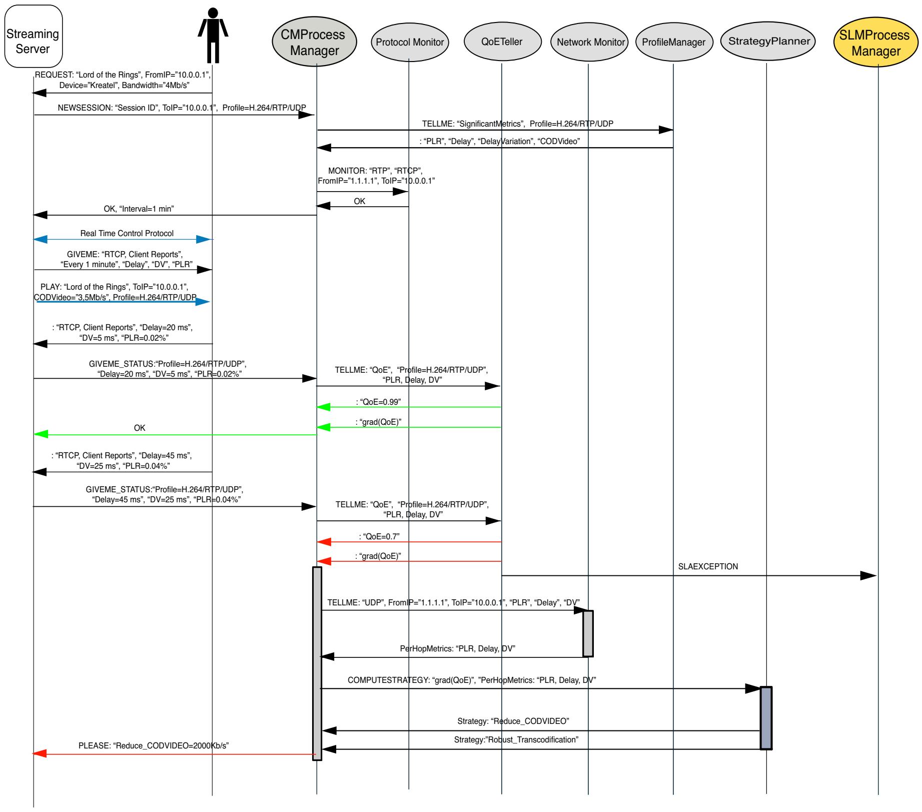

Fig. 1. Sequence diagram for a multimedia service capacity management

$$QoE_k = f(metric_1, ..., metric_n) \qquad (1)$$

For instance in [16] or [17] procedures to establish these type of relationships are described.

Based on the no-reference moving picture quality metric defined in [15], a multimedia adaptation scheme is proposed in [18]. The quality oriented adaptation scheme (QOAS) relies on a monitoring agent located at the client premises that periodically informs the server regarding the status of the delivered content. To do so the client computes an application-level quality score (QoDScores) based on network metrics such as loss rate, delay and jitter. According to the computed metric QoDScores the server decides, on a real time basis, whether to downgrade the delivered bit rate in order to adapt to existing network congestion. According to the results presented this quality oriented approach does provide a scalable solution from a management point of view, the feedback received by the server enables a self-regulating architecture.

In the following, elements and processes involved in the management of a multimedia delivery service are identified. By incorporating utility based restrictions into the framework a user oriented behavior is implicitly fixed in the resulting model. Then a state model precisely describing the inner workings of the architecture is introduced. Contrary to other approaches, this state model relies on proven modeling frameworks that facilitate its integration in existing operational procedures. Finally, all the elements are described by means of semantic annotation techniques in order to provide a complete and formal description of the architecture.

3 Managing QoE in multimedia over IP services

3.1 Involved processes in Multimedia over IP delivery service

In order to precisely characterize the processes involved in the delivery of multimedia over IP networks as well as the functional elements that conform a management platform, the sequence diagram depicted in figure 1 reflects a scenario of a user accessing a multimedia content. Firstly the negotiation and establishment process of the required sessions to transmit and control the media takes place. Whereas real time protocol (RTP) is mainly responsible for the transmission of multimedia content, real time control protocol (RTCP) deals with session control and negotiation to adapt the multimedia session to available resources [22].

This approach however, fails to ensure a graceful service degradation in case severe packet loss or delay took place. According to several results, as presented in [25] or [17] for instance, the resilience of a streaming service to network conditions is quite related to the codification scheme or bitrate delivery. As a general rule the more bitrate the more sensitive is the stream to network disruptions. This arguably calls for novel approaches to provide extra information regarding the conditions to ensure a proper service delivery, that is, the management platform needs to be aware of the particular context of a given streaming service in order to adjust to the restrictions imposed by the network.

This premise is reinforced by the fact, as shown in [23] and [24], that modeling human perception entails several factors to be taken into account. We humans integrate audio and video response when making a final opinion regarding the quality of the content. Besides, the content type does play an important role in the outcome of users opinion, while "talking heads" type content requires a special emphasis on audio quality and legibility of displayed news information, "high motion" content type such as sports requires special care of image resolution. In short, QoE predictive models do depend on the content delivered to consumers.

Furthermore gaining extra information, denoted as meta information, regarding the context of a streaming session will extend the possibilities to overcome network degradation by means of recent coding schemes such as H.264/AVC which provide new error-resilience features such as Flexible Macroblock Ordering (FMO) or Data Partitioning [25], [26], [27], [28].

Our proposal defines a new element, CMProcessManager, responsible for the monitoring and control of established sessions. Under this paradigm, for a given multimedia session the CMProcessManager, thanks to the information provided (such as available computing resources, access bandwidth, media coding, content type) is able to take pertinent actions to supervise opened sessions. The necessary actions, refer to figure 1, are:

1. Find the corresponding multimedia profile to a certain session request, task delegated to the element "ProfileManager".
2. Monitor established network protocols, task delegated to "ProtocolMonitor".
3. Monitor the delivered quality, expressed by QoE metric, task delegated to "QoETeller".
4. Monitor those network metrics described by the multimedia profile, task delegated to "NetworkMonitor".
5. Admission control and supervision of acceptable service level degradation, task delegated to "StrategyPlanner".

This actions partially corresponds to the Capacity Management process duties as described in the set of ITIL recommendations [19], [20] that is: resource supervision, SLAs fulfillment and demand management.

The proposed elements, thus collaborate together to ensure the best possible experience given some network conditions. In case any deviation in the committed QoE took place, the CMProcessManager would act accordingly to recover the committed level. To effectively restore the service, some network information -provided by "NetworkMonitor"- is needed in order to discriminate which actions will produce the desired outcome. For instance, in case a surge in demand produced an excessive packet loss it would be sufficient to reduce the coding bit rate whereas if the errors were due to the specific conditions of the access network, for instance wireless, then only more robust encoding schemes would work out.

3.2 Formal description of processes as Web Services

Several initiatives [4], [29] are at present time proposing a new software development approach based on web services as they promote distributed architectures with highly decoupled components and strict procedures to access offered functionalities.

For every required functionality an atomic process will be implemented. An atomic process, in the context of this article, represents a set of actions that a web service can handle on its own. In most cases an action consists of an incoming message, some calculations and related outgoing messages. Figure 2 shows the atomic processes required to perform every task as well as the incoming and outgoing parameters for every process.

Atomic Processes

Regarding the implementation of every atomic process by means of web services, the element "ProfileManager" provides the interface to fetch, from a

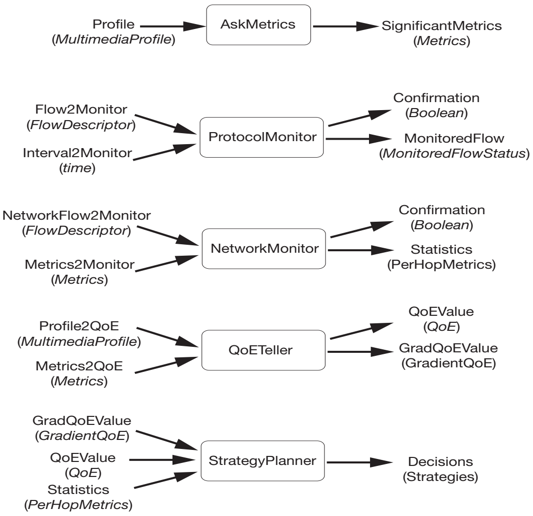

Fig. 2. Atomic processes corresponding to the Capacity Management Process

database, relevant data to compute QoE based on the expression defined in (1). The element "ProtocolMonitor" allows to supervise the status of every connection established by implementing an interface to access the real time transport MIB defined in [30]. Likewise the element "NetworkMonitor" aggregates information regarding layer 3 traffic flows obtained at each node, this information is obtained with flow-based accounting technologies such as netflow [31]. Contrary to other packet-based accounting technologies, netflow constitutes an efficient method to audit IP networks. Finally "QoETeller" is a web service which allows to compute the expression defined in (1) to do so it relies on software such as Mathematica [32]

Some extra parameters or entities specific to the domain in which the web services will run have been included. In this case some specific classes related to the network management field as:

1. *MultimediaProfile*. It details which information is relevant such as involved network metrics and required protocols. It has to be pointed out that each coding scheme will establish different protocol sessions and imposes different network restrictions.
2. *FlowDescriptor*. Characterizes the multimedia flow in terms of involved protocols, IP addresses and opened application ports.
3. *Metrics*. Represents those relevant metrics for a given Multimedia Profile.
4. *PerHopMetrics*. Represents the network metric values on a hop by hop basis, thus providing fine grained information regarding the network status.
5. *QoE*. Gives the computed QoE value.
6. *GradientQoE*. Represents the value of the gradient vector of the QoE function.

7. *Strategies*. Represents those feasible measures to overcome the multimedia service delivery degradation such as video rate downgrading or version switching.

Being able to model components as web services has turned out a useful approach when it comes to the integration of the elements defined in section 3.1. For instance a critical capability of the proposed model is the ability to compute, on a real time basis, the QoE metric according to available network information. Under the web services approach it is possible to hide the complexity involved in the calculus of the QoE metric by means of simple interfaces expressed in Web Services Description Language (WDSL). Then, for every involved element, there exists a WSDL interface clearly defining the access procedure in terms of expected input parameters, output ones or available resources [33]. This component-decoupled paradigm fosters not only reusability but at the same time future enhancements in case any extra functionalities were needed, for instance to manage added services such as Voice over IP.

Expressing the access to required elements as WSDL interfaces sets the foundation for a scalable QoE management platform, yet WSDL does not specify the correct sequence in which the required actions need to be executed in order to sustain the QoE level based on the status of the multimedia session. A process model is needed to completely define the dynamics of the management platform. Building such a process model outbound the capabilities assigned to WSDL, as it will be detailed in next sections it is necessary to go for alternative technologies, such as Service ontologies, to further enhance the web services based approach.

So far the architecture has been partially described in terms of the processes involved to manage the service as well as a formal representation, based on WSDL, for each defined process. However it is still necessary to specify the way atomic processes interact in order to achieve the objectives for a management platform. This specification will be achieved in following sections by means of a model of the process.

4 State model definition in Multimedia QoE Management

The development of ontologies to help modeling and developing software components has arisen great interest in related scientific forums, for those interested in the concept of ontology and its potential applications please refer to [34], [35], [36]. As far as web services is concerned, ontologies enables the enrichment of the information contained in a web service for another components to decide about the suitability of this web service to the component's purposes. Both web services and the related ontologies constitute the so called semantic web services, which are capable of exposing the functionalities offered by certain web service in a way not too obscure to avoid composition and neither too detailed to constrain reusability. This is commonly known as a "grey box" model in contrast to a "black model" in which everything but the essentials remain hidden. The idea of defining a web service and its related metainformation is a critical aspect to attain richer services based on the composition of atomic ones. As mentioned in section 3 the concept of composability is paramount to deploy scalable architectures able to adapt to future requirements.

OWL-S is a web service ontology based on the Web Ontology Language (OWL) that enables the description of web services in terms of their properties and capabilities in a way that might be understandable for both humans and software agents. In the end OWL-S contributes to automate the processes of discovery, invocation, composition and interoperation of web services.

Three are the main reasons that make OWL-S a suitable technology for the purposes of specifying involved components to manage the delivered QoE:

1. Conceptual framework closely related to proven modeling techniques such as activity diagrams, thus facilitating its usability.
2. Seamless composition of new components in order to enhance the capabilities according to previously unforeseen requirements.
3. Current development of related software to prototype and implement OWL-S based web services.

OWL-S defines a service as the combination of the following elements: *a process model*, *a related service profile. the service definition* and finally its *realization* in a concrete technology.

A model of the process, resembling a model of the state, describes the stages a process traverses according to its internal information and external inputs.

This model includes information regarding input variables, output ones, required preconditions for the service to be properly executed as well as potential effects on the environment (admission control, media transcoding and so on).

4.1 Semantic annotation of involved processes in QoE management

QoEManagementProcess is the process in charge of fulfilling the tasks corresponding to the capacity management process in multimedia sessions as defined in section 3.1. To do so it must provide the following functionalities:

1. Identify the appropriate multimedia profiles.
2. Monitor involved network protocols in the transmission and control of the multimedia delivery.
3. Evaluate the QoE metric corresponding to the established multimedia session.
4. Monitor relevant network metrics as specified in the multimedia profile.
5. Provide feasible strategies to provision a controlled service degradation to end customers in case it were necessary.

In figure 3 the behavior of the composite process “QoEManagementProcess” is shown. For a given multimedia session, and its corresponding multimedia profile, the atomic process “AskMetrics” provides the set of network metrics to be observed in order to estimate the QoE metric. Once these metrics are known, two concurrent actions are triggered: “QoETeller” y “ProtocolMonitor” responsible for the QoE estimation and protocol monitoring respectively.

The “ProtocolMonitor”process supervises the status of the established sessions, it is then a persistent process in the sense that once invoked it periodically informs about the status.

As far as “QoETeller” is concerned whenever the QoE metric remains at an acceptable level, this level being defined according to higher business rules, no extra actions will be required, but in case the level dropped below the minimum further invocations of another resources would be triggered in order to correct the deviation of committed quality levels. In this case “NetworkMonitor” and “StrategyPlanner” will be sequentially invoked.

Being the delivery of multimedia content over IP a complex service no straightforward recovery strategy could be suggested in advance. The right approach to handle a given service degradation is strongly related to network conditions and the type of transmission media thus prior to any decision some information regarding the networks across which the media traverses is needed. This constitutes the main role of the “NetworkMonitor” process: to provide finer details of network metrics, initially evaluated on an end to end basis, in order to locate the source of the problem. As shown in figure 1 the user periodically sends, in RTCP messages, measured values of end to end network metrics for QoE to be inferred, only in case the level dropped below the minimum the process “NetworkMonitor” would be invoked. This on demand approach spares network resources as it only requires hop-by-hop metric values when needed.

The process called “StrategyPlanner” is in charge of evaluating the available data to produce some actions to restore the QoE to its optimal level. The type of actions to be adopted are highly related to the root causes of the problem. For instance in case high packet errors rate were reported due to radio channel degradation in wireless environments then the best strategy would be to transcode into more robust schemes such as FMO or random intra macroblock refreshes, whereas if the problem were due to network congestion it would be required to downgrade the video coding rate or even perform admission control (deny access to bronze users for instance). This sort of tasks can be put into practice by some expert system, thus providing a self regulating scheme capable of adapt the service quality according to the restrictions imposed by the network.

4.2 Service Profile and Service Description

The Service Profile provides a generic description of the capabilities that the composite service is able to provide, whereas the realization of the service (denoted Grounding) specifies how to invoke and execute the aforementioned service. OWL-S allows several technologies for the service grounding, being WSDL the one selected in our approach mainly due to its wide adoption by the industry.

The code shown below realizes the atomic process QoETeller in RDF syntax which can be parsed and executed by several OWL-S interpreters such as [38]. Anyhow nothing precludes the possibility of service composition by means of

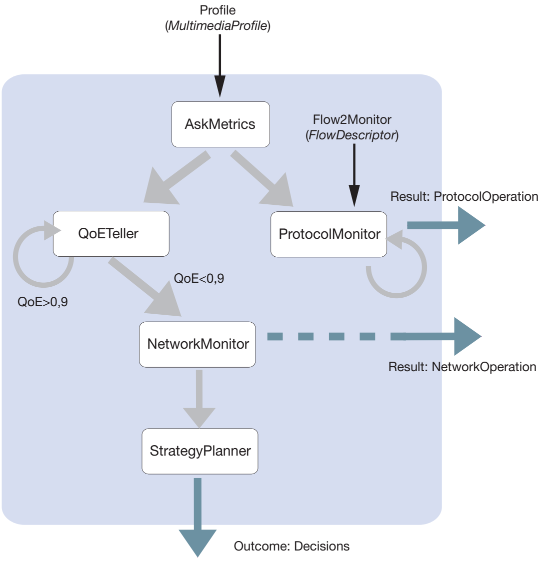

Fig. 3. OWL-S composite process -QoEManagementProcess-

alternative technologies, for instance Web Services Business Process Execution Language (WSBPEL) [42].

```
<?xml version="1.0"?>
<rdf:RDF
....
<grounding:owlsProcess rdf:resource="#QoETeller">
    <grounding:wsdlInput>
      <grounding:WsdlInputMessageMap rdf:ID="WsdlInputMessageMap_2">
        <grounding:owlsParameter rdf:resource="#ProfileToQoE"/>
        <grounding:wsdlMessagePart rdf:datatype="http://www.telefonica
        .net/2005/XMLSchema#MultimediaProfile">
        http://www.telefonica.net/wsdl/QoETeller.wsdl
        #ProfileToQoE
        </grounding:wsdlMessagePart>
      </grounding:WsdlInputMessageMap>
    </grounding:wsdlInput>

    <grounding:wsdlOutput>
      <grounding:WsdlOutputMessageMap rdf:ID=
      "WsdlOutputMessageMap_12">
        <grounding:owlsParameter rdf:resource="#QoEValue"/>
        <grounding:wsdlMessagePart rdf:datatype="http:
        //www.telefonica.net/2005/XMLSchema#QoE">
        http://www.telefonica.net/wsdl/QoETeller.wsdl
        #QoEValue
        </grounding:wsdlMessagePart>
        </grounding:WsdlOutputMessageMap>
    </grounding:wsdlOutput>
   </grounding:owlsProcess rdf:resource="#QoETeller">
...
</rdf:RDF>
```

In practice the atomic process "QoETeller" is implemented as as real web service described by a standard WSDL interface "QoETeller.wsdl". This web service accepts two MessageParts as inputs, "ProfileToQoE" and "MetricsToQoE" and two MessageParts as outputs, "QoEValue" and "GradQoEValue".

The WSDL interface "QoETeller.wsdl" provides a facade to a computing service [37] capable of evaluating, given certain specific profile and network metrics, the quality of experience that is to be expected for a streaming session. The profile represents the expression (1) for a specific type of streaming in terms of: network protocol -UDP versus TCP-, coding scheme -MPEG2 versus H.264-, content type - movie versus news- and so on.

This approach allows the extension of extra profiles, according to new user requirements or technologies to deliver media over IP, in a transparent manner to other elements.

The ServiceProfile details specifics of the service so a potential user, usually a component, could decide whether it fits its requirements or not.

Finally the service definition merges into one single entity the elements conforming the web service. The Service, an instance of the Service class, is related to at least one instance of the ServiceModel class by the property “describedBy”. Similarly every Service class is related to several ServiceProfile instances by the “presents” property although only one ServiceGrounding class is admitted for a given Service.

In this case the service “QoEManagerAgentService” is represented by an ontology that informs which capabilities does the service offer “QoEManagerAgentProfile”, how to use the service and how to invoke it “QoEManagementProcess”, “QoEManagerAgentGrounding” respectively.

5 Performance analysis

The technical and economical feasibility of the proposed architecture is closely related to the performance of the proposed atomic processes for high concurrency rates. There are two main factors to be taken into account: storage capacity to provide fine grained network status information and enough computing resources to adapt the service on a real time basis. Storage capacity depends on the network topology in the sense that the more links the streaming media traverses, the more information is needed for the processes “ProtocolMonitor” and “NetworkMonitor” to properly aggregate and manage, on a hop-by-hop basis, significant metrics. According to existing network topologies such as the one depicted in figure 4, the aggregation level remains at a low level compared to other services such as web surfing, this entails a management architecture able to cope with at most 3000 concurrent users. It is important to take into account that so far telcos are delivering multimedia content to customers from distributed facilities in order to bring content as close to end users as possible, usually this content is located in existing urban facilities and then accessed by customers on demand.

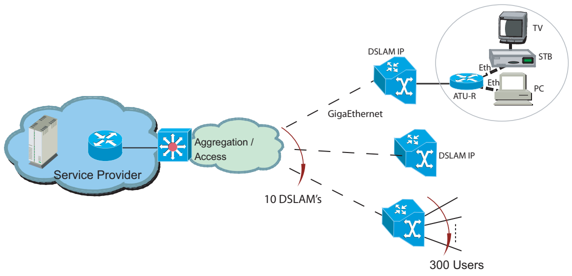

Fig. 4. Representative network topology for multimedia streaming services

For each established multimedia session the "ProtocolMonitor" process needs to supervise involved protocols such as RTP, RTCP and RTSP. This makes, for the considered monitoring technologies, a bandwidth consumption of 4604 bytes every 5 seconds for every active session. This amounts up to 0.2% of the total available bandwidth at the main network aggregation points (DSLAM's). On the other hand the process "NetworkMonitor" needs to consider not only every aggregation point but at the same time the backhaul layer 3 routing device, this doubles the total required bandwidth for each active session (14.7 Kbits/s).

When it comes to "QoETeller" and "StrategyPlanner" processes the main issue is the ability to compute QoE levels for active multimedia sessions. According to the considered topology at most 3000 concurrent accesses to compute the expression defined in (1) are required.

As performed experiments reveal, the average computing time of the process "QoETeller" in the most pessimistic scenario, 3000 users, reaches a maximum of 400ms far below the considered sampling rate (5 seconds)[4].

5.1 Real time QoE Management

Figure 5 represents the QoE behavior of a specific H.264/AVC streaming session delivered over RTP/UDP. In the context of this paper this contour plot constitutes a MultimediaProfile instance of a specific content, in this case a H.264/AVC coded movie delivered over RTP. Information regarding the way to determine such MultimediaProfiles can be obtained in [17].

Let's suppose that a user is receiving a movie coded at 3500 Kilobits/s with H.264/AVC. This constitutes a typical case for a type-3 ISMA[5] profile compliant stream. According to network metrics obtained by some network monitoring process (be it network probes, routing devices or RTCP[6] reports) the process responsible for service management, this is QoEManagementProcess, determines that current network conditions are $PLR = 0,06\%$, $Delay = 30ms$. As the

---

[4] Mathematica software executed on a Pentium IV 3.20 GHz

[5] Internet Streaming Media Alliance

[6] Real Time Control Protocol

central contour plot represented in figure 5 reveals, the actual session status lays in P_1 and thus the user is receiving a near zero-distortions quality level. The process in charge of computing the exact value, QoETeller, returns an exact value of $QoE = 0,8$.

Then as a result of network congestion, due to surge in demand because of some special event, the point of operation abruptly moves up to higher packet loss rates and delay starting to cause some quality degradation, represented by P_2, otherwise stated QoETeller returns $QoE = 0,5$.

A feasible approach to restore service level consists of downgrading the coding bit rate of the session taking place on a specific point in time in order to restore the quality up to optimal levels, represented by P_3. This alternative is able to restore the service without any extra resources and no other users noticing it. Furthermore reducing the amount of delivered information helps stabilizing the network as lesser bandwidth is required to deliver the content. It has to be pointed out, however, that for some type of content like high definition movies or sport events reducing the video rate may not be acceptable for some sensitive audiences. This approach is similar to the one proposed in [39].

Precisely at this point it is when an ontology driven approach pays off, by having meta information regarding inner details of the service it is possible to provide tailor made policies to enforce quality level in a consistent way.

Another way to cope with service degradation is by means of the so called version switching in which an established media session at certain point in time starts receiving content that is encoded in a different way. As a matter of fact bit rate reduction can be considered a simple case of version switching. Selecting the most suitable encoding scheme largely depends on the specific characteristics of the network and whether the service provider has control to provision QoS mechanisms. Flexible macroblock ordering (FMO) as defined in H.264/AVC, constitutes a robust encoding scheme capable of high error rates with minor bandwidth overhead [40]. By switching established multimedia sessions to FMO encoding the adverse influence of packet loss is lessened as far as user‘s perception is concerned. Version switching is possible to achieve on the fly thanks to the existence of SP and SI-Frames [27].

In case the underlying network provided advanced quality of service (QoS) mechanisms such as Differentiated Services (diffServ) [41] or any form of selective packet discarding then it would be possible to encode the media into several sub-streams each containing increasingly important information for the decoding process. This encoding scheme, denoted as data partitioning, enables graceful service degradation by discarding less relevant sub-streams.

6 Conclusions and future work

The ultimate goal for a multimedia over IP management solution is to satisfy customer's expectation in terms of displayed video and audio quality level. The architecture proposed in this article takes a user-centric approach that adapts delivered quality according to available network resources.

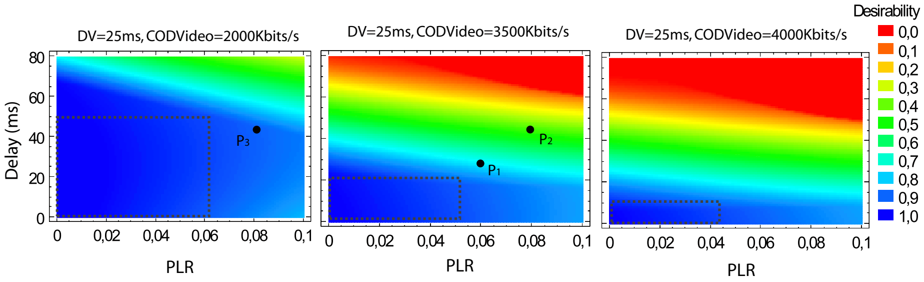

Fig. 5. Contour plots from the overall desirability function for H.264/AVC over RTP

The distributed nature of the proposed platform reinforces the required premises, that is maximum decoupling and composition, for adaptive solutions capable of scaling up to provide extra functionalities at a minimum development effort.

The modeling paradigm that web services offers has proved quite useful when it comes to component specification, providing simple mechanisms to hide the complexities involved in computing required metric values or fetching specific IP network data. However it was required to further extend those web services with technologies capable of describing the dynamics of the management process. In this matter OWL-S enables a formal representation capable of integrating functional elements (web services) and related processes into a single conceptual entity.

The resulting entity applies the concept of adaptive service elements (ASE) to the management of a multimedia service. It does so by defining an entity that, based on existing context information (in our case network status), controls the state of the service based on high level rules (optimize delivered quality to end users).

The resulting state model implements, in a formal computer-interpretable form, proven management techniques as defined in ISO/IEC 20000-1 Recommendation.

The benefits obtained with this approach are twofold, on the one hand it facilitates the integration of new management processes into existing ones in operations & support departments. On the other hand it relies on proven service management frameworks. This property is an important premise for reliable, human trusted, self-optimizing systems.

This paper addresses the management of a streaming service to ensure a proper delivery or, at least, to provide controlled service degradation. We believe this QoE management approach could be extended to more complex scenarios; so far it has been considered a closed topology, as depicted in figure 4, with clearly identified control levers such as admission control, bandwidth reduction or transcoding. However in open environments, such as the Internet, it would be required to enhance control policies with mechanisms such as content distribution frameworks or anycast routing, in order to have content delivered to customers from the closest node available. For instance once a certain QoE degradation threshold has been reached the management architecture will deploy new instances of the streaming service in distributed locations around the network, thanks to recent advances in virtualization technologies it will be feasible in the foreseeable future to deploy virtual instances of existing services just in time according to service demand variations.

Acknowledgment

This work has been partially funded by the Spanish Ministry of Education and Science under the project CASERTEL-NGN (TSI2005-07306-C02-01).

References

1. A. G. Ganek, T. A. Corbi, *The dawning of the autonomic computing era.* IBM Systems Journal, Vol 42, No. 1, 2003.
2. J. Schönwälder, A. Pras, J.P. Martin-Flatin, *On the Future of Internet Management Technologies.* IEEE Communications Magazine, Vol 41, No. 10, 2003.
3. L. Lymberopoulos, E. Lupu and M. Sloman, *An Adaptive Policy Based Framework for Network Services Management*, 1rd ed. Journal of Networks and Systems Management, Special Issue on Policy Based Management of Networks and Services.
4. G. Pavlou, P. Flegkas and S. Gouveris and A. Liotta, *On Management Technologies and the Potential of Web Services.* IEEE Communications Magazine, July 2004.
5. B. Levy, *The common capability approach to new service development*, 1st ed. BT Technology Journal, Volume 23 No 1, 2005.
6. J.M. Gomez, R. Lara and H. Lausen, *Semantic Web Enabled Web Services*, 1st ed. SWWS- Semantic Web Enabled Web Services, EU IST Project 2002-37134, Deliverable 2.1.
7. M.P. Papazoglou and D. Georgakopoulos, *Service-Oriented Computing.* Communications of the ACM, October 2003/Vol. 46, No. 10.
8. D. H. Yoon, B. Jeon, *Indication of key picture in AVC compatible base layer Joint Video Team (JVT) of ISO/IEC MPEG.* ITU-T SG16 Q.6 Input Document to JVT, 2005.
9. W. Li, F. Ling, X. Chen, *Fine granularity scalability in MPEG-4 for streaming video. Circuits and Systems.* (2000).
10. D. Lewis, T. ODonnell, K. Feeney, et al., *Managing User-centric Adaptive Services for Pervasive Computing.* Proc. IEEE International Conference on Autonomic Computing (ICAC'04)
11. J. E. López de Vergara, V. A. Villagra, J. Berrocal, *Application of OWL-S to define management interfaces based on Web Services.* Lecture Notes in Computer Science, Volume 3754, 2005
12. J. Keeney, K. Carey, D. lewis, et al., *Ontology-based Semantics for Composable Autonomic Elements.* Communications at 19th International Joint Conference on Artificial Intelligence, 2005.
13. A. van Moorsel, *Metrics for the Internet Age: Quality of Experience and Quality of Business.* Fifth Performability Workshop. (2001).

14. W. E. Walsh, G. Tesauro, J. O. Kephart, *Utility Functions in Autonomic Systems.* Proceedings of the International Conference on Autonomic COmputing, ICAC'04, 2004.
15. O. Versheure, P. Frossard, M. Hamdi, *User-Oriented QoS Analysis in MPEG-2 Video Delivery.* Real Time Imaging, 305-314, 1999.
16. S. Kanumuri, P. Cosman, A. Reibman, *A Generalized Linear Model for MPEG-2 Packet-Loss Visibility.* Proceedings of the Packet Video Workshop. (2004).
17. D. López, F. Vidal, L. Bellido, A. Alonso, *Adaptive multimedia streaming over IP based on customer oriented metrics.* 7th International Symposium On Computer Networks (ISCN 2006).
18. G. Muntean, P. Perry, L. Murphy, *Objective and Subjective Evaluation of QOAS Video Streaming over Broadband Networks.* eTransactions on Network and Service Management, First Quarter, 2005.
19. JTC 1/SC 7: ISO/IEC 20000-1:2005, Information technology – Service management – Part 1: Specification International Organization for Standarization JTC 1/SC7, ISO Recommendation (2005)
20. J. van Bon and M. Piper, *IT Service Management, an introduction based on ITIL*, 1rd ed. IT Service Management Forum, 2004.
21. Office of Goverment Commerce, *Best Practice, Service Delivery Office of Goverment Commerce* (2003)
22. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, *RTP: A Transport Protocol for Real-Time Applications*, 1st ed. RFC 1889, 1996.
23. D. S. Hands, *A Basic Multimedia Quality Model.* IEEE Transactions on Multimedia, Vol. 6, 2004.
24. H. Knoche, J. D. McCarthy, M. Angela Sasse *Can Small be Beautiful ?.* MM05, 2005.
25. S. Wenger, *H.264/AVC Over IP.* IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, 2003.
26. J. Ostermann, J. Bormans, P. List et al, *Video Coding with H.264/AVC: Tools, Performance and Complexity.* IEEE Circuits and Systems Magazine, First Quarter, 2004.
27. M. Karczewicz, R. Kurceren, *The Sp-and SI-Frames for H.264/AVC.* IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, 2003.
28. T. Stockhammer, M. Hannuksela, T. Wiegand, *H.264/AVC in Wireless Environments.* IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, 2003.
29. H. Haas, *Web Services Activity Statement.* World Wide Web Consortium, 2005.
30. M. Baugher, B. Strahm, *Real-Time Transport Protocol Management Information Base.* IETF Network Working Group, 2000.
31. C. Estan, K. Keys, D. Moore, G. Varghese, *Building a better NetFlow.* Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications.
32. Wolfram Research *webMathematica.* http://www.wolfram.com/products/ webmathematica/technology/.
33. E. Christensen, F. Curbera, G. Meredith, S. Weerawarana, *Web Services Description Language -WSDL- 1.1.* World Wide Web Consortium, 2001.
34. D. L. McGuinness, F. van Harmelen, *OWL Web Ontology Language -OWL-.* W3C Recommendation, 2004.
35. D. Martin, M. Paolucci, S. Mcilraith et al, *Bringing Semantics to Web Services: The OWL-S Approach.* Lecture Notes in Computer Science, Volume 3387, 2005.

36. J. E. López de Vergara, V. A. Villagra, J. Berrocal, *Applying the Web ontology language to management information definitions.* Communications Magazine, IEEE, Volume 42, 2004.
37. Wolfram Research *webServices.* http://webservices.wolfram.com/.
38. *Mindswap.* http://www.mindswap.org/ evren/composer/.
39. G. Muntean, P. Perry, L. Murphy: Objective ans Subjective Evaluation of QOAS Video Streaming over Broadband Networks. IEEE Electronic Transactions on network and service management. (2005).
40. T. Wiegand, G. Sullivan, G. Bjntegaard, A. Luthra: Overview of the H.264/AVC Video Coding Standard. IEEE Transctions on Circuits and Systems for Video Technology, VOL. 13, No. 7. (2003).
41. IETF Working Group:Differentiated Services (diffserv). [online].http://www.ietf.org/html.charters/OLD/diffserv-charter.html
42. J. M. Fuentes, J. E. López de Vergara, P. Castells: An Ontology-Based Approach to the Description and Execution of Composite Network Management Processes for Network Monitoring. Proceedings of 17th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management(DSOM 2006).