WiMAX QoS
Service Flow Management

Nuno Miguel Marques Rodrigues

Instituto Superior Técnico - Taguspark
Av. Prof. Dr. Cavaco Silva, 2744-016 Porto Salvo, Portugal

E-mail: nuno.rodrigues@tagus.ist.utl.pt

Abstract - *Providing Quality of Service (QoS) to clients from the access to the core network is possible with an 802.16 based access network. For this purpose, it is proposed the use of the Session Initiation Protocol (SIP) for signaling of client's QoS requirements. These requirements are processed by a modified SIP Proxy and sent to a Broker in the access network. This broker dynamically provisions connection requests from clients and, if necessary, communicates with the core Broker, so that a mapping between QoS paradigms of both networks is accomplished. This mapping allows that the QoS characteristics are maintained from the access to the core network.*

The prototype's performance tests show that dynamically provisioning connections in the access network can be accomplished in a time-window smaller than 100 ms. This time window is a reasonable price to pay for dynamic configuration of QoS-enabled connections for the access network.

Keywords - WiMAX, Quality of Service, Dynamic management

1 Introduction

The main purpose of this paper is to give an overview of a system with the capability to dynamically allocate QoS-enabled connections from the access to the core network. The access network is based on the 802.16d standard [1], while the core network is based on IP-DiffServ. For connection setup, users need to use the the SIP protocol [2]. The messages exchanged carry the client's QoS requirements and are processed by a modified SIP Proxy. This proxy extracts the relevant information from the SIP/SDP [3] messages and forwards requests to a QoS Broker. This allows dynamic establishment of connections in the access network.

The dynamic provisioning of connections in the access network provides clients the QoS they request and additionally, with the introduction of a mapping mechanism from the access to the core network, preserves QoS requirements to the core network, thus providing QoS across domains.

Within the broker there were considered functions like network element configuration and admission control. The former allows the configuration of the WiMAX Base Station, according to requests from clients, while the latter uses defined policies to allow/deny access to QoS-enabled resources. In the admission control function, it was additionally considered a degradation model that enables the degradation of low-priority connections (nrtPS). This degradation mechanism allows that more high-priority connections enter the network.

The rest of this paper is organized as follows: the next section contains related work in terms of heterogeneous networks and WiMAX-specific research solutions. In section 3 is explained the architecture of the system, with focus on the system components. Following the architecture are presented the main implementation solutions followed. Section 5 presents the evaluation of the prototype with focus on performance and finally in section 6 are drawn some conclusions and future work is pointed out.

2 Related Work

One of the main characteristics of the 802.16 standard [1] [4] is that it was standardized with embedded QoS support. This provides network operators the ability to make the distinction of traffic in the access network segment. This requires WiMAX specific solutions to provide control over resources in the air interface. However, the problem of quality of service is not circumscribed to the access network.

The heterogeneity of today's networks poses a challenge in terms of assuring that QoS requirements are assured in the access and also crossing domains. The challenge can be divided in two different components: protocol layering adaptation in the edge routers and

the preservation of resource's characteristics over different domains.

In terms of WiMAX specific solutions, [5] and [6] are two approaches which aim at resource allocation optimization. In [5], there is specified a degradation model that takes advantage of the characteristics of nrtPS connections. The idea is to degrade lower priority connections, so that higher-priority connections are allowed in the access network. In [6], the authors achieve a minimization of bandwidth provisioning, while keeping MAC signaling to a minimum. The authors use pre-determined steps of reserved resources, which vary according to network load.

In [7], the authors propose an architecture to provide multi-layer integrated QoS control, where the IP QoS architectures supported are IntServ and DiffServ. The architecture is clearly cross-layered and defines mappings from the 802.16d standard to the IntServ and DiffServ IP QoS architectures.

In [8], the authors propose a different cross-layered QoS architecture. The singularity in it is that it uses the IEEE 802.1p [9] recommendation to classify packets. Besides this, it also introduces the Channel Adapter and SNR sniffer, which is responsible for the evaluation of propagation conditions. It allows the inspection of the wireless medium, providing information related to SNR, fading, etc.

Concerning the topic of heterogeneous networks and end-to-end QoS support, the WiMAX Extension to Isolated Research Data Networks (WEIRD) group [10], [11] defined an architecture with this support.

WEIRD supports both signaling capable and legacy applications. The former use SIP/SDP as a signaling protocol, while the latter have no SIP capabilities. In order to make resource reservation, they use the NSIS protocol, which allows resource reservation across domains.

The EuQoS project [12] also addressed the problem of QoS support over heterogeneous networks. They consider different domains as Autonomous system, with different administrative authorities, thus they use qBGP [13], which is a variant of BGP [14] that integrates QoS features, to reserve QoS paths.

When the use of some ideas that are specific to the 802.16 standard, like the ones presented in [5] and [6], which optimize resource usage in the WiMAX domain, are combined with cross-layered QoS architectures [7] [8], plus concerns about the rest of the IP network [10] [12], the result should be a framework that is able to provide the best of both worlds.

3 Architecture

3.1 System Components

The main system components of the system are those identified in figure 1.

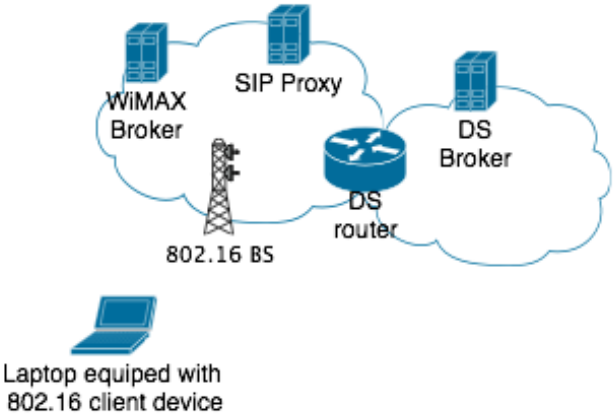

Figure 1: Components overview

There are considered two domains: the WiMAX, which is considered the access network and the DiffServ, which is represented as the core network. Inside each of these domains is a QoS broker. This broker has the responsibility of configuring the network elements affected to his domain and admission control. Thus, they are the brains of the operation, each with the technological specificity of his domain. Also presented is a SIP Proxy. This proxy is responsible for mediating SIP communication between clients, which gives him access to when are clients initiating connections and also who are the corresponding hosts.

3.1.1 WiMAX QoS Broker

Figure 2 depicts the main functions of the WiMAX QoS Broker and the components of the access network.

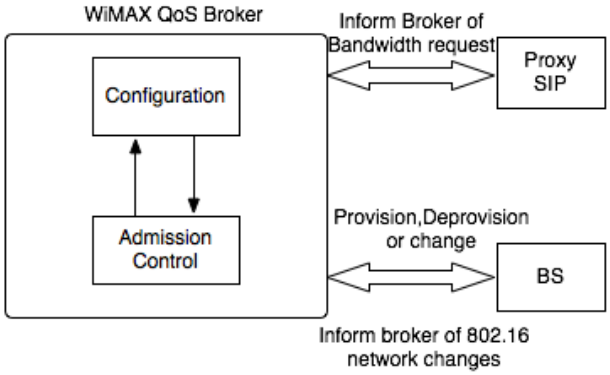

Figure 2: Access network entitites

The main functions of this broker are configuration and admission control in the WiMAX domain. Also represented are the interfaces to the WiMAX Base Station and SIP Proxy.

Regarding the Base Station, the information that the Broker sends is Provisioning and de-provisioning of Service Flows, Service Classes and Classifying rules. What it receives from Base Station is information about clients joining and leaving the network. This allows the broker to provision connections to clients that enter the network and remove unused resources when they leave. In a fixed access usage (802.16d standard), this may happen if a terminal is unplugged, but in a nomadic environment (802.16e standard), this may mean that a mobile station changed from one sector to another.

The SIP Proxy plays a very important role in the system. It inspects SIP/SDP messages that come from clients, gathers the information about the participants in the call and then dispatches this information to the QoS Broker. The broker listens to these requests and does the appropriate connection provisioning in the Base Station, if there are still resources available.

The gathering of information in the Proxy, regarding client's QoS requirements plus the resource provisioning functions in the WiMAX Broker allow the dynamic provisioning of connections in the access network. This way, when clients ask for resources in the access network (through means of a SIP INVITE message), their connections are provisioned with the required QoS, but when they no longer need these resources (signaled through a SIP BYE message), these resources are de-provisioned from the WiMAX Base Station.

Another point of interest in the WiMAX QoS Broker is the architecture of the broker itself. It is depicted in figure 3.

It was stratified in different layers: events, events processing and technology specific actions. This allows the broker to deal with different events independently, but simultaneously. The events trigger a series of processing, like checking profiles or gathering index information about Service Flows/Classifiers and finally, after processing, the result is a technological action. This action may be the provisioning or removal of service flow, classifier configuration, etc. This architecture clearly separates the type of events that occur in the access network, giving the broker the ability to deal with different types of events simultaneously. It also presents a modular architecture, which can be useful if there are some new events that need to be added or even if the technological support changes. These changes may occur if the Broker deals with different vendor equipment.

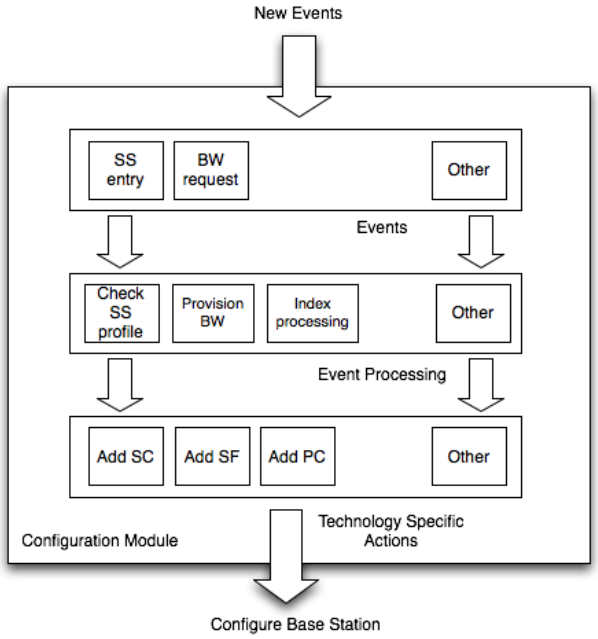

Figure 3: Configuration module architecture

3.1.2 DiffServ QoS Broker

The DiffServ QoS Broker plays a similar role to the WiMAX Broker. It has also functions of installing policies and classifying rules for the core network. Figure 4 gives an overview of it's role. When the WiMAX

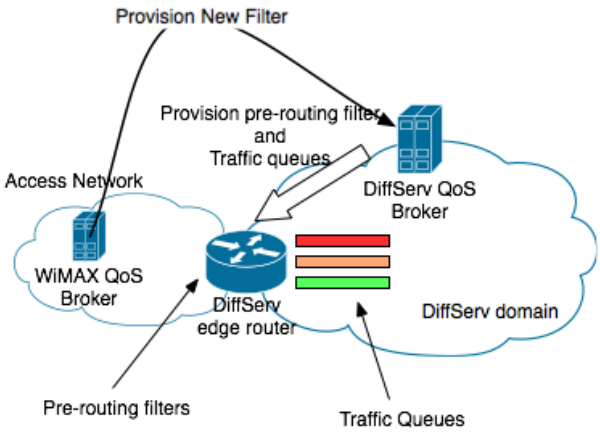

Figure 4: DiffServ QoS Broker role

QoS Broker receives resource allocation requests from the SIP Proxy, it will provision the resources that are needed for the access network clients. If any participant in the call is not a WiMAX client, he will forward the request to the DiffServ QoS Broker, so that it provisions the necessary filters for mapping.

When the DiffServ QoS Broker receives the information that it needs to create a mapping from the access to the core network, he will take action in the

corresponding edge router and provision the necessary classifying rules. When the SIP conversation ends, the inverse action is taken, i.e., the provisioned rules will be deleted from the edge router.

The installation of new classifying rules allows that traffic coming from the WiMAX domain with QoS guarantees, has also privileged treatment in the core network. Worth mentioning is that these classifying rules are installed as pre-routing filters. This allows that the traffic is marked in the entrance of the edge router and when it is routed, he will fall in the correct traffic queue. In figure are represented three traffic queues (red, orange and green) that represent EF, AF and BE classes.

3.2 Protocol Layering

As the previously mentioned domains are technologically different, there is also a difference in the associated protocol layers. Figure 5 shows the protocol stack and the differences between technologies.

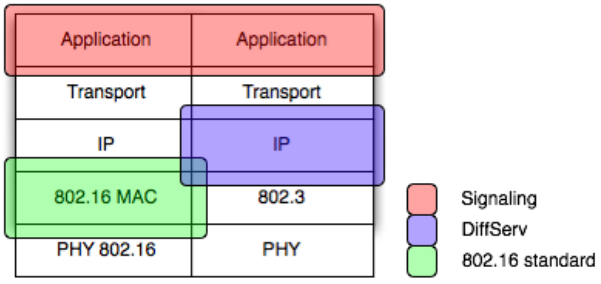

Figure 5: Protocol Layering

On one side, we have the QoS concept of the 802.16 standard, which is at MAC layer. This concept is connection-oriented and each connection has a specific set of parameters which defines the characteristics of traffic using that connection. The other approach is the IP-DiffServ, which is at the IP Layer, with a non-connection-oriented style.

This difference imposes that, on one hand, the brokers make their admission control and configurations functions at different levels and on the other hand, it will need a mapping strategy for packets that cross from one domain to the other.

On top of these protocol stacks, we have the Signaling function. Signaling carries the information about client's QoS requirements and thus provides information to the brokers, so that they make the necessary Network Element configuration. This mechanism works on the Application layer and is technology-agnostic.

3.3 Signaling

SIP is the standard signaling protocol used, which allows users to have access to QoS connections. The signaling model is represented in figure 6.

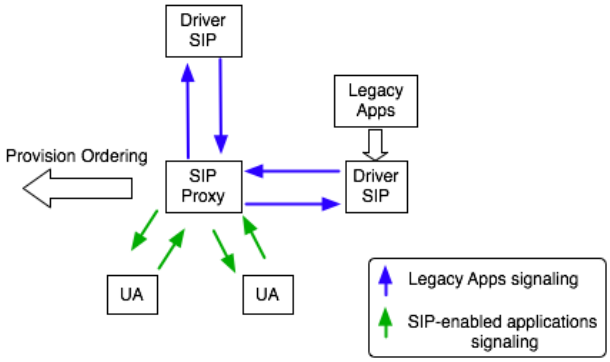

Figure 6: Signaling model

The SIP Proxy is in the center of the process. For applications that use the SIP proxy, the only requirement is that they use SDP with the *b=*:** modifier. For applications that do not support the SIP protocol, it is necessary to use a special driver, which receives information about the client's QoS requirements and encapsulates the requests in a SIP/SDP message. The driver will need to be installed on two ends, so that communication is mediated by the SIP proxy and the information is extracted and passed on to the Broker.

4 Solution implementation

4.1 Implementation considerations

The prototype was developed using the Java programming language. This provides a portable prototype across OSs, as Java is platform independent.

In terms of network element configuration, the protocol used is SNMP [15]. The API used was SNMP4J [16], which is an open-source API for Java. SNMP was used instead of a proprietary Northbound interface (vendor Airspan), as it should provide compatibility between different vendors (if vendors comply with the 802.16f standard [17]).

For improved performance using the SNMP API, it was necessary to have two concerns: first use multiple sets in each SNMP command. This allowed that, even though it was necessary to configure several parameters, it was accomplished with one SNMP command. The other concern is related with socket operations. As socket operations tend to be heavy in terms of performance, a new thread is launched whenever it is necessary to close sockets. These two concerns allowed some performance gains in terms of configuration.

4.2 WiMAX QoS Broker

4.2.1 Configuration

The following is an explanation of the events, processing events and technology specific actions that are supported by the WiMAX QoS Broker.

Events:

- New SS Event - detection of SS entry/exit
- Provision BW Event - detection of connection requests
- Degradation Event - order connection degradation
- Provision DiffServ Filter Event - order filter provisioning in the edge router

Processing Events:

- Create Subscriber Station - create an SS with a given profile in the access network (includes gathering of SFs and classifiers)
- Create Default Profile - similar to the previous, with the singularity of provisioning a BE profile
- Index Generation and Aggregation - calculation of indexes and associations between SFs and Service Class.
- De-provision Subscriber Station - gathers associated classifiers and Service Flows and removes them.
- Provision BW for SIP Call - Gather the necessary classifier info, associate with Service Flow and Service Class.

Technology Specific actions:

- Add/Remove Classifier - classifier provisioning and removal in Base Station
- Add/Remove Subscriber Station - creation and removal of Subscriber Station
- Add/Remove Service Flow - creation and removal of Service Flow
- Add/Remove Service Class - creation and removal of Service Class
- Change Service Classes' characteristics - in case of degradation, change the class.

4.2.2 Admission Control

The admission control function is considered to be divided in two components: pool of resources and degradation model.

For the pool of resources, it is considered a mechanism where connections are admitted based on two threshold values L and U, where L is considered the Lower threshold and U the Upper threshold. The value of used connections will be in the range [L,U]. The value of admitted connections for a given class is considered to be always higher than U.

It is also relevant to mention the profile-based mechanism that is used. This assumes that there are defined profiles, where each bandwidth request should fit in. Let's consider an example for VoIP applications. The profiles that may be defined in this case are 16, 32, 64 and 128 kbps (it will depend on the codec used). Each of these profiles is considered to have a relative weight W, defined in equation 1.

$$W = \frac{Profle_{BW}}{MaxProfile_{BW}} \quad (1)$$

For the degradation model, it is considered the degradation of nrtPS classes. This takes advantage of the characteristics of the class definition itself, i.e., the bandwidth of the nrtPS class varies in the range [MinRR,MaxSR], where MinRR is the Minimum Reserved Traffic Rate and MaxSR is the Maximum Sustained Rate. This leaves an interval MaxSR - MinRR where the bandwidth given to the class may vary. Thus, the degradation will reduce the value of MaxSR until it reaches the minimum (where MaxSR = MinRR).

To deal with bandwidth requests that come from clients, it is considered the state machine depicted in figure 7. When a new request arrives, the system will check the defined profiles, to see if one exists that suits the requested bandwidth request. At this point, if there is no defined profile that suits the request, the request for a new connection will be immediately rejected.

For the case where there is a profile matching the request, the system continues and calculates the relative weight of the class that was chosen (W). Then, this value is added to the value of currently used flows, to check if it reached a maximum value. If it has, it will first try to steal resources from other classes (reallocating bandwidth to this class) and if this option is not available, it will try to use the degradation model in order to allow this new connection in the network. To use the degradation model, the system first needs to calculate what are the currently enabled nrtPS connections and calculate the values that may be stolen to each of these classes. Only after, it is possible possible to make the degradation. The final step before allowing the connection (post-degradation) includes the update of the MaxSR values of the nrtPS classes.

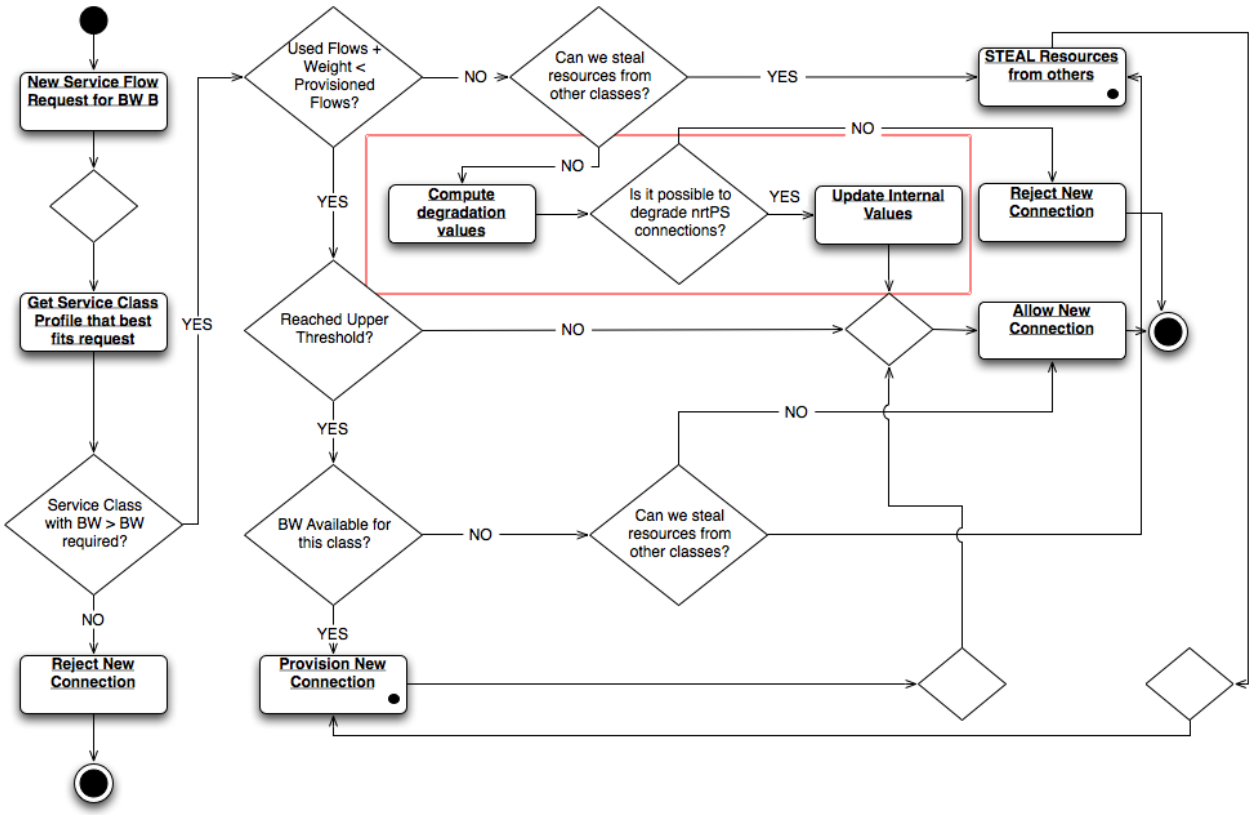

Figure 7: Pool state machine with focus on the degradation model

If the maximum value is not reached, it will check if the system has reached an upper threshold. If this is the case, it will check if there is still bandwidth available in the class, so that it can admit new resources. If no threshold is reached, the connection is simply allowed in the network.

4.3 DiffServ QoS Broker

In the prototype implementation, the DiffServ QoS Broker is physically in the same machine as the WiMAX QoS Broker. The Traffic queues for DiffServ traffic are provisioned statically and are dimensioned so that packets from and to the access network are not dropped. Still, it has a dynamic behavior in filter provisioning.

As previously mentioned, when a SIP conversation between a WiMAX and a DiffServ client occurs, there is a need to provision filters so that packets are inserted in the correct queue and served with the necessary QoS. This enables the preservation of resource's QoS requirements across networks.

To provision the necessary filters, it is installed in the *pre-routing* chain of the router a new rule. To build those rules, there are mandatory attributes:

- Source IP / Network Mask
- Transport Protocol / Source or Destination Port

The first group of attributes (SourceIP / Network Mask) are both at network layer. They identify unambiguously the origin of packets. The second group of attributes are at transport layer. Along with these attributes is an associated mark (depending on the QoS requirement's of associated packets).

The scheme described allows that the originating host is unambiguously defined for a determined service (in this case, associating a port with a service), which allows the host to have different connections, with different QoS requirements, for each of the requested services.

4.4 SIP Proxy

The SIP Proxy is the entity that deals with the signaling originated by clients. As typical SIP proxy implementations lack an analysis module, the proxy had to be extended to support it. Figure 8 depicts the architecture of the modified proxy.

The Proxy still needs to take care of the mediation of client signaling. For this purpose, the proxying functions are maintained. What was added was a logging mechanism, so that messages are kept in a file, an analysis component called *Message Analyzer*, which takes

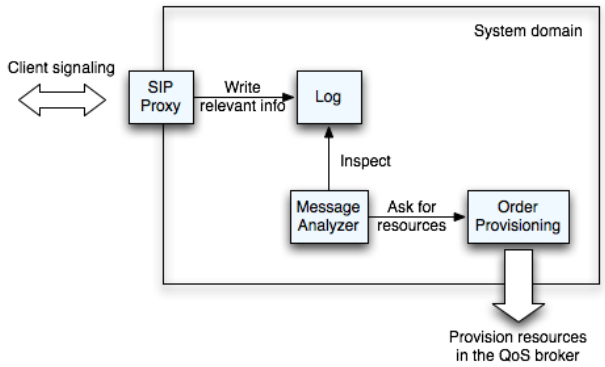

Figure 8: Proposed SIP Proxy Architecture

care of the message processing and a provisioning component called *Order Provisioning*, which dispatches requests to the Broker.

As messages are being logged, the Message Analyzer component processes them and extracts the relevant information, adding it to a call list. This call list keeps track of the current on-going calls. In each call is the information about the participants in the call, keeping record about what types of media are the participants using (audio, video) and the host information (Source and Destination address, source and destination port, etc).

Note that the information on the call list is being filled while the participants are exchanging the call setup messages. The information on the participants is then forwarded to the Order Provisioning module, where it will be dispatched to the QoS Broker.

Worth mentioning is that, only if the called terminal accepts the connection are resources provisioned. If the called user is away or if it rejects the call, messages will not be dispatched and resources will not be provisioned.

4.5 Mapping Strategy

A mapping between traffic from the 802.16d to the IP-DiffServ domain was also considered. The rules are defined in Table 1.

802.16d	IP(DiffServ)
UGS	Expedited Forwarding
-	Assured Forwarding 4x
rtPS	Assured Forwarding 3x
-	Assured Forwarding 2x
nrtPS	Assured Forwarding 1x
Best Effort	Best Effort

Table 1: Mapping 802.16d to IP-DiffServ

The UGS traffic is directly translated into Expedited Forwarding. This can be justified because UGS traffic has hard QoS requirements (e.g. VoIP / Leased line E1/T1). The rtPS traffic is considered to be mapped to the AF3 class (soft QoS requirements), while the nrtPS class maps to AF1 (even softer QoS requirements). This leaves out, for now, the AF4 and AF2 classes of DiffServ. These classes may be seen as future expansions to the core administrator. For example, if it wishes to make a distinction between gaming and video traffic, it could make that distinction in the core network, by using one the AF classes that is available. This way, the rtPS class would map to AF3 and AF4 classes, depending on the traffic type.

With the inclusion of a new scheduling class by the 802.16e standard (extended real time Polling Service - ertPS), the mapping should not be changed. What would happen to this class is that it should map to an EF Per Hop Behavior. This is justifiable because ertPS is suited for services such as Voice Over IP with silence suppression.

5 Prototype Evaluation

5.1 Test scenario

The test scenario is depicted in figure 9. The scenario uses 802.16d compliant equipment (from vendor Airspan) in the access network, while the core network is composed by a DiffServ edge router. Both brokers have been joined in one machine and there exists only one SIP Proxy, that serves all clients.

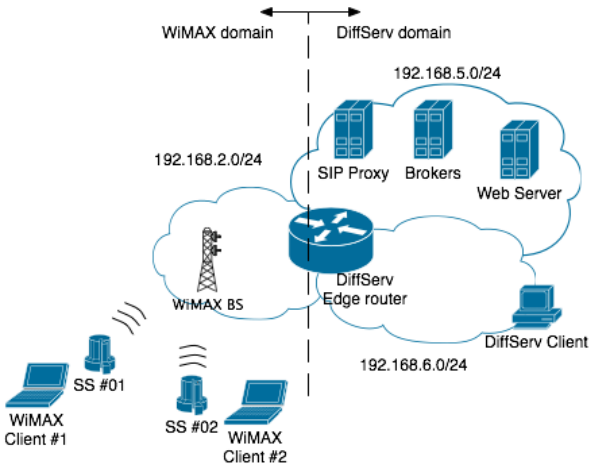

Figure 9: Test scenario

The evaluation of the prototype will be divided in three components. The first component will show the degradation mechanism working, degrading an nrtPS connection, as clients enter the network. Then, the presented tests will show the influence of background traffic in the establishment of a conference call (with

and without the WiMAX QoS broker in action) and finally, the times spent in the operations of provisioning, de-provisioning and degradation will be evaluated.

5.2 Results

An example of the degradation of resources of an nrtPS connection is presented in figure 10. The figure shows

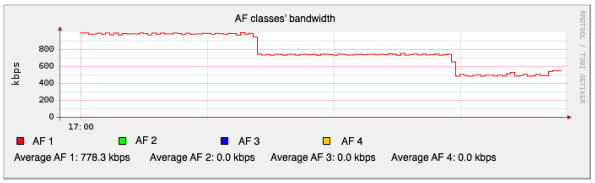

Figure 10: Degradation of an nrtPS connection

an existing nrtPS connection, that was transmitting data using this connection, getting degraded over time. It is clear in the figure that there are three steps. In the first moment, the nrtPS was operating at it's full rate (1Mbps in this case). Then, it was degraded in 256 kbps, with the entrance of a higher priority connection, dropping to 768kbps. Finally, it dropped another 256 kbps, to 512 kbps.

What is interesting is that, by the degradation of a lower priority connection (nrtPS), there were allowed two more connections. If the degradation model was not being used, they would not have been allowed.

Figures 11, 12 and 13 show the influence of background traffic during a conference call. The test was

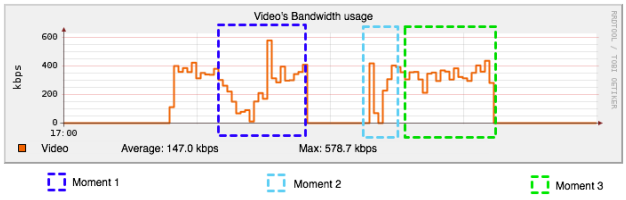

Figure 11: Bandwidth usage of the video connection

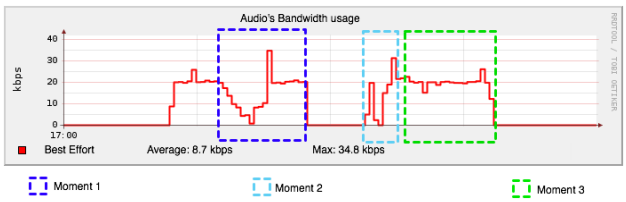

Figure 12: Bandwidth usage of the audio connection

divided in three different moments. Moment 1 represents the influence of background traffic when there weren't provisioned QoS connections to clients. It is

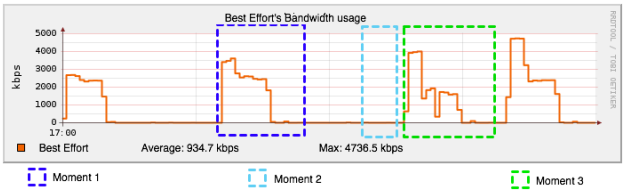

Figure 13: Background traffic

possible to see in figures 11 and 12 that video and audio connections suffer with the introduction of background traffic.

For moment 2, the QoS solution was turned on and it can be seen in the audio and video figures that there is an influence caused by provisioning of new connections. Moment 2 represents a time window of about 1-2 seconds, where the traffic is unstable. This may represent some limitation in the WiMAX Base Station, which affects the traffic streams when new Service Flows and classifiers are provisioned.

Finally, we get to moment 3, where the audio and video connections have stabilized and background traffic is introduced. It can be seen that, unlike moment 1, the audio and video connections do not suffer with the introduction of background traffic. Instead, it is the background traffic that is affected, because it is being served only after the other two connections.

Note that, in the background traffic, there were taken two control samples (beginning and end of the graph in figure 13), which show the values reached by the background traffic when there were no other connections. Comparing these samples to moment 1, there is no clear difference (in average), but in comparison to moment 3, it can be seen that, in average, the background traffic's performance decreased, as was expected.

In table 2 are represented the time values that were obtained in configuration of Base Station.

Parameter	**Average (ms)**	σ	**C.I. (ms)**
Time to provision	70.6	14.1	6.5
Time to de-provision	36.8	18.7	8.6

Table 2: Time spent in Base Station configuration (confidence interval of 95 %)

The values presented correspond to the time that is necessary between detection of a request and the real provisioning in the Network Element (Base Station). The processing time is included in these values. The value obtained for provisioning is 70.6 $\pm$ 6.5 ms, while the time to de-provision resources is 36.8 $\pm$ 8.6 ms.

Comparing both values, it is possible to see that the time needed for provisioning is higher than the time needed for de-provisioning. This is due to the number of operations necessary to each action.

As the interface with the WiMAX Base Station is done through means of SNMP, the provisioning process needs three different operations before the action is finished. The first operation is the creation of a new entry in an SNMP table. Next, come the values to fill the entry. For this purpose, it is necessary to create a new SNMP PDU, fill in the values and send this PDU to the Base Station. Finally, the SNMP entry is marked as active and the action is considered finished.

Concerning the de-provisioning process, it is only necessary to send one SNMP message to the Base Station, deleting the entry. This involves only one operation, which explains the difference in time between provisioning and de-provisioning.

Table 3 represents the time spent in the degradation process.

Parameter	**Average (ms)**	σ	**C.I. (ms)**
Time to degrade	57.7	8.8	5.5

Table 3: Time spent in the degradation process (confidence interval of 95 %)

In the table is represented the time that is spent in the degradation process. This time represents the processing time needed to evaluate and calculate the degradation values for the nrtPS class plus the time necessary to make the appropriate change in the Base Station. The value for degradation was found to be 57.7 ± 8.8 ms.

In terms of SNMP operations, degradation is comparable to the de-provisioning process, as it is only necessary to make 1 operation (in this case, change the connection's characteristics). However, the degradation process involves floating point operations, so that new connection's characteristics are calculated and additionally, communication between the different entities. These operations add delay to the overall time.

Results show that the time between detection of a new connection and provisioning in the Base Station, can be accomplished in less than 100 ms. However, the Base Station appears to drop packets when new connections are provisioned, causing a 1-2 seconds instability period. However, passed this period, the connections stabilize, and QoS is delivered to clients. Additionally, the inclusion of the degradation model allows that more connections enter the network, at the cost of deteriorating the nrtPS connection's characteristics.

6 Conclusions

In this paper were presented the components necessary to make a dynamic management of resources in the access network, for an 802.16 based network. Further, it was defined a mapping strategy to allow the preservation of QoS characteristics from the access to the core network.

In terms of the access network, there was presented the WiMAX QoS Broker, which has the functions of network element configuration and admission control. Additionally, it was presented a modified SIP Proxy to process SIP requests from clients and send the information to this broker.

The evaluation of the prototype showed that, in terms of performance, the prototype is able to provision connections in the access network under 100 ms, which should not affect user's QoS opinion. Despite this fact, when this was tested in a real SIP-call (with audio and video enabled), it was noticed that the allocation of these connections affected the stability of the QoS-enabled connections for one or two seconds. However, passed this period, it was demonstrated that background traffic does not affect the QoS of these connections.

6.1 Future Work

Although the prototype is working, further tests should be conducted to spot any problems that weren't detected so far.

Additionally, it should be considered the physical separation of both brokers and the implementation of a communication interface between them.

Regarding the DiffServ QoS Broker, it should also be considered the implementation of a mechanism that dynamically adjusts the traffic queues capacity, so that they adapt to network load.

References

[1] IEEE 802.16 working group. *IEEE Std 802.16-2004 (Revision of IEEE Std 802.16-2001): IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems.* IEEE Press

[2] Rosenberg, J.; Schulzrinne, H.; Camarillo, G.; Johnston, A.; Peterson, J.; Sparks, R; Handley, M.; Schooler, E.; *RFC 3261 - SIP: Session Initiation Protocol*

[3] Handley, M; Jacobson, V.; Perkins, C.; *RFC 4566 - SDP: Session Description Protocol*

[4] IEEE 802.16 working group. *IEEE Std 802.16e-2005 and IEEE Std 802.16-2004/Cor 1-2005 (Ammendment and Corrigendum to IEEE 802.16-*

2004): IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Ammendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Lecensed Bands and Corrigendum 1. IEEE

[5] Wang, H; Li, Wei; Agrawal, Dharma P. : *Dynamic Admission Control and QoS for 802.16 Wireless MAN*, IEEE 2005.

[6] Gakhar, Kamal; Achir, Mounir, Gravey, Annie: *Dynamic Resource Reservation in IEEE 802.16 Broadband Wireless Networks*, IEEE 2006.

[7] Chen, Jianfeng; Jiao, Wenhua; Guo, Qian: *Providing Integrated QoS Control for IEEE 802.16 Broadband Wireless Systems*, Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd.

[8] Delicado, J; Orozco-Barbosa, L.; Delicado, F.; Cuenca, P.: *A QoS-aware protocol architecture for WiMAX*, IEEE CCECE/CCGEI 2006

[9] *Information Technology - Telecommunications and Information Exchange between System - Local and Metropolitan Area Networks - Common Specifications - Part 3: Media Access Control (MAC) Bridges (ANSI/IEEE Std 802.1D)*, ISO/IEC (ANSI/IEEE) Std. 15802-3, 1998

[10] Angori, Enrico; Borcoci, Eugen; Mignanti, Silvano; Nardini, Cristina; Landi, Giada; Ciulli, Nicola; Sergio, Giacomo; Neves, Pedro. *Extending WiMAX technology to support End to End QoS guarantees.*

[11] Bohnert, Thomas Michael; Castrucci, Marco; Ciulli, Nicola; Landi, Giada; Marchetti, Ilaria; Nardini, Cristina. *Architectural Solution for QoS Management in a WiMAX Network.*

[12] http://www.euqos.eu/, online January 2008

[13] Boucadair, M.: *QoS-Enhanced Border Gateway Protocol*, draft boucadair-qos-bgp-spec-01.txt, Jul. 2005

[14] Rekhter, Y.; Li, T.: *A Border Gateway Protocol 4 (BGP-4)*, RFC 1711, March 1995

[15] Case, J.; Fedor, M.; *A Simple Network Management Protocol (SNMP)*, RFC 1157, May 1990

[16] Fock, F.; Katz, J.; *SNMP4J - The Object Oriented SNMP API for Java Managers and Agents.* - Web-site: http://snmp4j.org/index.html

[17] IEEE 802.16 working group.; *IEEE Standard for Local and metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems - Ammendment 1: Management Information Base.* IEEE Press