Enabling Mobile Heterogeneous Networking Environments With End-to-End User Perceived QoS - The BRAIN vision and the MIND approach

Andreas Kassler, University of Ulm, Germany, kassler@informatik.uni-ulm.de
Teodora Guenkova-Luy, University of Ulm, Germany, guenkova@vs.informatik.uni-ulm.de
Davide Mandato, Sony International (Europe) GmbH, Germany, mandato@sony.de
Peter Schoo, DoCoMo Communications Labs GmbH, Germany, schoo@docomolab-euro.com
Iván Armuelles, Technical University of Madrid, Spain, ivan@dit.upm.es
Tomás Robles, Technical University of Madrid, Spain, robles@dit.upm.es
Pedro Ruiz, Agora Systems S.A., Spain, pedro.ruiz@agoratechnologies.com
Alejandro Bascuñana, Ericsson España S.A, Spain, ecealbm@madrid.ericsson.se

ABSTRACT
Results of the conceptual work carried out during the BRAIN project concerning the BRAIN QoS Framework and the Brain ENd Terminal Architecture (BRENTA) is presented. This includes mechanisms to specify and manage QoS End-to-End including mobile end-systems and IP based wireless access networks in the transmission path. One of the key points is to manage QoS end-to-end from the users and media point of view and provide well-defined adaptation strategies in the case that QoS at the network level cannot be guaranteed for the whole session. This is likely to occur in mobile environments, because of potential handovers (horizontal and vertical) and error rate patterns of wireless links in contrast to fixed networks. The design of selected parts of BRENTA is currently under study within the MIND (www.ist-mind.org) project with respect to ad hoc network aspects, and security.

1. Introduction

The introduction of QoS within IP networks, and especially within radio access IP networks, is one of the key factors that will push packet-switched technology as the base of all multimedia services (i.e. videoconference, video on demand, etc) in the wireless environment. However, radio access technologies are not a fully reliable way of accessing IP networks. The wireless world faces in fact several challenges like: handovers within and between several radio access networks (HiperLAN/2, WLAN 802.11, etc), change of the radio-link quality, loss of communications, etc.

Current and future networks and operating system environments will be heterogeneous. End-systems and networks show a fair amount of performance variations. Local and network resource availability will be limited and applications may suffer from variations in resource availability. Mobility of terminals and users will also contribute in making more difficult to provide users with predictable end-to-end Quality of Service. Therefore, applications are desired to adapt themselves and adjust their resource demands dynamically to cope with changes in environmental conditions. Typical distributed multimedia applications use feedback information to adapt but cannot provide fairness to all applications, since global information is not available. Also, each application implements similar mechanisms to adapt, like changing the frame rate, frame size or video quality of a video stream. On the other hand, resource management systems built into operating systems have not the knowledge about the semantics of applications that use the resources. Integration of media management, mobility and resource management is thus necessary, to provide predictable end-to-end QoS. This calls for an architecture, where system level properties, resources, application level semantics and user preferences can be observed and managed so as to best decide when and how adaptation has to be performed.

Figure 1 shows the evolutionary approach taken by BRAIN and MIND projects towards 4G. Brain extended the current telecommunication network and the Internet to address micro-mobility and QoS [1]. MIND provides a further evolution to allow ad-hoc networks and Personal Area Networks (PAN) leveraging the public infrastructure to gain access to the Internet, to long distance calls facilities, and to services, but foremost to security support. For providing user perceived QoS management, we provide a QoS framework that could enable distributed multimedia applications with adaptation mechanism and QoS provision on an end-to-end basis, across heterogeneous networks including Wireless access. The BRAIN End Terminal Architecture [1] (BRENTA) was designed for providing applications with QoS adaptation and transparent mobility handling, through the induction of a middleware solution above transport layer.

This paper is structured as follows. In Section 2, we review the BRENTA and introduce an end-to-end negotiation protocol that is used to agree on a set of capabilities and QoS requirements. Section 3 addresses QoS aspects in ad hoc networks. Section 4 considers security aspects. Finally, Section 5 concludes the paper.

2. The BRAIN End Terminal Architecture

Key requirements of BRENTA design are: Modularity, to enable a graceful transition from legacy applications to future middleware solutions; Openness, to being interoperable with other architectures and IETF protocols; and Flexibility, to cope with multi-stream, multi-session multimedia data and to enable a dynamical upgrade of the system during runtime. In order to address both existing and future applications, BRENTA (see Figure 2) introduces a classification of applications based on the degree of QoS awareness they feature.

Type A applications typically access IP services through legacy transport layer interfaces (API - 0).

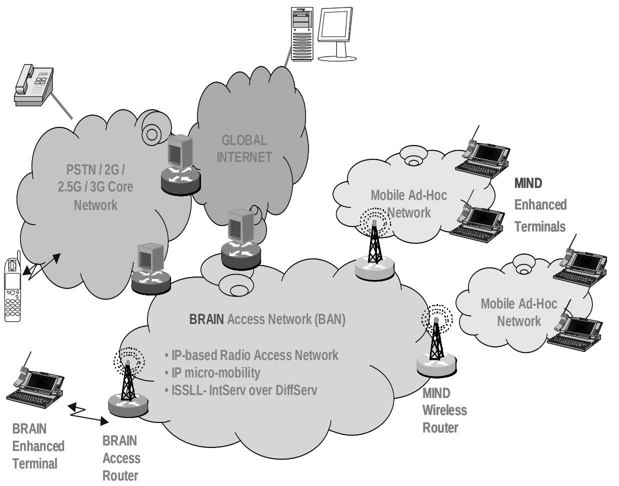

Figure 1: BRAIN and MIND evolutionary approach

External configurator tools built on top of the Enhanced Socket Interface (ESI [2]) can be used to provide differentiated services to legacy applications. Type B applications use session layer protocols (e.g. H.323, SIP) across API B, which may be partly embedded within applications. Since type B applications autonomously manage QoS- and mobility-related issues, they only use standard protocols (e.g. IETF protocols), enhanced by some mobility-related functionality. BRENTA Type C applications leverage existing implementations of protocol and multimedia functionality (like frame grabbers, codecs, packetizers, renderers…) and resource management functionality through the API C, which is offered by a set of software components. The QoS adaptation logic still resides within the applications. Type D applications will entirely rely on an external entity, the QoS Broker, via the API D. Based on user-QoS profiles or application-supplied policies, the QoS Broker co-ordinates local, network, and remote (peer's) resources, in order to cope with QoS violations.

In the following sections we assume that the various peers participating in multiparty communication sessions may generally take the role of media stream-sender and/or receiver. Additionally, we assume some intermediate components may passively assist peers with the connection establishment and management, e.g. by providing service and/or peer discovery functionality.

2.1 BRENTA QoS Adaptation Strategy

BRENTA follows a QoS adaptation strategy based on the automatic selection of predefined alternative QoS Contracts, in correspondence to well-defined events. Such adaptation-triggering events coincide with the detection of changes in terminal- and/or network-resource availability (QoS Violations): e.g. the admission of a new flow, the termination of an old flow, radio propagation degradation or handovers. Alternatively, adaptations can also be triggered based on user commands (QoS Changes): e.g. changing user profile information and/or starting/stopping of media streams.

The information required for the automatic selection of alternative QoS Contracts is called Adaptation Path (AP), and is more specifically used for selecting, coordinating, and configuring multimedia components, as well as reconfiguring ESI-provided services. To efficiently react to QoS Violations and/or QoS Changes, BRENTA allows peers pro-actively negotiating AP before the actual media streams are established.

2.1.1 Hierarchical QoS Specification

Multi-media sessions may contain several streams of basic types (i.e. audio, video, and data). Users may thus wish to specify not only the desired level of QoS for each single stream, but also any parameter that might determine the inter-stream behavior (Multi-stream Time Synchronization): e.g. voice and video streams synchronization at each end-terminal.

Additionally, some level of correlation may be required between some or all of the aforementioned streams, on a more general basis (Multi-stream QoS Correlation). For instance, electronic game applications and/or media-rich interactive applications might feature bundles of audio and video streams, which are associated with objects to be presented to the user. For example, a rotating cube can be displayed on a monitor with its faces textured with images from different video streams; different audio streams, each associated with a cube face, can be played whenever the corresponding face is oriented to a certain direction.

We model multi-stream Time Synchronization and QoS Correlation constraints as high-level QoS Contracts themselves, associated with the list of the affected streams. Furthermore, it allows recursively bundling such high-level QoS Contracts among themselves, thus leading to a hierarchical QoS Specification scheme, i.e. equivalent to a tree. Each such leaf represents a stream and has a QoS Contract associated. Parent node are associated with a high-level QoS Contract, specifying for their children QoS levels in terms of multi-stream, Time Synchronization and QoS Correlation constraints.
Furthermore, users may prioritize and grant different amounts of resources to various (multimedia) applications. This is especially important for handheld devices with limited resources, like memory, battery-power. This approach leads to even higher-level Time Synchronization and QoS Correlation constraints, which are to be enforced locally by the terminal device. The corresponding high-level QoS Contracts extend the tree data model at the root. Such additional high-level QoS Contracts are however not meant to be negotiated with peers. Rather, each peer can enforce high-level QoS Contracts independently. Alternatively, high-level QoS Contracts can be enforced globally throughout a given closed set of peers, once a coordinator has been selected.

2.1.2 End-To-End QoS Negotiation Protocol

To cope with QoS Violations and/or QoS Changes, peers normally engage in QoS negotiations and renegotiations processes. More specifically, we name End-to-End QoS Full Negotiation the process that peers perform - either before or at the actual start of a session - in order to agree on a given QoS-level to be enforced for the given session and streams, eventually by redefining some of the originally proposed configurations of QoS specifications.

By End-to-End QoS Full Re-Negotiation we refer to the process that peers trigger upon detection of either QoS Changes or QoS Violations, in order to agree on a given QoS-level to be enforced for the given session and

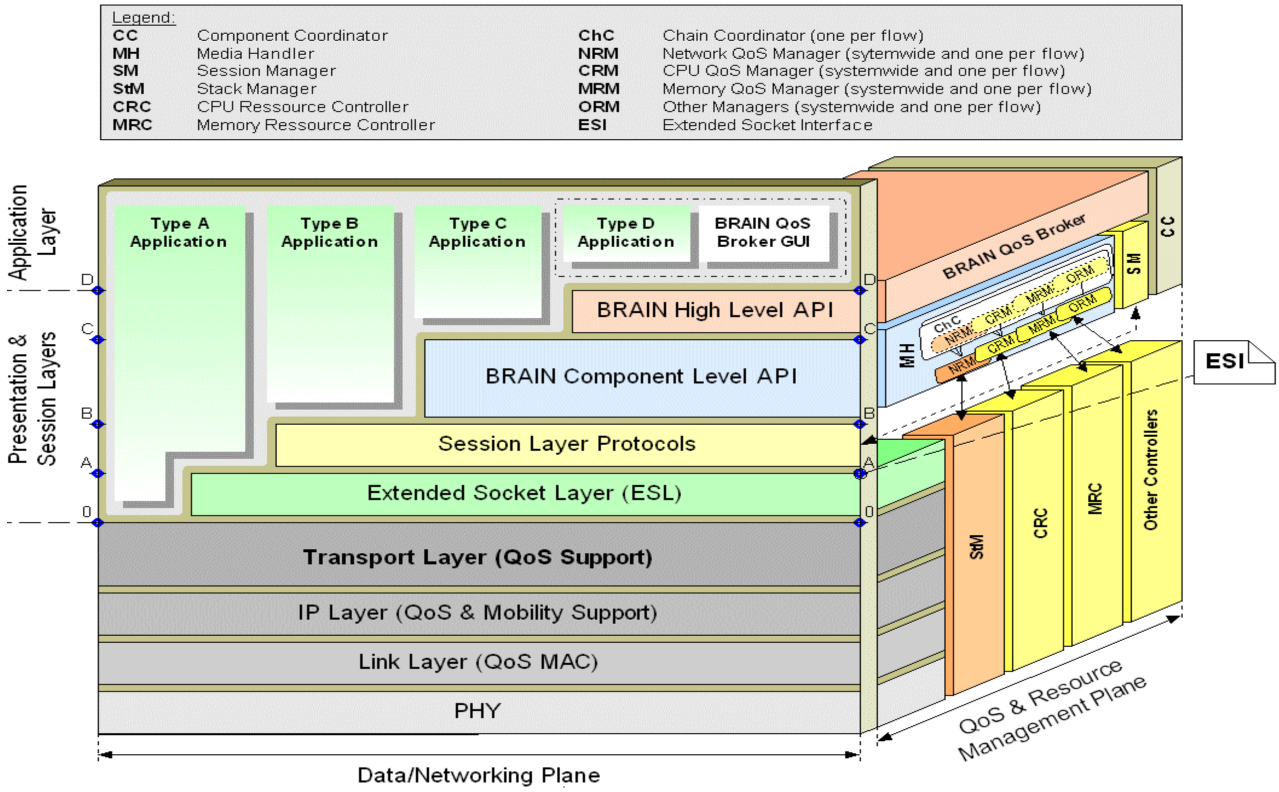

Figure 2: Architecture of the BRAIN End Terminal - BRENTA

streams, eventually by redefining some of the originally proposed configurations of QoS specifications.

As aforementioned, BRENTA allows peers pro-actively negotiating APs among themselves before the actual media streams are established, in order to effectively and efficiently react to QoS Violations and/or QoS Changes, whenever these occur. To this extent, we have developed the concept of End-To-End QoS Negotiation Protocol (E2ENP), an application-level protocol that Applications Type C or QoS Brokers can use for coordinating with peer ones. The E2ENP allows mobile peers to cope with unstable environment conditions, by enabling their applications to efficiently and timely react to QoS violations, by *planning counteractions ahead*.

The E2ENP allows peers negotiating off-line APs for each level of the aforementioned tree model, so that at runtime they can perform negotiation and renegotiations by simply exchanging QoS Contract identifiers. The E2ENP is based on a non-iterative negotiation process, whereby peers simply exchange among themselves information on a set of pre-negotiated APs.

Furthermore, the E2ENP includes a specific procedure for effectively enforcing QoS contracts, by applying the so-called “Economy Principle”. According to this principle, local and peers' resources are reserved before any network resource reservation is made. This is motivated by the fact that network resources are shared among a multiplicity of users, and thus are more expensive than terminal resources.

The E2ENP comprises four key phases (see Figure 3), which can be concatenated within the lifetime of a given session. Alternatively, the first two phases may be executed independently of the latter two and at different times, but strictly following the given order. More specifically, the phases are:

1. *End-to-End QoS Pre-Negotiation Phase*
   Peers perform this phase before the actual start of a communication session, and independently of the session itself, in order to reach an agreement on which APs peers should later enforce. Peers are thus able to establish a common vocabulary, a priori of any specific business. During this phase, the negotiation initiator (the offerer) proposes a bid to the negotiation responders (the answerers), which in turn reply with a counteroffer. This can only be a subset of the proposed bid.
2. *Multi-stream QoS Correlation and Time Synchronization Enforcement Phase*
   An optional phase concerning the establishment of dependencies among the multiple streams of a given multimedia session. A given peer applies this phase only if QoS correlation and/or time synchronization constraints are required. A central control entity (e.g. a conference call bridge) could also employ this phase, should the various peers agree upon delegating such entity to carry out complex negotiations.
3. *End-to-End QoS Compact Negotiation Phase*
   Peers can perform this phase either before or at the actual start of a session, in order to agree on which QoS-level to enforce for the given session and streams, based on results of previously applied phases. This Compact Negotiation is considerably quicker compared to the case of an End-to-End QoS Full Negotiation process, since only references of pre-negotiated information are actually exchanged among the peers. At completion of this process, peers have agreed upon the QoS Contracts they are going to enforce.
4. *End-to-End QoS Compact Re-Negotiation Phase*
   Peers can trigger this phase upon detection of QoS Violations or QoS Changes, so as to agree on a new

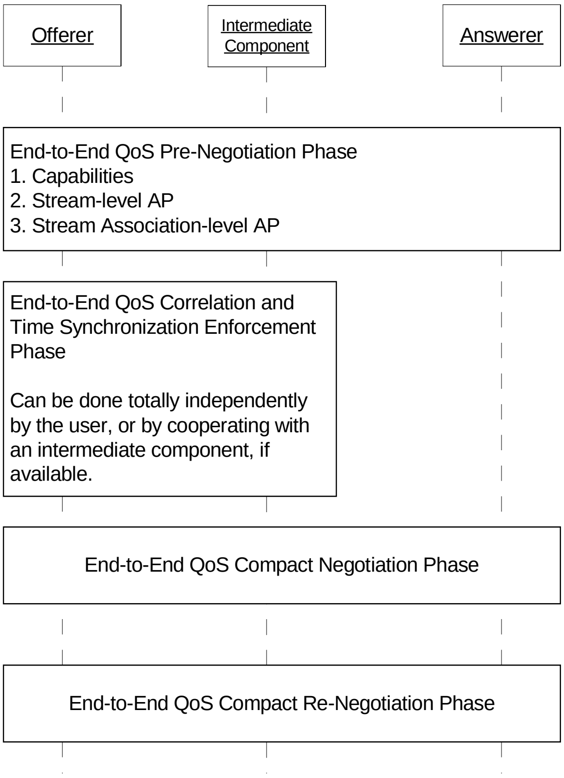

Figure 3: End-To-End QoS Negotiation Protocol

QoS-level to enforce for the given session, based on results of a previously applied End-to-End QoS Pre-Negotiation. This process is considerably faster compared to the case of the End-to-End QoS Full Re-Negotiation one, since only references of pre-negotiated information are actually exchanged among peers. This phase can be applied several times during the lifetime of any given session.

The E2ENP interacts with the resource management functions during all the four phases. More specifically, the E2ENP interacts with the local and network resource management functions during both the third and forth phase, according to the "Economy Principle". The rules for handling the joining/leaving of peers to a conference service are heavily dependent on the type of management applied to the given communication sessions. This can eventually involve external mechanisms and protocols, which are outside of the scope of the E2ENP.

The E2ENP is envisioned to be implemented as an application level protocol for both peer-to-peer and multiparty scenarios, by considering existing models [3]. We are currently studying E2ENP as an extension of SDPng [4], piggybacked on PDUs of existing session protocols, like SIP [5].

2.2 BRENTA Details

BRENTA is split into two major planes: the usual data networking plane, and a QoS- and resource-management one [6]. The former deals with data handling through the use of multimedia components (e.g. codecs) and ESI calls. The latter deals with the management of the data networking plane - including the coordination of local, network, and peer's resource management - in order to achieve given QoS levels.

2.2.1 Resource Management

In the QoS- and resource-management plane, BRENTA features the following mechanisms.

- *Management of local resources* (like CPU, memory, battery power…)*:* centralized *Resource Controllers* (RC - one per resource type) monitor and control resources on a system-wide basis. Sets of *Resource Managers* (RM) provide resource management on a per-flow basis, by interacting with the corresponding RC through a RC-specific API. An example is the CPU RC (CRC) - which allows monitoring the task scheduler, reserving CPU time-slices, and/or setting deadlines for a given task - along with CPU RMs (CRM).
- *Network resource management* provides a guaranteed amount of network resources (like bandwidth) to mission critical applications, through the ESI. A Stack Manager (StM) can *also* be used for accessing link level resource availability information in order to identify when QoS can be upgraded. Being modeled according to the RC/RM design pattern, the StM is associated with one or multiple Network Resource Managers (NRM).
- *Remote resource management* guarantees that also the called parties can provide the desired QoS level, before reserving network resources (which, being shared by a huge number of peers, are *expensive*).
- *QoS orchestration* mechanisms *co-ordinate* resource reservation and management tasks, in order to provide predictable end-to-end QoS, even in front of mutated situations. Applications type C or the QoS Broker enforce these mechanisms at a high-level, whereas lower-level QoS adaptation tasks are delegated to the Media Handler component (MH), which in turn spawns and supervises instances of the Chain Coordinator component (ChC). Once provided with low-level APs derived from information provided by the QoS Broker (or by applications type C), each ChC accordingly combines multimedia components and ESI services *on a per-flow basis*, and configures and manages them so as to enforce the selected adaptation policies (using RM services). Peers can coordinate their QoS orchestration efforts by using the E2ENP.

2.2.2 User Perceived QoS Management

The QoS Broker provides controlled and predictable QoS adaptation means for multimedia services, based on high-level QoS Contracts and well-defined strategies for each subscribed service. To this extent, the QoS Broker manages sets of active flows by (i) deciding the distribution of resources among them, (ii) resolving any conflicts between flows whenever required resources are lacking, and (iii) handling priorities in order to provide the optimal repartition according to the user privilege and application QoS. The data management of user profile and system configuration information is handled through the use of (respectively) User Profile RM/RC and Policy RM/RC.

As aforementioned, BRENTA adaptation mechanisms operate on the basis of pre-negotiated APs, modeled as hierarchical Finite State Machine (H-FSM) [7]. In this model, each QoS Contract is mapped to a state of a H-FSM. At the lowest level of this hierarchical structure, the FSM states correspond to the QoS

Contracts of the individual streams. The nominal QoS Contract (i.e. the one, which the user wishes to enable by default) corresponds to the initial state of the FSM associated with the given AP. Each AP corresponds to an elemental FSM, whereby states are mutually exclusive. States and/or complete elemental FSMs can be nested within higher-level states, which in turn are associated with high-level QoS Contracts. Each transition of such hierarchical FSM describes a pre-defined change of QoS Contract in response to a given adaptation-triggering event (QoS Violations and/or QoS Changes).

Inputs to these FSMs are ESI primitives and/or the output of the QoS assessment function, a sub-component of the QoS Broker. This entity derives estimates [8] of the actual high-level QoS from QoS measurement information obtained from (hierarchically structured) QoS monitors (e.g. RTCP reports). This function then compares such estimates against the states of the FSM representing the given APs, checking whether the currently enforced QoS Contracts (i.e. the currently active states) comply with the estimated QoS Level, and to what extent. A compatibility factor and a time window are used for tuning the sensitivity of this algorithm. The goal is to determine whether a new QoS Contract (out of the given pre-negotiated AP), which better fits the estimated QoS level. Control techniques (e.g. the use of fuzzy logic [8]) can be employed to dynamically select at runtime a new QoS Contract to enforce. Alternatively, a static approach is to rely on FSM state transitions, triggered by specific ESI primitives and/or by the results of the QoS Assessment function. Once a new QoS Contract has been identified, the QoS Broker delegates any low-level QoS adaptation task to ChCs.

3. QoS aspects in ad hoc Networks

Part of the research driven by the MIND Project focuses on extended IP-based radio access networks with QoS-aware ad hoc networks enhanced to support End-to-End QoS. Ad hoc networks consist of a collection of mobile wireless nodes that dynamically create a network among them without using any infrastructure or administrative support, i.e., the necessary control and administration functions on such networks are only accomplished by the interactions among their constituent nodes [9]. In this section we *identify* and *analyze* which are the main QoS issues introduced in MIND because of the use of ad hoc networks and propose a set of requirements to prepare BRENTA for these environments.

3.1 QoS issues in ad hoc networks

In such networks, the provision of QoS is extremely difficult due to the error-prone nature of the wireless links, the unpredictability of the network topology and the limited power autonomy of each node. The establishment of connections with stable QoS parameters (delay, required bandwidth, allowed packet loss, etc) is challenging at different layers of the protocol stack and components of the terminal architecture.

- At physical layer, it’s necessary to use retransmission schemes to prevent packet losses. This choice depends on the nature of packets being retransmitted and the energy consumption policy.
- The lack of fixed infrastructure means that there is no dedicated entity to manage the channel resources for the network, then carefully designed *distributed medium access techniques* must be used for management of channel resources avoiding the disproportionate usage of bandwidth by some nodes and providing mechanisms for fast recovery from packet collisions.
- The mobility of each node makes necessary to employ *very effective routing protocols* different from the existing in the fixed networks. In ad hoc networks, the amount and frequency of the information exchanged between the nodes in each topology update may produce a considerable *overhead* that reduces the resources available in the whole network. On the other hand, the search of a path just when is required attempting to preserve the available bandwidth may increase the *route acquisition time* and the end-to-end delay for the QoS provisioning [10].

  After finding a route (using some ad hoc routing protocols) is necessary to implement a resource reservation mechanism. Traditional signaling mechanisms like RSVP are not suitable to these environments as it may happen that while I'm establishing a QoS reservation through a path, the path become no longer available. In addition these mechanisms introduce delay and overhead affecting the performance of the QoS support.

  In this sense, is useful the usage of a QoS Routing process, i.e. the process of finding a suitable path to be used by a flow with requested QoS guarantees, but the route computation with this strategy may take too long or be complex [11]. Another routing method to reduce QoS violations in ad hoc networks is the simultaneous usage of alternatives routes, especially if these routes are disjoint to increase the throughput of a flow. In general, routing in ad hoc networks consume network bandwidth, CPU capacity and other local resources and increase the delay jitter experienced that must be kept under a bound to satisfy the real-time traffic.

  Finally, due to node mobility, the global state information exchanged during the *topology update* may never be accurate. Some degree of network *topology* stability is required to allow the ad hoc routing protocols to distribute it.
- At higher levels of the stack, new *transport protocols* are necessary (or modification have to be made) to react properly and detect packet losses caused by transmission errors as well as the node mobility rather than congestion, that is the predominant problem in fixed networks [12]. Additionally, during a session negotiation the applications can agree to use some special encoding techniques to improve the packet loss recovery like Priority Encoding Transmission and Forward Error Correction.

  Furthermore, several administrative functionalities like address allocation and name resolution are needed to support applications in absence of a pre-established or centralized management infrastructure. Then, in isolated ad hoc networks, the nodes must cooperate to provide these services in a decentralized way taking into account the frequent changes of the network population and service availability [13]. The focus of MIND is specifically not such isolated ad hoc networks, but ad hoc networks which are incorporated to a fixed infrastructure.

3.2 Roles of a BRAIN Terminal in ad hoc networks and power consumption issues

In ad hoc networks, each node operates as (i) a terminal and (ii) router, both at the same time. Then, energy expense of the mobile node powered by batteries is an important issue that affects the topology of the network too [14]. So, it's convenient to use the shortest path between a source-destination pair to minimize the overall power consumption during the routing process or use an alternative route in which intermediate nodes have enough power to expend supporting this activity for enough time. Sometimes, the latter is difficult or impossible when just only one node acts as (iii) a gateway between the ad hoc network and a fixed radio access network. In MIND project, this gateway allows the exchange of packets between the Internet and the ad hoc network, the use of external administrative functionalities (e.g. DNS) and the seamless handoff of visiting mobile nodes. So, it could be also convenient to include (iv) foreign agent functionalities in a node to facilitate it. Taking these considerations into account, the gateway could become a bottleneck or a single point of failure without sufficient local resources. Considering all these constrains, it is important to extend the power economy requirement to other layers of the protocol stack to guarantee End-to-End QoS provision. .

3.3 Requirements for QoS support in ad hoc networks with BRENTA

BRENTA (Fig. 2) requires some additions to fulfill the End-to-End QoS provisioning in mobile networks extended with ad hoc networks. These extensions are necessaries to facilitate:

1. The means for the applications to request QoS for their flows in ad hoc environments.
2. The forwarding of QoS traffic generated in other nodes through suitable paths.
3. The reservation of the preferable QoS paths with the coordination with task #4.
4. The management of local resources taking into account the amount of resources required for external traffic.
5. The management of multiple protocol stacks for seamless vertical handovers to/from ad hoc networks.

Therefore, to accomplish the first two tasks, is necessary the usage of ESI primitives with QoS related information derived from the knowledge about all the available resources collected by the QoS Broker and routing protocols. Then, the applications could request QoS support by means of the ESI primitives in transparent and efficient manner depending of the transport/QoS Service Provider (QoSSP) implementation (like RSVP, DiffServ or a suitable one for ad hoc networks). A QoSSP in BRENTA is in charge of configuring the proper local network stack.

QoS routing functionalities could be necessary to maintain QoS information of the network topology. It is preferable to use a *distributed QoS routing protocol* in which the path calculation is shared by various nodes avoiding computation burden on a single node. Once the route (or disjoint routes) is established, a reservation (task #3) could be developed by an efficient signaling system that minimizes the overhead and delay of path establishment [15].

In the management of local resources the Resource Controllers must be aware of the resources granted to another's traffic (e.g. memory, CPU, bandwidth) to report more precise information to per flow Resource Managers. Then, a kind of "usage admission control" algorithm must be implemented for accepting a request for resource reservation (for local and foreign flows) evaluating e.g. the impact on the CPU, memory, power and bandwidth. In this sense, taking into account the possible vertical handovers to/from ad hoc networks too (task #5), BRENTA not only requires the extensions of protocol sets and functionalities at the lowers levels, it requires also some extensions to the Stack Manager (StM) for the management of local resources that allows QoS related configuration and information necessary for local traffic control and supporting vertical handoff.

4. Security Aspects

Security concerns can be raised when envisioning the usage of mobile end-systems and IP based wireless access networks in the transmission path. These range from location privacy deficiencies up to application level specific security models, which may ensure trusted business transactions. In each of these problem domains specific trust and confidence requirements exist. The scope of just started security related activities in MIND, however, is concentrating on

Confidentiality

- users don't want their personal/payment details transported insecurely,
- network operators want to identify customers taking responsibility for the costs raised once they allow access to the network;

Integrity

- users don't want someone else using their account,
- network operator want a reliable relations to a user once having granted access;

Accountability

- the system doesn't want users denying transactions undertaken,
- users want network operator to provide the service having been promised.

The aim is to eventually charge and bill for the usage of resources in mobile networks. The necessary prerequisite is here access control so that authorized usage of resources is ensured. Considering security within this scope is addressed along the line of some identified scenarios. They describe typical situations in which mobile IP is in particular meaningful for network operators, and which encompass specific usage and accounting patterns. The scenarios are the onset to identify the vulnerabilities and threads that can be encountered.

During the course of this thread analysis, we will identify a set of security policies. The policies reflect the various interests of the parties involved in the scenarios. It can be assumed that of applications providers, network operators, users and customers, having authorized users, can express divergent interests in the scenarios. Once these are better understood and the security policies have been selected that are to be supported in MIND, potential solutions to increase the confidence in the accounting will be addressed. Here it is foreseen to preferably select existing basic security mechanisms, which, on an engineering level, can support the selected security policies thus raising the confidence of the involved parties in the correct and accurate accounting when using IP based wireless access network or the ad hoc networks.

5. Conclusions

In the previous sections, remarkable aspects of BRENTA were discussed. Key concepts of BRENTA, such as modularity, openness and flexibility as well as the Enhanced Service Interface (ESI) have been presented. The induction of these concepts allows, as a key factor within the BRAIN framework, the deployment of a suitable set of technology-independent applications, and provides an architecture open for future improvements.

The importance of this QoS architecture for deploying multimedia services and applications over Wireless IP networks has been highlighted. Services and applications are envisioned to progressively delegate QoS and mobility management tasks to middleware: BRENTA encompasses applications ranging from legacy ones (type A) to the future ones, fully relying on middleware (type D). To this extent, BRENTA design is split in two major planes: the usual data networking plane, and the QoS- and resource-management one. The former deals with data handling through the use of multimedia components (e.g. codecs) and ESI calls, whilst in the latter BRENTA manages and coordinates local and network resources.

An overview of ongoing work within the MIND project related to QoS issues has been presented. The End-to End QoS Negotiation Protocol (E2ENP) has been introduced as a mechanism to orchestrate adaptation, QoS negotiation and resource management in distributed scenarios, where multi-stream, multi-session QoS correlation shall be enforced. Currently, we are studying how to implement the E2ENP using SIP together with SDPng extensions. QoS problems that arise in ad hoc networks have been introduced too. Due to special wireless characteristics, many remarkable issues are to be solved at different layers. BRENTA extensions are thus proposed to this extent. In the last section the current work concerning security issues in MIND is revised. The scope of this work is concentrating on Confidentiality, Integrity and Accountability.

Acknowledgements

This work has been performed in the framework of the IST project IST-2000-28584 MIND, which is partly funded by the European Union. The authors would like to acknowledge the contributions of their colleagues from Siemens AG, British Telecommunications PLC, Agora Systems S.A., Ericsson Radio Systems AB, France Télécom S.A., King's College London, Nokia Corporation, NTT DoCoMo Inc, Sony International (Europe) GmbH, T-Systems Nova GmbH, University of Madrid, and Infineon Technologies AG.

References

[1] T. Robles, A. Kadelka, H. Velayos, A. Lappetelainen, A. Kassler, H. Li, D. Mandato, J. Ojala, B. Wegmann, *QoS support for an all-IP system beyond 3G*, in: IEEE Communications Magazine, pp. 64-72, August 2001.

[2] A. Kassler et.al.: *BRENTA - Supporting Mobility and Quality of Service for Adaptable Multimedia Communication*, in Proceedings of IST Mobile Summit 2000, Galway, Ireland, October 2000

[3] J. Rosenberg, H. Schulzrinne, *Models for Multi Party Conferencing in SIP*, IETF SIPPING working group, Work-in-progress, July 2001

[4] D. Kutscher et. al. *Session Description and Capability Negotiation*, work in progress, draft-ietf-mmusic-sdpng-03.txt, November 2001

[5] Handley et. al.: SIP: *Session Initiation Protocol*, IETF SIP working group, Work-in-progress, July 2001

[6] T. P. Barzillai et al., *Design and Implementation of an RSVP-Based Quality of Service Architecture for an Integrated Services Internet*, IEEE Journal on Selected Areas of Communications, Vol. 16, No. 3, April 1998

[7] G. Booch, J. Rumbaugh, I. Jacobson, *The Unified Modeling Language User Guide*, Addison Wesley Longman, 1999.

[8] Salem N. Bhatti, Graham Knight, *Enabling QoS adaptation decisions for Internet applications*, London, UK, 1999.

[9] C. Perkins (Editor). *Ad Hoc Networking*. Addison-Wesley. USA. 2001

[10] L. Feeney. *A Taxonomy for Routing Protocols*, in Mobile Ad Hoc Networks. SICS Technical Report T99/07. Swedish Institute of Computer Science. Octuber, 1999

[11] Shigang Chen. *Routing Support for Providing Guaranteed End-to-End Quality-of-Service*. PH. D. Thesis. University of Illinois at Urbana-Champaign. USA. 1999.

[12] Kartik Chandran et al. *A Feedback-Based Scheme for Improving TCP Performance in Ad Hoc Wireless Networks*. IEEE Personal Communications. February 2001.

[13] L. Feeney, B. Ahlgren, A. Westerlund. *Spontaneous Networking: an application-oriented approach to Ad Hoc Networking*. IEEE Communications Magazine. June 2001

[14] C.-K. Toh. *Maximun Battery Life Routing to Support Ubiquitos Mobile Computing in Wireless Ad Hoc Networks*. IEEE Communications Magazine. June 2001

[15] S.-B. Lee, G.-S. Ahn and A. Campbell. *Improving UDP and TCP performance in Mobile Ad Hoc Networks with INSIGNIA*. IEEE Communications Magazine. June 2001