| | |
|---|---|
| **Project Number:** | **IST-1999-10077** |
| **Project Title:** | **AQUILA**<br>**Adaptive Resource Control for QoS Using an IP-based Layered Architecture** |
| **Deliverable Type:** | **PU - public** |

| | |
|---|---|
| **Deliverable Number:** | **IST-1999-10077-WP0-SAG-002-PU-R/b0** |
| **Contractual Date of Delivery to the CEC:** | **April 30, 2000** |
| **Actual Date of Delivery to the CEC:** | **April 30, 2000** |
| **Title of Deliverable:** | **Project Presentation** |
| **Workpackage contributing to the Deliverable:** | **WP 0** |
| **Nature of the Deliverable:** | **R – Report** |
| **Editor:** | **Bert F. Koch (SAG)** |
| **Author(s):** | **Bert F. Koch et al.** |


| | |
|---|---|
| **Abstract:** | This deliverable D002 gives an overview on the objectives and the technical approach of the AQUILA project.<br><br>Additional information can be found at the project's home page at: http://www-st.inf.tu-dresden.de/aquila/ |
| **Keyword List:** | AQUILA, IST, QoS, Internet, Resource Control |

# Executive Summary

This deliverable includes an overall description of the AQUILA project. It gives an overview on the objectives and the technical background. The document also contains the project's technical approach and a description of the various workpackages.

# Table of Contents

# 1 Project Summary

AQUILA defines, evaluates, and implements an enhanced architecture for QoS in the Internet. Existing approaches e.g. Differentiated Services, Integrated Services and label switching technologies will be exploited and significantly enhanced, contributing to international standardisation. The architecture will be designed to be cost-effective and scalable. It introduces a software layer for distributed and adaptive resource control and facilitates migration from existing networks and end-user applications. Technical solutions will be verified by testbed experiments and user trials, including QoS-enhanced on-line multimedia services. Business plans for further exploitation will be studied.

See also the project's home page at: http://www-st.inf.tu-dresden.de/aquila/

# 2 Project Objectives and Background

## 2.1 Project Objectives

In order to satisfy the huge commercial demand for Quality of Service (QoS) solutions over IP networks, the project AQUILA defines, evaluates, and implements an enhanced architecture for Quality of Service. Existing approaches for QoS provisioning in the Internet e.g. Differentiated Services (DiffServ), Integrated Services (IntServ) and label switching technologies (e.g. MPLS) are used as basis for the specification of this architecture and will be significantly enhanced and exploited.

The achieved technical solutions will be verified by testbed experiments and by trials involving end-users. The trials will include QoS demanding on-line services, e.g. multimedia services. The straight exploitation of the results will be achieved by the development of business models and business plans.

**The key objectives of the project are:**

1. To enable ***dynamic end to end QoS*** provisioning in IP networks for QoS sensitive applications e.g. Internet telephony, premium web surfing and video streaming. Static resource assignments will be considered as well as dynamic resource control. The latter can take into account the actual load situations in the IP network and can adapt the network to dynamic load changes.

2. To continuously analyse ***market situations*** and ***technological trends*** for QoS solutions and to exploit the results of the project creating applicable business plans based on the user and service provider requirements.

3. To design a ***QoS architecture*** including an ***extra layer for resource control*** for scalable QoS control and to facilitate migration from existing networks. The Differentiated Services architecture for IP networks will be enhanced introducing dynamic resource and admission control. The main features of this architecture are as follows:

- The architecture will be usable by any relevant kind of IP application; i.e. to provide several options for the establishment of QoS requests by user applications, e.g. via CORBA, RSVP or HTTP.

- The architecture will be cost-effective, scalable and backward compatible for the provisioning of QoS in IP networks covering both the inter- and intra-domain QoS.

- The architecture will consider the requirements for QoS accounting.

- The architecture will be kept open and flexible so that the project can incorporate new concepts and knowledge from other research projects (in particular the European Quantum project, the Internet-2 and the Qbone initiative) and from standards bodies (e.g. IETF, OMG).

4. To ***implement prototypes*** of the QoS architecture as well as QoS based end-user services and tools in order to validate the technical approach of the solution design. This includes:

- To develop a novel resource control layer extending Bandwidth Broker functionality.
- To provide an End-user Application Toolkit (EAToolkit) in order to support the establishment of QoS by end-users and applications.
- To create tools for QoS provisioning, monitoring and management in order to facilitate operators to control QoS IP networks.
- To develop a distributed measurement infrastructure for end-to-end QoS parameters. The results are a basis for optimisation of protocol parameters and QoS network management.

5. To validate the QoS architecture in a ***field trial*** involving a commercial online service. To prove the concepts for larger scale networks, higher network load and different kinds of end-user services within a distributed ***testlab*** and by ***simulations***. To examine the commercial acceptance of the offered QoS, for both business and residential users.

6. To enable ***migration*** to QoS-enabled networks including ***deployment*** aspects. The project will support incremental transition from best effort to differentiated QoS. The change will involve both the technology and the offered services. The project will evaluate different deployment scenarios and define a migration strategy for the operators and service/content providers.

7. To ***contribute to standardisation*** bodies like IETF, ITU, ETSI, OMG etc. This includes regular attendance of key personnel at the standardisation meetings and active contributions.

In order to achieve these objectives, the AQUILA project joins the strengths of a consortium of partners covering all the appropriate competence areas: multimedia content and services providers, network operators and Internet service providers, network technology manufacturers for routers, remote access servers and access technology, and research institutes with experiences in router technology, IP test and measurement tools, network simulation and software technology. This comprehensive combination enables the AQUILA consortium to design, implement and practically demonstrate novel prototypes for future end-to-end IP-Quality of Service solutions to technological, service provision, operational and economical aspects.

## 2.2 Project Consortium

The AQUILA consortium consists of 12 partners from 6 European countries, including 1 manufacturer, 5 Internet Service Providers and Network Operators, 5 Universities and research institutes and 1 Web application provider.

The list of project partners is as follows:

Siemens AG, Germany, co-ordinating partners,

National Technical University of Athens, Greece,

Bertelsmann Media Systems, Germany,

Helsinki Telephone Corporation (Elisa Communications), Finland,

Technische Universität Dresden (Dresden University of Technology), Germany,

Consorzio di Ricerca sulle Telecomunicazioni (CoRiTeL), Italy,

Techno-Z FH Forschung und Entwicklung GmbH (Salzburg Polytechnic University), Austria,

Q Systems Association, Greece,

T-Nova Deutsche Telekom Innovationsgesellschaft mbH, Germany,

Telekom Austria AG, Austria,

Telekomunikacja Polska S.A. (Polish Telecom), Poland,

Politechnika Warszawska (Warsaw University of Technology), Poland.

## 2.3 Project Details

The project is funded by the IST Programme of the European Commission within the Fifth Framework Programme and has been launched during the Programme's first call with the project number IST-1999-10077.

The total cost of the project is planned to be at 11.531.175 Euro, the Commission's funding is at 5.998.457 Euro.

The duration of the project is 36 months, starting at January 1st, 2000.

The Project Management is being performed by the co-ordinating partner, Siemens AG.

Contact address is the Project Manager, Bert F. Koch, at "Berthold.Koch@icn.siemens.de".

## 2.4 Technical Summary

The dramatic growth of the Internet in size and popularity makes it a safe prediction that the next generation of large-scale integrated services network technology will be based on Internet concepts. One of the main technical reasons for this success is that the Internet Protocol (IP) introduces a level of abstraction, which allows seamless interconnection of heterogeneous subnetworks. These may use different transport technologies (e.g. ATM, SDH, WDM) and may be managed by different organisations. However, this strength has also led to a serious problem: current Internet technology is neither suitable for good-quality voice transmission nor for the requirements of advanced multimedia telecommunication services, yet. Furthermore, extension or replacement of the basic protocol (e.g. using IPv6) is very difficult to achieve in practice.

The well-known challenge for this project is to provide reliable Quality of Service (QoS) by using current IP technologies. The new architecture to be developed by this project shall provide a similar degree of interoperability for heterogeneous networks as current IP, but on the higher layer of resource control for IP networks. The project will take up and further develop some promising new concepts that came up recently (and very rapidly) and were applied in the US prototype networks for “Internet2”. Europe currently needs more activities towards future IP networks. The proposed project aims at providing a significant European research contribution to this area. The proposed project will contribute in three different aspects. *Technically*, the project will focus on a highly scalable architecture that can be introduced by relatively small migration steps on top of existing infrastructure, using results from other European research projects. *Economically*, the project focuses on commercial usage of the network by high numbers of end-users (mainly private persons and small or medium enterprises) in contrast to the US-approach of an academic research network. *Methodically*, the project faces the problem of aiming at a rapidly moving target, and therefore provides mechanisms for dynamic fine-adjustment of some project objectives during its execution.

So the general motivation for the project proposal can be seen as a genuinely European contribution to the development of an Internet of the next generation.

## 2.5 State of the art

The current Internet architecture is not designed to support QoS per se and there exist different approaches for providing QoS over IP-based networks. Due to the different underlying mechanisms of these approaches and the complexity of end-to-end QoS, there is currently no solution suitable for global operation. In particular, management of these QoS mechanisms and the provisioning of inter-domain QoS are open issues. The most important approaches are the following:

- *Optimised IP traffic over ATM (e.g. MPOA, IP Switching).* These approaches provide an alternative way of establishing switched virtual circuits over an ATM network for IP traffic. It is very difficult to associate the resource usage in the ATM network with QoS demands of individual users. This causes problems, e.g. in creating the revenue for the cost of the ATM network. Moreover, this approach relies on ATM networks, which deviates from the idea of inter-networking of heterogeneous networks.

- *Integrated Services (IntServ)*. The basic idea of this Internet Engineering Task Force (IETF) approach is the use of additional components within routers and a signalling protocol (e.g. the resource reservation protocol RSVP) for requesting a specific QoS from the network on behalf of an application data stream. However, there is severe doubt in the Internet community whether this approach with the granularity of a single flow scales well into a reliable wide-area network. Many important issues are unsolved, in particular appropriate charging and admission control mechanisms in order to make an integrated services architecture economically viable.

- *Differentiated Services (DiffServ)*. This approach was also developed by the IETF and it is the most recent one in the area of QoS provisioning over IP networks. DiffServ is strictly a Layer 3 proposition and makes no assumptions about the underlying transport. It is planned to be used extensively in the Internet2 initiative in the US. The DiffServ architecture focuses on the per-hop behaviour of classes of aggregated traffic and is therefore scalable. The basic idea is the separation of the operations performed in the borders of the network from those accomplished in the core network. Border routers perform more complex tasks such as traffic policing and shaping, marking, and prioritised routing. The only remaining task of core routers is the forwarding process, based on the assigned per-hop behaviour.

- *Multi Protocol Label Switching (MPLS)*. MPLS is also an IETF standardisation activity and it has its roots in several proprietary contributions from different vendors (e.g. CISCO, Ascend, IBM). MPLS specifies ways that Layer 3 traffic can be mapped to connection-oriented Layer 2 transports like ATM and Frame Relay. The main idea is to add a label containing specific routing information to each IP packet and to allow routers to assign explicit paths to various classes of traffic. It also offers capabilities not directly related to QoS like traffic engineering techniques that can boost IP routing efficiency.

- *QoS Routing*. While currently used IP routing protocols are insensitive to QoS needs, QoS routing enables the path selection based on metrics, which consider the availability of resources. This may result in different forwarding decisions in comparison with those, which are based on the traditional “shortest path” metric usually adopted in the Internet. The IETF QoS routing working group is focussing on how to select and maintain packet-forwarding paths capable of meeting specific service class objectives. The provision of QoS routing information in edge devices or routers can extend all mentioned QoS mechanisms for an utilisation of the available network resources. For example the combination of QoS-based routing can be used within MPLS-aware network domains to avoid congestion by setting alternate routes (explicit routes) to the destination.

- *Bandwidth over-provisioning*. Recently, a rather radical approach towards QoS is being discussed in the Internet community. This approach is based on the prediction that a drastic decline in transmission cost will remove the QoS problem in the long run, since the network can be massively over-dimensioned at reasonable cost. However, this seems a rather limited perspective, and in particular in a world-wide network infrastructure it is likely that some portions of the network will be overloaded for sometime or even permanently. The DiffServ approach is able to cope well with such a situation since it causes no overhead on network portions, which are over-dimensioned.

Most of the above mentioned technical solutions on how to bring QoS into IP networks are still under discussion. Some of them are divergent, while some are complementary. No integrated scaleable solutions are available right now. Furthermore, management and interoperability aspects of the mentioned approaches are currently treated poorly.

There is a strong need to make new technical features accessible to users in a customer friendly manner. Technical details should be hidden to the user. This is of growing importance since more and more new applications will include QoS sensitive components. For the wide range of traditional applications, we need customer-friendly ways to use them properly with the new network QoS interfaces.

# 3 Technical Project Content

## 3.1 Technical Approach

The technical approach taken by the proposed project integrates all relevant aspects for future integrated services networks based on the Internet. Therefore it covers economical aspects, technological concepts, aspects of network planning and analysis, aspects of network service creation and administration as well as aspects of end-user service provision. For all these aspects, the project applies a common philosophy of taking the best available concepts and technologies, integrating them into a holistic view of future telecommunication services and providing tools for migration from the current to the future situation.

In terms of networking technology, the project assumes the DiffServ architecture as the most promising starting point for its work. The project develops extensions of this architecture in order to avoid the statically fixed pre-allocation of resources to users. Dynamic adaptation of resource allocation to user requests shall be enabled in a way that keeps the overall architecture scalable to very large networks. As an example for an alternative approach, the RSVP protocol defined in the Integrated Services architecture should be present in the access network where the load is small and the scalability issue is not important. Moreover, innovative techniques should be utilised wherever they appear in the network infrastructure.

For end-user applications, the project attempts to make its infrastructure available to a large class of already existing Internet applications. So the project will not attempt to develop end-user applications “from the scratch”, but will instead develop a toolkit, which can be used to make existing applications aware of the QoS capabilities of the underlying network. The usability of the toolkit for migration and adaptation of applications will be demonstrated by developing applications to be used in the trial phases.

### *3.1.1 Overall Description*

Figure 1 illustrates a schematic QoS scenario of the project. End-users as well as content providers are connected via Edge Devices (ED), which control the access to the core network. This consists of multiple independent IP domains, which might be maintained by different operators. Resource Control Agents form an overlay network which monitors and controls the resources of the core DiffServ network. End-users initiate new IP flows with specific QoS requirements by using the so-called EAToolkit. The EAToolkit does not constitute a new signalling protocol for IP networks. Instead, it can be described as a QoS middleware that brings the functionality of the Resource Control Agents Network into the end-user terminals and servers. The internal signalling protocol used between user terminals and the main network can be based on existing schemes (e.g. RSVP) or even on CORBA or DCOM interfaces depending on application needs. In any case, the ED analyses the user request and executes the user policy control and the local admission control operations in order to determine whether the specific user has the administrative rights and whether there are enough internal resources for the handling of the particular request. However, end-to-end guarantees can be provided only if the level of available resources of all intermediate routers to the final destination is known. Therefore, the ED uses its interface with the Resource Control Agents Network for performing the network admission control operation.

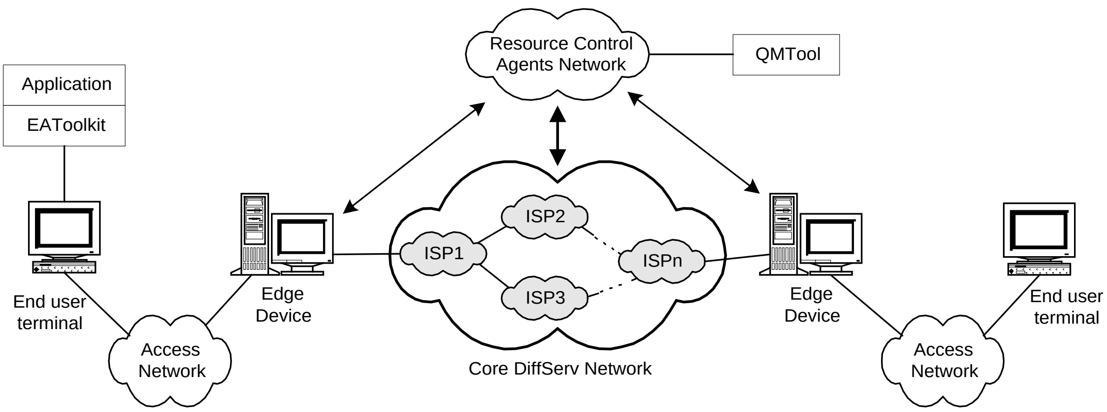



***Figure 1: AQUILA QoS Scenario***

### 3.1.2 Resource Control Agents

The support of QoS, as proposed above, leads to the introduction of a new logical layer, the *Resource Control Layer* (see Figure 2). The Resource Control Layer provides an abstraction of the underlying layers. A node in the Resource Control Layer is called a *Resource Control Agent (RCA)* and represents a portion of the IP network, which has internally the same QoS control mechanisms. An RCA is a generalisation of the concept of the Bandwidth Broker in the DiffServ architecture. RCAs are logical units that run on several physical configurations, e.g. one server per RCA or several RCAs co-located on one server. The QoS control mechanisms used in the underlying network are of varying nature, e.g. in some part the routers may not even support DiffServ (which means that there is only a trivial best-effort QoS control), while in other parts they may be DiffServ capable. Moreover, some parts of the network may allow dynamic reconfiguration of resources, e.g. by adding ATM connections, others may have a more or less fixed configuration, e.g. pure SDH or WDM subnetworks. Another reason for the introduction of separate RCAs is that subnetworks are domains managed by different operators. The following figure depicts the three-layered approach:

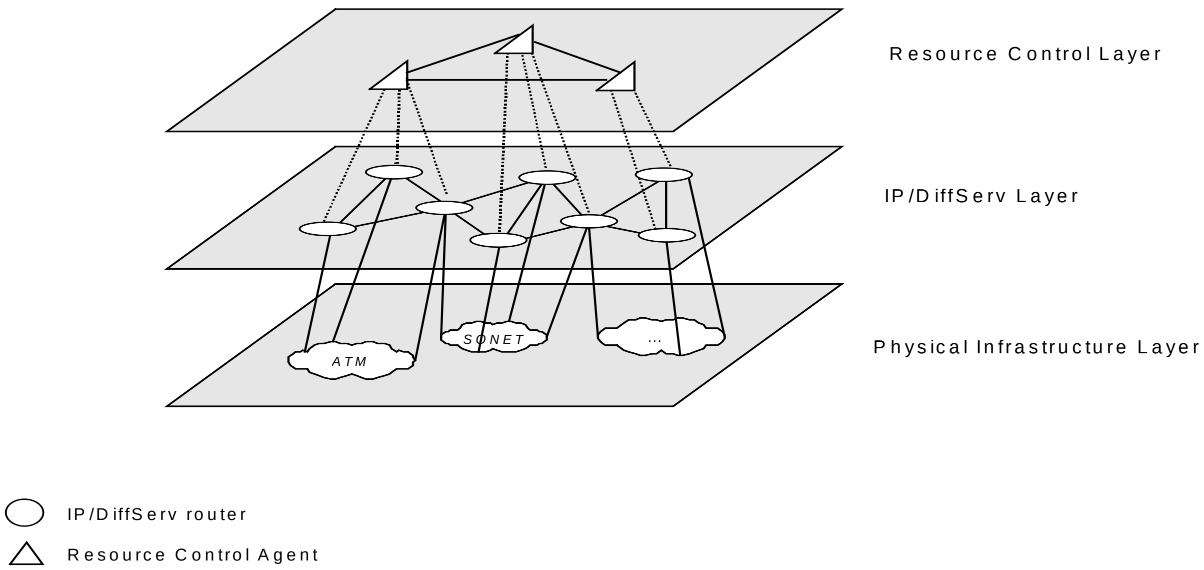



***Figure 2: Layered architecture of AQUILA***

A Resource Control Agent is able to observe and in some sense to influence the actual configuration in the network portion it represents. Configuration parameters may describe the fraction of a network connection devoted to a specific DiffServ traffic class or the existence of a virtual connection (in ATM networks) with a specified bandwidth. For this administration purpose, a mechanism is required to access IP routers and possibly other network elements.

The RCAs employ distributed computation that adapts the network according to user requests for QoS. The RCAs always try to establish a situation where the network configuration is "slightly over-dimensioned" such that user requests can be immediately satisfied by just checking the admissibility of the user and recording the additional resource usage. As soon as some "watermarks" are reached, the RCAs start a dynamic reconfiguration process in order to avoid congestion. The adaptation algorithm of an RCA is the reason why it is called an "agent" (even an intelligent agent) since the RCA can act autonomously, in contrast to admission control which is often non-local. For the purposes of a first prototype, locally fixed agents (which communicate over CORBA) seem to be sufficient. However, it is basically feasible to make use also of mobile agents here e.g. to move the "master" agent of an Internet backbone provider to a different physical location when its current home gets overloaded.

The RCA must also support flexible accounting schemes for QoS services, including both end-user-ISP and ISP-ISP accounting. In addition, the RCA interacts with network management, firstly for configuration and secondly for partial network failures. For configuration and service creation, the RCA interacts with the QMTool, which offers integrated and flexible service control.

### 3.1.3 QoS tools for supporting applications and network services

In order to demonstrate features of QoS enabled networks particular applications will be selected. These applications must be able to take advantage of the AQUILA architecture.

Broadband applications, such as Internet telephony or video stream services match this requirement and will be demonstrated.

There are two important aspects, which should be taken into account in order to deploy QoS over large scale IP networks (i.e. up to the whole Internet). One aspect is how to make QoS usable by end-user applications. Another aspect is how to deploy and operate the QoS mechanisms in a large scale IP network, including the aspect of the interaction between different ISPs and the interworking of different mechanisms (i.e. IntServ / DiffServ / MPLS / traffic engineering). AQUILA comprises the development of tools for both aspects. Both kinds of tools will be used for providing the trial services of AQUILA. They are research prototypes for a technology, which may get incorporated in products rather soon. The two kinds of tools are:

- ***End-user application Toolkit (EAToolkit)*** - to be used by application developers
- ***QoS Management Tool (QMTool)*** - to be used for network operation

The *EAToolkit* includes the development and maintenance of applications that can benefit from QoS features offered by QoS capable networks. This includes the “client side” part of the applications residing in the end-user host and eventually a “server side” part of the application, which could reside on a provider server. This aspect also includes all the mechanisms which enable a “legacy” application (i.e. built with no native QoS) to benefit from network QoS or to be slightly modified to interact with QoS mechanism. This EAToolkit should provide the functionality to:

- Allow non-QoS aware application to benefit from QoS
- Allow newly developed applications or modified existing applications to have an easy access to the QoS feature.
- Ensure compatibility with various methods for transporting QoS requests between end-user applications and the resource control layer (e.g. RSVP, CORBA- or DCOM-based APIs, Java applets)
- Ensure compatibility with several network service models for the provision of QoS (beyond the scope of RSVP)

The *QMTool* assists an Internet Service Provider in building services, even in co-operation with other ISPs and content providers. It handles the set of policies needed to ensure QoS in a real network like the client devices and the customers, which are allowed to make certain QoS requests. Additionally, it is responsible for the amount of bandwidth (or other QoS parameters) that can be assigned to a specific customer and is controlled by a particular server. The QMTool can be seen as belonging to a fourth layer interacting with the resource control layer and with the IP layer if needed, see Figure 3. The QMTool has some similarity to a network management application. However, it performs policy-based network management and therefore it controls the operation of the underlying resource control layer.

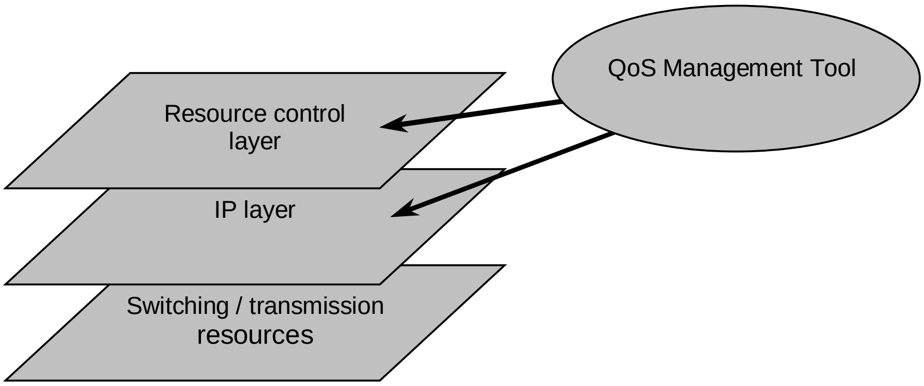



***Figure 3: QoS Management Toolkit in the layered architecture***

### *3.1.4 Distributed QoS Measurement*

Advanced measurement and monitoring is a base to ensure and supervise QoS and performance parameters of applications over QoS-based IP networks and to verify them and to optimise their behaviour.

To ensure that the developed architecture and its implementation (EAToolkit, QMTool, RCA, Edge Device and Router components) is practically useable for providing application-specific QoS demands, a distributed QoS measurement infrastructure with several application-like traffic generators is developed. Measurements are able to simulate real users with different behaviours simultaneously within a testlab trial to prove the stability of the proposed architecture before integrating it into the more complex and expensive field trial with real user scenarios. The distributed QoS measurement enables the project to evaluate specific network and application profiles which cannot be realised in a field trial.

Distributed QoS measurement introduces new functionality for the analysis of applications and protocols like automation of test scenarios with different protocol parameters and network configurations ("tuning"). It is based on distributed measurement agents within different kind of networking components controlled remotely by an operator via a user-friendly graphical interface. This allows specification and execution of measurement test suites in different modes (multiplexed, point-to-point, point-to-multipoint, multicast), traffic characteristics, QoS measurement requirements and resource reservation parameters dependent on the applications. These can be stored in a measurement information database together with network device parameters and the measurement results. The calibration curves for admission control can be calculated from this database.

### 3.1.5 Development Process and Methodology

One of the most challenging aspects of the project is the high speed of technological development in the area addressed by the project. It is very difficult - if not impossible - to define detailed objectives and technical approach for such a project in advance for a period of three years. Therefore, this project defines a project structure, which is able to adapt the project to the rapid changes in its scientific and market environment.

In the discipline of software engineering, approaches for dealing with projects of high-risk have been investigated already for some time. One of the most famous results is the so-called “spiral model” of system development (Barry Boehm 1988). Figure 4 shows a simplified version of this model. The basic idea is that the move from requirements to final product does not only go through several phases (as the basic “waterfall model” assumes) but that progress takes place in circles where some activities are repeated several times, but applied to more detailed intermediate products. In particular, the assessment of risk and the evaluation of alternatives should be part of any pass of the circle. Moreover, it is recommended to develop some kind of prototype at the end of each circle, even in the early phases of the project.

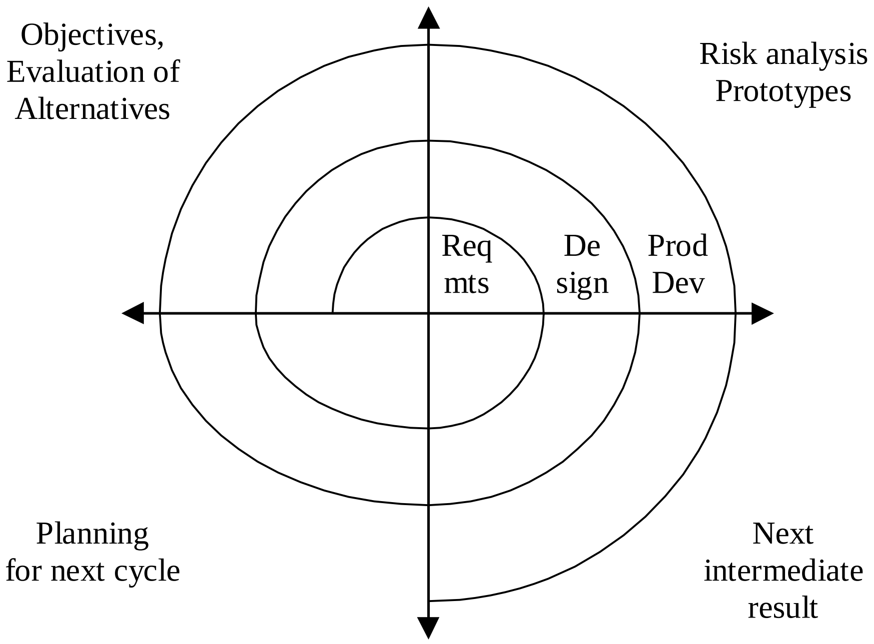



***Figure 4: Spiral Model of Development***

This basic idea has been adapted to the project structure for this proposal as follows:

- There is a workpackage (WP 1.1) for the whole project lifetime, which watches the current technological trends and evaluates alternatives to achieve the project objectives.

- There are six pre-planned cycles of development, leading to a first trial (in three iterations through requirement analysis, design and development) and to a second trial (in the same three kinds of iterations). Compared to Figure 4 above, the project adds three more cycles.
- In each development cycle, it is possible that some technological decisions of the project are changed. This is trivially true for the two iterations that explicitly deal with requirement analysis. But also during other project phases (design, product development), changes of direction are possible. Such changes may be initiated by the ongoing trend watch workpackage, but of course are carefully evaluated against the potential risks (i.e. risk of not using adequate technology vs. risk of delay or risk of reduced functionality).

This idea of a rather dynamic project underlies all further descriptions of the project structure.

## 3.2 Work Breakdown Structure

### *3.2.1 System Architecture and Traffic Issues (Workpackage Group 1)*

Workpackage Group 1 of AQUILA is analysing the requirements of multi-service applications and future users on the QoS in an IP-based network and specifies the system architecture and the traffic control mechanisms necessary to fulfil these requirements. In addition, it analyses the market relevance and the market potential of offered solutions.

The Workpackage Group is separated in 3 workpackages:

- **WP 1.1** studies the market situation and formalises requirements for the AQUILA project. Based on state-of -the-art technologies, it studies necessary enhancements to fulfil the user, provider and application requirements in the near future. Continuously, it follows current market trends and technologies with a good business potential. These results will have influence on the specification of the necessary QoS for future specification in WP 1.2.
- **WP 1.2** provides a complete specification of the network and system architecture, the services, the protocols and the algorithms to fulfil the user requirements defined in WP 1.1. This WP is the crucial basis for the implementation of the specified mechanisms in WPG 2.
- **WP 1.3** analyses in detail the necessary traffic control mechanisms inside the network to provide the required QoS to the users respectively the applications in a very efficient way. This enables network and service providers to get the most possible benefit from the installed network and to optimally use the network resources. The necessary traffic control algorithms are optimised either by simulation or by analytical modelling and offered as input to WP 1.2.

### *3.2.2 Prototype Implementation (Workpackage Group 2)*

The Workpackage Group I is split into three workpackages, each of which is responsible for the implementation of a specific component of AQUILA architecture.

- **WP 2.1** will be responsible for the design and implementation of a Resource Control Agent prototype. As a second result of this WP, a prototypical *QoS Management Tool (QMTool)* will be developed, which is used by network operators and Internet service providers to enable a given network service rather than a specific application.

- **WP 2.2** will focus on the design and implementation of a supporting toolkit for the construction of QoS-aware end-user applications. As a first result, a flexible *End-user Application Toolkit (EAToolkit)* will be developed. As a second result, the application software required for the user trials is provided according to the requirements of WP 1.1. This is restricted mainly to the integration of existing software with the help of the EAToolkit.

- **WP 2.3** will focus on the implementation of distributed end-to-end QoS measurement software, which is used to ensure the implemented architecture is capable of providing determined QoS guarantees. It is used to run the measurements in the trials of WP 3.2 and is therefore implemented in two steps. The first release of the implementation, completed for the first trial, includes the basic implementation with the measurement information database and several traffic generators for different types of applications. In a second step enhanced traffic generators and an interface to network management is integrated.

### 3.2.3 Integration and Trial (Workpackage Group 3)

The main contribution of the project is to devise and implement an integrated solution for QoS over IP networks. Since QoS support affects all major parts of the network, it is essential to integrate, demonstrate and analyse the developed components.

- **WP 3.1** is responsible for the testing of the individual system components and their integration. WP 3.1 is mainly active in the time frame between the global project milestones M11 and M12 as well as M21 and M22, respectively, as shown in the chart in chapter 9.5. They define begin and end of the integration and trial phase for the first and the second trial.

- **WP 3.2** will concentrate on the execution of a field trial. As the project follows a two-phased approach, it will run two trials. The first one, which will take place in a quite early stage of the project, will be done in a laboratory environment. The main purpose of the first trial is to get experience from the integration of the available equipment (e.g. routers, Edge Devices) with the developed prototypes (e.g. Resource Control Agents) and to get some early feedback for the performance of the AQUILA architecture. The results of the first trial will give input to all other WPs in order to refine their work. The second trial will follow a real user scenario and targets at the evaluation of the whole project (see also Figure 5).

- **WP 3.3** aims at the evaluation and commercial exploitation of the project results. Business plans for commercially promising cases will be developed. Both business-to-business relations and business-to-end-customer relations are included. Based on real cost and revenue figures, resulting models for alternative business cases will be simulated. As a major simulation result, break even point definitions are expected.

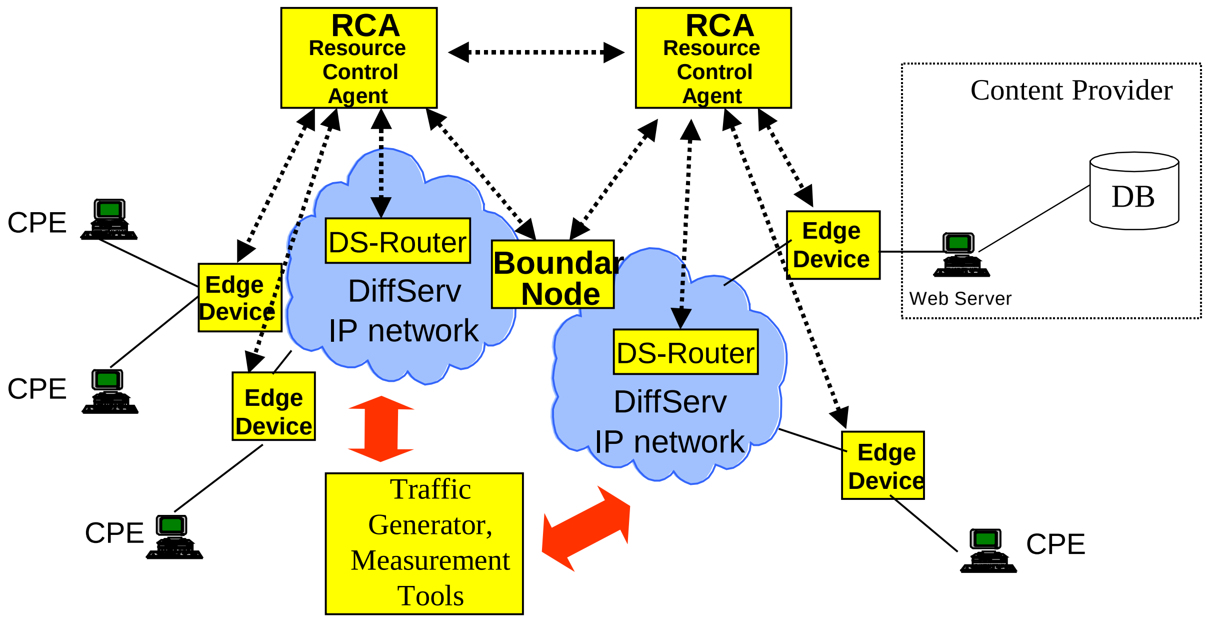
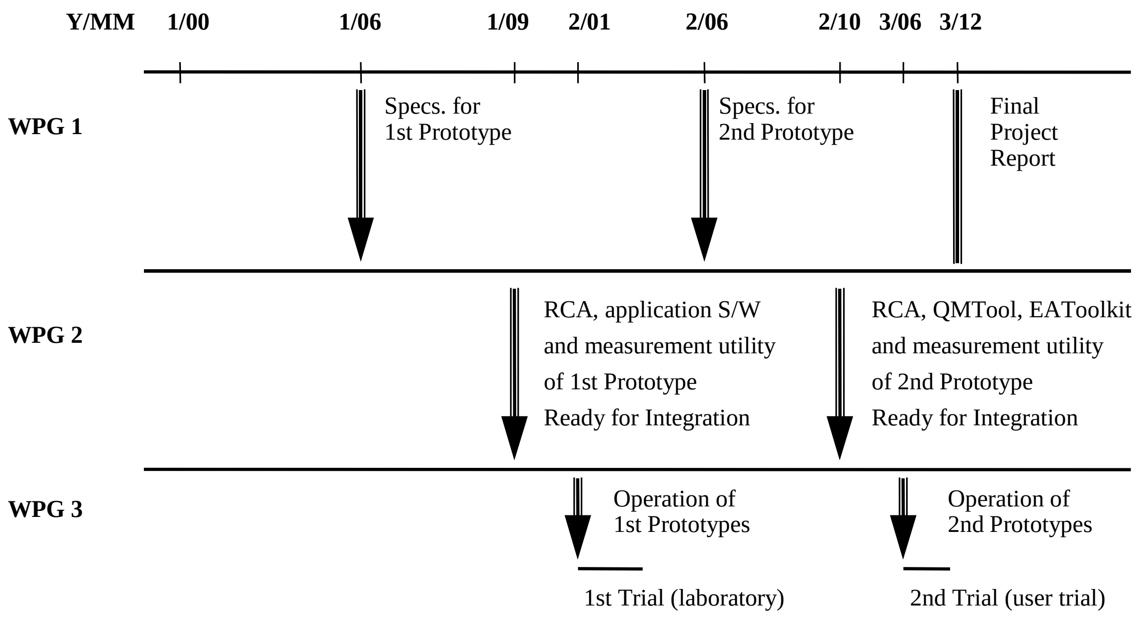



***Figure 5: Trial Network Architecture***

Figure 5 shows a possible Inter-ISP trial scenario with two DiffServ networks (Not shown here are the QMTool and the EAToolkit). The goal is to offer QoS for the users for applications offered by the content provider and other applications including several users. The RCAs control the DiffServ core and edge routers and manage QoS requests. This is accompanied by measurements to monitor QoS levels under various load situations (using traffic generators).

## 3.3 Interrelationship of Workpackages and Dependencies of Deliverables




The technical workpackages are organised in such a way that clear interfaces exist between them. These interfaces are represented by the project's deliverables. The flow of information based on deliverables among the three workpackage groups is such that WPG 3 depends on results from WPG 2 and WPG 1 and WPG 2 depend on results from WPG 1. This cycle occurs twice within the project, once for an early trial (phase 1) and once for the final trial close to the project's end (phase 2). The above figure shows this structure in more detail. For reasons of simplicity the feedback is omitted in this figure, i.e. the experience from implementation, integration and trial of phase 1 will be considered for phase 2. The experience from phase 2 will be incorporated into the final project report to be delivered by WP 0.

The interrelationships for the project's second phase are similar to the first. However, the focus during the second phase will be on the integration of enhanced versions of the RCA, QMTool, the EAToolkit prototype and performance measurements during the final user trial.

## 3.4 Workpackage descriptions

**Workpackage 0: Project Management**

**Objectives**

WP 0 will provide internal project management and overall co-ordination of activities, planning and control. It will ensure that the project objectives are met and will represent the contact point of the project to the Commission and the IST community.

**Description of work**

The management structure adopted for the project is described in chapter 9.7 of this document. Essentially, the Project Manager carries out the overall project management in co-operation with the members of the Project Management Committee. Workpackage leaders will be responsible for content and timely availability of project deliverables and formal progress reports.

Dissemination and implementation activities, as requested by the Commission, will be handled in this workpackage in order to prepare the Dissemination and Use Plan as well as the Technology Implementation Plan.

**Workpackage 1.1: Requirement Analysis**

**Objectives**

The WP gives project support in order to correlate the outcome with ongoing developments, user requirements and market trends. Key objectives are:

- Continuous market analysis to identify available products and implementations
- Requirement analysis from user's perspective for customer friendly application

Requirement identification from provider's point of view to allow seamless access as well as interaction of applications and networks on settled service level agreements

**Description of work**

WP 1.1 settles the initial step to place the project in the state of the art of Quality of Service support for IP networks. Following the standardisation activities of IETF and other relevant bodies on one hand and collecting requirements from users as well as providers on the other hand, it leads the development process of the project. This analysis will continue over the entire project period to ensure the specifications fit into the real Internet world. Upcoming products will be identified and compared to the requirements. Access and core network architectures are also analysed with particular emphasis on user friendliness.

## Workpackage 1.2: Specification

### Objectives

The aim of this workpackage is to define and specify services for QoS demanding applications, the QoS-architecture, QoS tools and toolkits and the interfaces between the architectural elements. In particular, it will:

- perform technical analysis of cutting edge technologies and prepare standardisation proposals for Internet-QoS provisioning
- define the network and user services including user applications and content servers;
- specify a cost-effective and scalable end-to-end QoS architecture, taking into account current IETF standards (e.g. DiffServ, IntServ, MPLS);
- define the architectural elements for scalable QoS provisioning over IP networks and their interfaces;
- specify the resource control layer for dynamic resource and admission control, including active network re-configuration based on dynamic load changes;
- specify the needed router and edge device mechanisms and interfaces;
- specify toolkits for end-user, server, management applications and distributed QoS measurement;
- contribute to standardisation bodies like IETF, OMG, ITU, etc.

### Description of work

This WP will select appropriate network and user services as well as applications, demanding Quality of Service, based on continuous interaction with the requirement analysis of WP 1.1. It will define and specify the system architecture, based on a systematic analysis, and produce a specification of network architecture and network elements on several levels of abstraction. A two-phased approach will be used. For the first trial, the system specification will cover a single ISP scenario. For the second trial, a scenario with several ISPs will be specified.

Together with WP 1.3 scalability analysis and performance estimations on several levels of abstractions will be carried out. Contributions to standardisation bodies will be part of the WP.

### Workpackage 1.3: Traffic Studies and Engineering

**Objectives**

This WP deals with traffic engineering and network dimensioning, admission control and mechanisms for traffic control in DiffServ based networks. The key-objectives are :

- to define the set of traffic engineering rules necessary to dimension a DiffServ network taking into account both static and dynamic admission control policies and the set of DiffServ PHBs to be provided;
- to analyse the performance and computational costs of existing algorithms for traffic and admission control by simulation and mathematical modelling;
- to specify the required traffic and admission control mechanisms as needed for the implementation in the devices used in the trial network;
- to study and propose novel traffic-control mechanisms where existing approaches are sub-optimal.

**Description of work**

As far as traffic engineering is concerned, this WP will define a set of models to properly dimension a DiffServ network, taking into account the QoS requirements of the different DiffServ PHBs to be supported.

The WP will additionally investigate highly dynamic procedures for admission control, allowing QoS guarantees on a “per call” level while maximising link utilisation. The overhead of these mechanisms in terms of processing-cost and message exchange between the architectural elements will be analysed. Simulation and analytical modelling will be used as needed.

Concerning traffic control, the WP will investigate mechanisms for scheduling and queue-management at router output-ports, traffic conditioning at edge devices and end-to-end congestion control in hosts.

### Workpackage 2.1: Service and Resource Control

**Objectives**

The workpackage aims at the provisioning of:

- a scalable and reliable design as well as the implementation of a dynamic RCA, which will offer

- distributed, measurement-based admission control;
- authentication and authorisation procedures based on a simple policy data base;
- interfaces to existing implementations of DiffServ routers and Edge Devices;
- a QMTool to support the deployment and operation of the new services.

**Description of work**

For the first trial, the WP will design and implement a prototypical Resource Control Agent with functionality restricted to admission control over a small portion of the DiffServ network (e.g. the DiffServ domain of a single ISP). The WP will take into account the specifications from WP 1.2 and WP 1.3 as well as the related work from IETF and Internet2 for bandwidth brokers.

For the second trial, the workpackage will utilise the experience from the first trial and will add further features in the prototype. The enhancements include procedures for inter-communication and inter-operation between multiple RCAs (multiple ISPs) and automatic network configuration according to the current traffic load. A prototype of the QMTool will be developed for that trial. At this phase, the WP will take input from WPs 1.2 and 1.3.

**Workpackage 2.2: QoS aware Applications and User Services**

**Objectives**

- To provide a methodology and a supporting toolkit for construction of QoS-aware end-user applications
- To enable the migration of existing end-user applications to QoS-awareness
- To ensure compatibility with various methods for communicating QoS requests between end-user applications and resource control layer
- To define and integrate the suite of application software required for the user services to be used in the trial

**Description of work**

As a first result, a flexible End-user Application toolkit (EAToolkit) is developed. It consists of middleware for installation on servers and end-user terminals. A modular architecture is defined, which enables the configuration of different adapters and wrappers for different application needs. A methodology is defined which enables migration of existing applications to QoS-awareness.

As a second result, the application software required for the user trials is provided according to the requirements of WP 1.1. This activity is restricted mainly to the integration of existing software with the help of the EAToolkit.

This EAToolkit will be used for providing the services in trial. The first trial is used for experiments with the underlying technologies for the EAToolkit. The final prototype of the EAToolkit is made available in time to be used for the provision of the application services for the second trial.

### Workpackage 2.3: Distributed QoS Measurement

#### Objectives

- To develop a set of utilities enabling distributed measurements of end-to-end QoS parameters within IP environments
- To design and implement application specific traffic generators
- To implement an interface to network management

#### Description of work

A distributed QoS measurement utility, which measures the QoS parameters (delay, jitter, packet loss and throughput) is implemented. This new utility features advanced facilities for remote performance and end-to-end QoS analysis (calibration curves) within IP networks.

For realistic loads on the network, application-like traffic generators (audio, video, data) at socket level and at application level are developed.

Because of statistical variances and measurement errors mechanisms for detection and evaluation of errors are included.

To get a complete specification of parameters used within the network components, an interface to network management is integrated to analyse the correlation between parameter settings of network components and end-to-end QoS.

### Workpackage 3.1: System and Network Integration

#### Objectives

This workpackage aims at:

- elaboration of a two-phase integration workplan and definition of the interoperability tests;
- continuous check of availability of all software and hardware components, identification and monitoring of the critical integration paths;

- stepwise integration and testing of all software and hardware modules including protocol stacks.

**Description of work**

The system will be integrated and tested step by step. Finally the whole system is object of the integration test. The main purpose of the integration will be the interoperability test between the system components.

Then, an overall integration workplan must be derived by considering a phased approach, which starts with standalone verifications, moving to island integration and finally obtaining global interconnection. Additionally, a monitoring process will be launched in order to control the equipment and software delivery and to guarantee the components interoperability.

The definition of the required tests to obtain interoperability between the islands and the selection of test equipment for that will also be considered within this workpackage.

### Workpackage 3.2: Trials and Measurements

**Objectives**

This workpackage aims at:

- Elaboration of a two–phased trial workplan with realistic user scenarios
- Elaboration of scenarios of service deployment and network configuration
- Running the trials
- Taking the required measurements needed for the evaluation of the network architecture
- Evaluation of the proposed architecture: provision efficiency, scalability, and resource control.

**Description of work**

As the project follows a two-phased approach, it will run two trials. The first one, which will take place in a quite early stage of the project, will be done in a laboratory environment. The main purpose of the first trial is to get experience from the integration of the available equipment (e.g. routers) with the developed prototypes (e.g. Resource Control Agents, Edge Devices) and to get some early feedback for the performance of the AQUILA architecture. The results of the first trial will give input to all other WPs in order to refine their work. The second trial will follow a real user scenario and targets at the evaluation of the whole project.

In both trials, the WP 3.2 will provide a set of scenarios that will describe:

- the network configuration;

- the required scenarios for the deployment of each service;
- the schedule for each service to be deployed in the trials;
- the schedule of the performance measurements.

For the performance measurements, the WP will use new test utilities developed within the project by the WP 2.3. Some performance measurements may also be made outside the planned trial seasons.

**Workpackage 3.3: Exploitation and Business Models**

**Objectives**

- To evaluate the project results,
- To develop a commercial exploitation plan based on the project result,
- To provide alternative business plans.

**Description of work**

WP 3.3 aims at the evaluation and commercial exploitation of the project results. Business plans for commercially promising cases will be developed. Both business-to-business relations and business-to-end-customer relations are included. Models for alternative business cases will be simulated based on estimated cost and revenue figures derived from trial results. As a major simulation result, break even point definitions are expected.