Managing QoS in a component-based framework

Extended Abstract

Hani Naguib*, George Coulouris*†, Heather Liddell*, Scott Mitchell†
{hanin,george,heather}@dcs.qmw.ac.uk, rsm31@eng.cam.ac.uk
** Department of Computer Science,Queen Mary & Westfield College*
University of London
†Laboratory for Communications Engineering, Cambridge University

Introduction

The QoS and configuration management aspects of distributed multimedia applications have received much attention in recent years and some progress has been made in the provision of network support for streaming data [Zha93] [Rot96] and dead-line based CPU scheduling [Chu99][Cou95] etc. Less progress has been made with integrating these mechanisms into object-oriented frameworks to facilitate the rapid construction and reconfiguration of multimedia applications. Real-time extensions to Corba [Sch00] have been suggested, and although some platforms integrate some of the low-level mechanisms required by multimedia applications, they tend to lack higher level mechanisms, for example automatic admission tests.

Our research has focused on the design and implementation of a distributed multimedia framework with explicit QoS support, called Djinn. Djinn adopts the component-based approach to application modelling (loosely based on Gibbs & Tsichritzis multimedia framework [Gib95]); applications are constructed from sets of interconnected components. Streams of multimedia data flow to/from the connection points (media ports) of components. In addition, model components can hierarchically encapsulate complexes of sub-components. These 'composite' components are useful in constructing complex applications and in creating reusable components.

In order to provide a heterogeneous distributed environment, our components are implemented as Corba objects. One of the novel characteristics of Djinn is the separation of the application into two layers; a Model layer, consisting of components used to model the QoS and connectivity of applications; and an Active layer, consisting of components that actually drive the application (the components that 'touch' the multimedia data). This separation is completely masked from application programmers, who in fact only interact with the model layer.

One of the benefits of this split-level approach is that it allows us to perform dynamic checks on the application before deploying and reconfiguring the distributed (active) components (model components are usually locally grouped). These checks include the verification that applications have been properly constructed and their components properly connected (part of Djinn's high-level configuration/management of applications). QoS tests can also be performed automatically, ensuring that system resources are properly managed.

QoS Support

We make QoS information an explicit part of the model layer. Model components export a QoSObject that represents the QoS characteristics of components and can be used by Djinn's runtime while performing QoS tests. Currently QoSObjects provide information regarding the resource requirements of modelled active components and any constraints it imposes on the media streams it processes. Although this approach currently focuses on the resource requirement aspects of QoS we believe that this can be extended to other QoS aspects such as security and fault tolerance requirements.

To provide QoS support, Djinn must be able to reason about these QoS characteristics. Our approach, has been to express the QoS characteristics in terms of mathematical relations whose variables are stream attributes. For example the CPU resource requirement of a Video Display component, would be related to the width, height and depth of the frames in the video it is receiving as well as the frame rate and is thus expressed as a mathematical relation of these attributes. During admission tests the various QoS characteristics of the application's components, the resource availability and application-specific QoS characteristics are transformed into a mathematical model. This model is then solved to find specific values for the various streams within the application. Unfortunately multimedia applications are complex and these models are initially highly non-linear, making obtaining solutions impractical. To circumvent this we take into consideration various characteristics of multimedia applications allowing us to simplify the QoS model into a linear model, which can be easily and efficiently solved.

Conclusion

We have aimed to build a framework for the construction of distributed multimedia applications using industry-standard mechanisms. Corba is used to provide remote access to heterogeneous application components. A certain level of lower-level system support is required (ability to reserve resources for instance), but specific mechanisms are not specified, allowing for the integration and extensions of various techniques. By modelling QoS characteristics as mathematical relations, we provide a standard and extensible approach to QoS support. For further information on Djinn please refer to [Mit99] or http://www.dcs.qmw.ac.uk/research/distrib/refs.html.

References

[Chu99] Hao-Hua Chu, Klara Nahrstedt: *CPU Service Classes for Multimedia Applications*. ICMCS, Vol. 1 1999: 296-301

[Cou95] G. Coulson, A. Campbell and P. Robin, *Design of a QoS Controlled ATM Based Communication System in Chorus*, IEEE Journal of Selected Areas in Communications (JSAC), Special Issue on ATM LANs: Implementation and Experiences with Emerging Technology, May 1995

[Gib95] Simon J. Gibbs & Dionysios C. Tsichritzis. *Multimedia Programming: Objects, Frameworks and Environments.* Addison-Wesley, Wokingham, England, 1995.

[Mit99] S. Mitchell, H. Naguib, G. Coulouris and T. Kindberg, "A QoS Support Framework for Dynamically Reconfigurable Mul-timedia Applications", in L. Kutvonen, H. König and M. Tienari (eds), Distributed Applications and Interoperable Systems II, pp 17-30. Kluwer Academic Publishers, Boston, 1999.

[Rot96] K. Rothermel, G. Dermler, W. Fiederer, *QoS Negotiation and Resource Reservation for Distributed Multimedia Applications*, Technical report 1996-rothermel-04, Fakultat Informatik, University of Stuttgart, July 1996

[Sch00] Douglas C. Schmidt and Fred Kuhns, *An Overview of the Real-time CORBA Specification*, IEEE Computer special issue on Object-Oriented Real-time Distributed Computing, edited by Eltefaat Shokri and Philip Sheu, June 2000.

[Zha93] L.Zhang, S.Deering, D.Estrin, S.Shenker and D.Zappala, *RSVP: A New Resource ReSerVation Protocol*, IEEE Network Magazine, September 1993.