A Customizable, Reconfigurable Deployment Environment for QoS Aware Multimedia Applications

Muhammad A. Khan
Institute of Operating Systems & Networks
Braunschweig University of Technology
Germany
0049-531-391-3154
khan@ibr.cs.tu-bs.de

Stefan Fischer
Institute of Telematics
University of Luebeck
Germany
0049-451-500-5380
fischer@itm.uni-luebeck.de

ABSTRACT

Despite QoS being a widely researched area, it has mostly been treated in the context of providing 'hard QoS guarantees'. Therefore, providing QoS has been clearly distinguished from the Best Effort Service. For the same reason most of the existing work to provide QoS in wireless environments relies on reserving resources (e.g; wireless bandwidth) in excessive amount, since the maximum available is not precisely known. At runtime, if the resources still fall short, a runtime re-negotiation is carried out and reallocations made. In situations where resources are scarce and hard QoS guarantees are not desired, pre-allocation of resources in excess is not justified. There is a class of applications, which relies on 'user perceived QoS' instead of demanding any hard QoS guarantees. In wireless networks of small-footprint devices, in particular where audio/video transmission is involved, a customizable environment with minimum footprint which can facilitate adaptive application execution is required. In this paper we present our ongoing work to handle dynamic adaptation of multimedia applications in order to provide certain level of user perceived QoS. We describe our implementation of a customizable and reconfigurable deployment environment which facilitates dynamic application adaptation. The deployment environment is statically reconfigurable (so that it has only those components which are needed – minimum footprint).

The work presented here builds on our on-going research in applying reflection and aspect oriented techniques to handle systemic quality of service concerns with emphasis on audio/video applications. The overall system is composed of two main entities, the development framework and the deployment environment. The development framework facilities application composition with minimum base functionality and the deployment environment together with the adaptation engine coordinates application customization and adaptive reconfiguration at runtime. The deployment environment is adaptive in two respects: it is statically customizable according to the deployment device (e.g., a laptop computer, PDA or a mobile phone) and dynamically reconfigurable according to user preferences and adaptation profiles.

Different devices have associated limitations, (e.g; a laptop can have enough power to encode video, however a PDA or a mobile phone may not have). Due to this, it would be useless to install those parts of the deployment environment on such resource-constrained devices. On the other hand different wireless networks have different charging policies (amount of data based, connection time based or free of charge) which affect the kind of adaptation to be invoked in a particular surrounding. This determines which parts of the deployment environment are not needed, so that those can be left out to save space. Other resource-conflicting adaptations (like those with respect to CPU and network bandwidth) are handled at runtime.

Device characteristics, adaptation preferences given by the user and networks' charging policies constitute different profiles. Individual components composing the application can have specific QoS properties associated with them, (like a codec has frame-rate, frame-size etc) or a video-renderer is capable of displaying the video in different sizes.

Adaptation preferences obtained from the user, device and network profiles (which are specified on system-wide basis) are mapped onto individual application components to customize the application statically. This customization is done by the aspect engine and the adaptation engine keeps balancing and reconfiguring these static customizations at runtime, using the feedback from the resource monitor.

In the poster, we show that our approach performs adaptively with very low overhead in case of video transmission in wireless environments and show some experimental results obtained from testing the system in a wireless video conferencing application.