

An Adaptation Architecture to Improve User-Perceived QoS of Multimedia Services for Enterprise Remote Desktop Protocols

Marcin Lubonski, Valérie Gay, Andrew Simmonds

University of Technology, Sydney, PO Box 123, Broadway NSW 2007, Australia

{marcin, valerie, simmonds}@it.uts.edu.au

Abstract— Limitations in the contemporary best effort WAN combined with an increase in demand of the business users for multimedia are the main challenges for remote desktop solutions. Thin client remote desktop protocols are now mature enough to work efficiently over low bandwidth and low quality links for normal office applications. However, they face significant challenges when working with more demanding data flows such as multimedia streams.

This paper introduces the server-side component of the architecture for remote desktop environment aiming at improving user perception of the more demanding and network sensitive multimedia applications and services. Our component maps of user-perceived Quality of Service (QoS) onto network-level performance parameters in order to improve overall end-to-end QoS. The set of perception rules defined for each application and type of user triggers the adaptation mechanisms to change the transmission parameters in order to meet end user expectations.

The adaptation mechanisms can optimize the usage of the available network resources within a dedicated network link both for different flows within single client connection and for all clients connected to the same server module. The result of such an optimization is a new set of connection parameters adjusted to network state changes and user perception of remote desktop.

In the paper we discuss requirements of such QoS control mechanisms and respective adaptation mechanisms, followed by the detailed description of the proposed component architecture.

Keywords: multimedia, user-perceived QoS, adaptation, traffic management.

I. INTRODUCTION

In an enterprise environment thin client remote desktop systems make it possible to work remotely on enterprise mainframe from the local branch PC over WAN or virtually from any place using mobile hand-held devices. Remote desktop systems client communicates with server over thin client remote display protocol [1]. The protocol encapsulates display updates of remote application, sends them from the server to the client and transmits back user input. Examples of this kind of systems

include: Citrix Metaframe Server [2][3], Microsoft Terminal Services [4], X-Window System [5], AT&T VNC [6][7] and Tarantella [8].

In recent years the popularity of those systems saw significant increase mainly due to growing interest in distributed model of desktop computing along with the emergence of Application Service Provider (ASP) business model and increasing mobility of employees. According to Gartner in the next years this trend is expected to continue and the number of desktop computers configured in thin-client mode should grow.

Along with the increase in popularity of remote desktops, a similar increase in user demands can be seen. The profile of the applications used in enterprise environment has changed diametrically nowadays including video conferencing, multimedia presentations and all types of entertainment. This creates a significant challenge for remote desktop systems, which were originally designed to work with standard office applications in LAN environment.

So far the focus was mainly on reducing bandwidth usage and providing smooth relay of display updates over low-bandwidth or poor quality network links. In comparison with the user-perceived Quality of Services (QoS) in LAN, remote desktop systems perform rather poorly over the best-effort WAN networks such as the Internet. Significant network latency and changing network conditions in the Internet links which are shared by multiple applications with different traffic characteristics result in the particularly bad performance for the multimedia content.

Even though there are technologies for QoS management in WAN such as Differentiated Services (DiffServ) [9] and Integrated Services (IntServ) [10] in combination with MPLS, they cannot be directly used to improve user performance of remote desktop systems for rich content. This is due to encapsulation of traffic within proprietary protocols in most remote desktop environments and due to application level encryption for each connection. In result, the implementation of any traffic management and QoS provisioning mechanisms on network-level can be realized within access networks only. But access networks so far do not offer almost any mechanisms to support QoS control [11].

Our approach addresses the existing limitations of remote desktop computing with regard to new multimedia services and applications by introducing the server-side component controlling performance of user connections. We recognize the need to place users perception and preferences in the centre of attention in order to meet their expectations. To address those needs our component maps user-perceived QoS to the performance of the underlying and dedicated network link between a remote desktop client and a server. In our approach, the data flows of different characteristics - corresponding to different types of applications or services - within one client connection are being dynamically shaped to the current network conditions and user-perceived performance of remote applications and services. Apart from these adaptation mechanisms the component we propose can optimize network resource usage among all clients sharing the same link.

The proposed server-side component is the first step to create a complete end-to-end/client-server multilayer architecture for improving user-perceived QoS in remote desktop systems. The flexible adaptation mechanisms implemented within our proposed architecture should lead to an overall amelioration of the user-perceived QoS for remote desktop applications and services.

This paper studies the requirements and design of architecture of the server side component for mapping between the user-perceived and the network-level QoS.

We use the term *Quality of Service (QoS)* in the following meaning: 'QoS is the collective effect of service performance which determines the degree of satisfaction of a user of the service' [12]. Throughout the paper we make clear distinction between network-level QoS and user-perceived QoS. Network QoS is expressed by network-level parameters such as: jitter, delay, loss, throughput and availability while user-perceived QoS, sometimes referred to as *Quality of Experience (QoE)* is defined in terms of application-level performance parameters specific to each application (for video streaming this could be: resolution, frame rate, color depth, etc.).

We also will be using term the Traffic Management referring to *all* the cooperative mechanisms which make possible to offer QoS [13]. Although Traffic Management refers in most cases to mechanisms at the network-level its meaning is often extended to the mechanisms on higher levels. In the paper we indicate when we talk about traffic management on application level.

The paper is organized as follows. Section II outlines the research relevant to the scope of this paper. Section III defines the problem addressed in this paper Section IV provides details of environment for remote desktop solutions. Section V presents the assumptions made for the design of our architecture. Section VI describes each component of the architecture and its functionality. Finally Section VII concludes the paper.

II. RELATED WORK

The state-of-the-art in the remote desktop domain is mainly created by the technological achievements and research progress of the main competitors on the market. The most popular solutions include X for Unix/Linux systems, Microsoft Terminal Services and RDP, Citrix Metaframe server with ICA protocol, VNC and its enhanced version TightVNC and finally Tarantella and SunRay systems. Apart from the main-stream players several backstage players contributed to the current image of desktop remoting. These include LBX, Laplink, PC Anywhere or GoToMyPC. While these systems were mainly designed for LAN environment providers such as FutureLink [14], or Runaware [15] use thin-client remote desktop technologies in ASP model delivering desktop relying over WAN links.

The change of the profile of applications and services used by remote desktop systems end-users, towards rich content, created a significant challenge to overcome the in best-effort networks such as the Internet. J. Nieh in his comparison [1] outlines the most significant limitations of the main players on this market and gives indication of future improvements to build multimedia-ready thin-client remote desktop solution. His research shows that remote desktop protocols cannot operate in networks having significant latency and with quickly changing network characteristics.

The resource requirements are normally expressed in terms of network QoS parameters such as throughput, packet loss, delay and jitter. Among those the three latter are the most important for streamed multimedia. There are a number of different factors having the most significant influence on these parameters for such a resource-demanding traffic in WAN. Ashton et al. [16] classifies them into three main groups:

- **Compression and/or Coding** – this encompasses number of technologies from video codecs, error correction techniques to adaptive streaming over best-effort networks
- **Caching** – referred in this context to application level caching, the technologies in this area range from local caches deployed on the edge of access network to sophisticated infrastructures referred as Content Delivery Networks (CDN)
- **Traffic Management/QoS** – this includes big selection of different approaches on all levels of ISO-OSI model from scheduling and queuing algorithms, through network QoS frameworks built on top of network equipment supporting this technologies, to application level specific devices or adaptation mechanisms on the protocol

As mentioned earlier, due to specifics of thin-client remote desktop systems, the usage of infrastructure elements such as application level proxy or caches in order to improve performance of applications and services rich in multimedia content is very limited. Similarly, compression and coding are

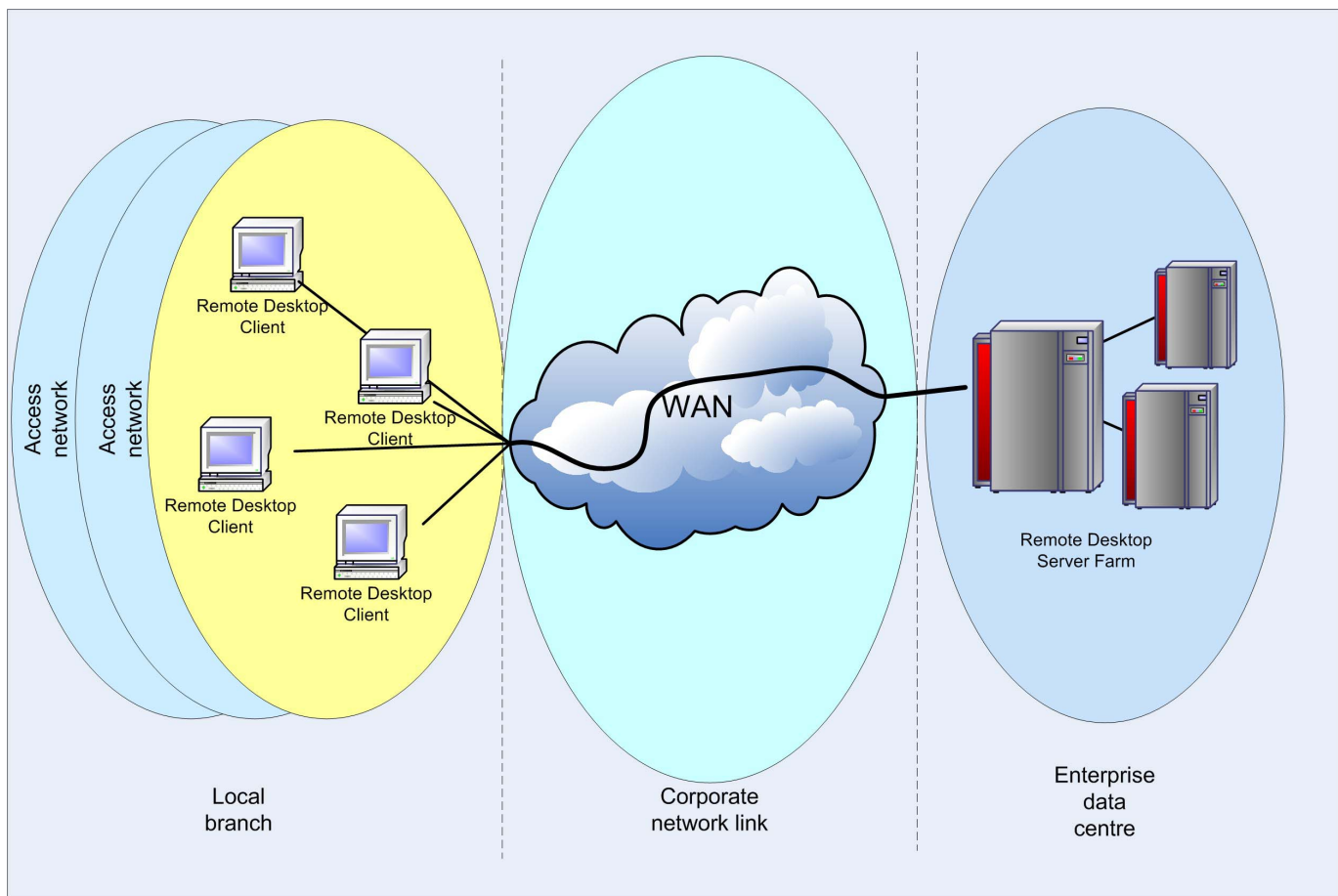


Figure 1. Remote desktop environment

more in the domain of the proprietary protocol design, which is out of the scope of this paper. A curious reader who wants to learn more about research in those areas will find an interesting overview of those techniques for multimedia streaming in a paper by Wu et al. [17].

Traffic Management and QoS research is dominated by work on DiffServ and IntServ frameworks combined with such WAN network technologies such as MPLS and lambda switching. From these environments originate such now well-established traffic management techniques such as: admission control, traffic shaping, traffic aggregation and classification or policy adaptation, which laid basis for the server-side component we are introducing here.

Despite existence of feasible network QoS techniques for WAN their implementation is still very limited. This is due to many reasons including economic as well as difficulties in providing end-to-end QoS mechanisms between different ISPs. Provided that those problems are solved there is still no sufficient support for QoS in access networks.

Although the traffic management and QoS techniques originate from the networking level, they can be successfully implemented at higher-levels providing mechanisms operating on the same level as application protocols. For instance, products such as Packeteer Packet Shaper and Expand Network Accelerator can provide QoS control in WAN for application protocols with specific focus on business environment. They use traffic management techniques on both application and network layer shaping the application flows depending on its resource requirements for network resources.

Applying multilayer approach to QoS provisioning across ISO-OSI proved to be a very interesting and promising area of research. The results of the joint research projects AQUILA, TEQUILA and CADENIUS show that introducing hierarchical approach to QoS provisioning can result in very powerful control over resource utilization [18][19][20].

Despite all those efforts both from research and industry environments there is no complete multilayer and end-to-end solution to QoS management for contemporary WANs. At the same time many network-sensitive application level protocols developed their internal adaptation mechanisms. Those

mechanisms are able to adjust to the particular network conditions taking into consideration information about the application performance. They are implemented in many streaming platforms as the internal mechanisms of the specialized audio or video codecs. Those approaches are unfortunately very often proprietary or limited to particular application or protocol. The more generic approach, which we to some extent follow in our study, is MPEG-4 Delivery Multimedia Integration Framework (DMIF). DMIF layer acts as a negotiator adjusting multimedia stream's parameters to current network performance. In DMIF framework, network requirements are defined for each type of multimedia stream using network QoS parameters. This approach is based on static end-to-end resource reservation and requires appropriate support in the transport layer which limits its application in contemporary highly heterogeneous networks.

A more flexible adaptation framework was proposed by U. Shrinivasan and S. Nepal [21]. Their pilot DAVE system uses information about user perception to dynamically adapt audio/video transmission to the user requirements. The focus in their work is mainly on audio/video streams which are known to be the most demanding in terms of network resources and stability of network parameters.

III. PROBLEM DEFINITION

In general the structure of the enterprise remote desktop environment can be divided into three main areas (refer to Figure 1):

- Access networks from where end-users connect to the central server over remote desktop protocol
- A WAN/LAN link connecting user terminals in the remote office with the server located in the company headquarters
- Remote desktop server or server farm located in the headquarters data centre

In enterprise remote desktop environment end-users use virtual desktop to work on the remote server. The WAN (or LAN) network link connects end-user terminal with a central server in the enterprise headquarters. From the end-user point of view they are working on a local machine and the network in between the terminal device and remote server is completely transparent. The last element – the server – manages user connections and creates a virtual work environment according to the user profile and permissions.

The effect of a virtual desktop is achieved by capturing screen updates by the server and encapsulation into remote desktop protocol. The protocol sends the desktop updates to the client and transmits back user input. So far there is no single standard for remote desktop protocol. In result the

communication between server and the client uses proprietary – and in case of commercial products closed – protocol.

Majority of the remote desktop systems propagate display changes on the server to their clients using server-push model [1]. Such an approach, when the server initiates data transmission, seems to be more efficient comparing to the more resource demanding situation when a client requests an update. In order to reduce the required bandwidth usage many servers send updates to connected clients only when the actual display on the server changes.

Systems used nowadays in enterprise environment applications might combine many different types of content, (such as textual data, multimedia streams or print spooling information). This leads to a more complex situation where in the same connection from client to server there is a multitude of flows, each with its specific network characteristics and requirements (for instance, Citrix [22]). Those requirements can range from time- insensitive data transfers such as WWW or FTP to the real-time transmissions of voice or video. In result even application protocols having some internal adaptation to current network conditions combined with other type of traffic can experience a decrease and not an improvement of performance. In more complex interaction among different flows and heavily used network this can lead even to total decrease of quality.

Apart from the use of proprietary protocols in the leading remote desktop solutions user connections can be additionally encrypted on the application level before entering the WAN link. This is in order to ensure that, often very sensitive data transferred over the network, are not tampered with.

In the environment with such characteristics there are two main areas for possible enhancements aiming at preparing remote desktop systems for multimedia content. Those are: protocol modifications and resource management.

At the protocol-level the enhancements will include modification of protocol primitives, encapsulation of high-level display calls into the protocol packets as well buffering and scheduling of screen updates. Recently there is a significant contribution to the work in that area and the first tests of the implementations gave some promising results.

On the other hand the techniques such as traffic management, resource reservation, or various prioritization schemes can be used to intelligently manage available resources.

In our approach we focus on the resource management aspects. The paper studies the design of an extension of existing remote desktop systems which using traffic management techniques and user-perception models will be capable of improving the multimedia performance. Based on the characteristics of the environment we are working in, we define following requirements for our approach:

- dynamic adaptation of the remote desktop protocol characteristics to changing network conditions
- application of the adaptation mechanisms on per-flow basis addressing the issue of possible multiple flows in one user connection
- network resource management based on the mapping between user-perceived QoS and network-level QoS parameters
- positioning of all management mechanisms on the network edge because of the proprietary nature of remote desktop protocols and application level encryption of those protocols when entering WAN
- guarantee of accessibility for the most critical or system applications as long as the link client-server is functioning
- minimization of possible changes in the existing structure of the protocol and remote server implementations
- easy integration with existing proprietary remote display protocols
- support for multiple user terminal devices

From the traffic management perspective the difficulty of fulfilling the requirements of network-sensitive multimedia applications, is caused by intrinsic limitations of contemporary networks. Lack of deterministic QoS parameters in contemporary best-effort networks make it impossible to guarantee any strict link parameters guarantees such important for multimedia traffic. As the analysis conducted by Packeteer [23] shows the problem cannot be solved by simply increasing the bandwidth. Without the control over resource usage in a shared link any additional resources are very quickly used up by the most resource-consuming applications. Additionally, for multimedia streams more important than sufficient bandwidth are such parameters as jitter and latency.

Our approach in designing the server-side component for remote desktop systems is based on the presented adaptation solutions. We extend them, however, by using perception information from the end-user as the source of adaptation changes. Additionally, following the design of the leading thin-client remote desktop protocols available we design our component in such a way that it is able to manage multiple flows with different requirements depending on the type of application/group of applications they are associated with.

IV. ASSUMPTIONS

We made several assumptions designing the architecture of the component for remote desktop protocols adaptation. Those assumptions define both working environment and specify scope of the proposed enhancements. They are as follows:

- complete control over the link parameters between remote client and remote server by creating, for instance, a dedicated channel over WAN with agreed QoS parameters
- existence of the user model for each application available to remote users over thin-client protocol;
- sufficient bandwidth: in the links between clients and the server, for the incoming and outgoing traffic on the end-user machine
- QoS control in the access networks preventing possible congestion in the local network
- transparency of implemented network technologies for applications; both in WAN and LAN links

As depicted in Figure 1 access network providing access for remote clients is connected to the server farm in the data centre using dedicated link. We assume that this link is separated from other traffic by external mechanism and is provided with some QoS guarantees on bandwidth, jitter, packet drop, etc. So even though there is possible variation of network parameters in this link they are bounded as specified in Service Level Agreement (SLA).

The user model for each application is a core element of our approach. It is defined to formalize user-perception information. For the purpose of this study we assume that we have user models for each application users can request to be added to the pool of applications available on their virtual desktop. User model is expressed terms of a mapping between application-level and network QoS parameters.

In order to improve overall perception of remote desktop protocols we have to make sure that other data flows originating from the same client and then the same local network are not interfering with our traffic. To limit any possible influence of other data flows we assume the existence of two additional elements in the scenario from Figure 1.

The first one is working on the end-user machine (or similar DCE device such as ADSL modem) and is responsible for bandwidth management for access to LAN. This component will guarantee required bandwidth from end user to the edge of WAN assuming that the network has sufficient resources and is correctly configured to handle all traffic to and from WAN. There is a selection of software available on the market offering such functionality [24][25].

The second element is responsible for management of traffic exchange with WAN. Even though we assume in this paper that there is a dedicated link over WAN for remote desktop data traffic, it could be shared among several clients. Therefore it is essential to provide an intelligent element on the edge of network controlling outgoing traffic and provide fair access to resources for all users. To do so it has to be aware of the application level information such as user session, user type if there is user service differentiation.

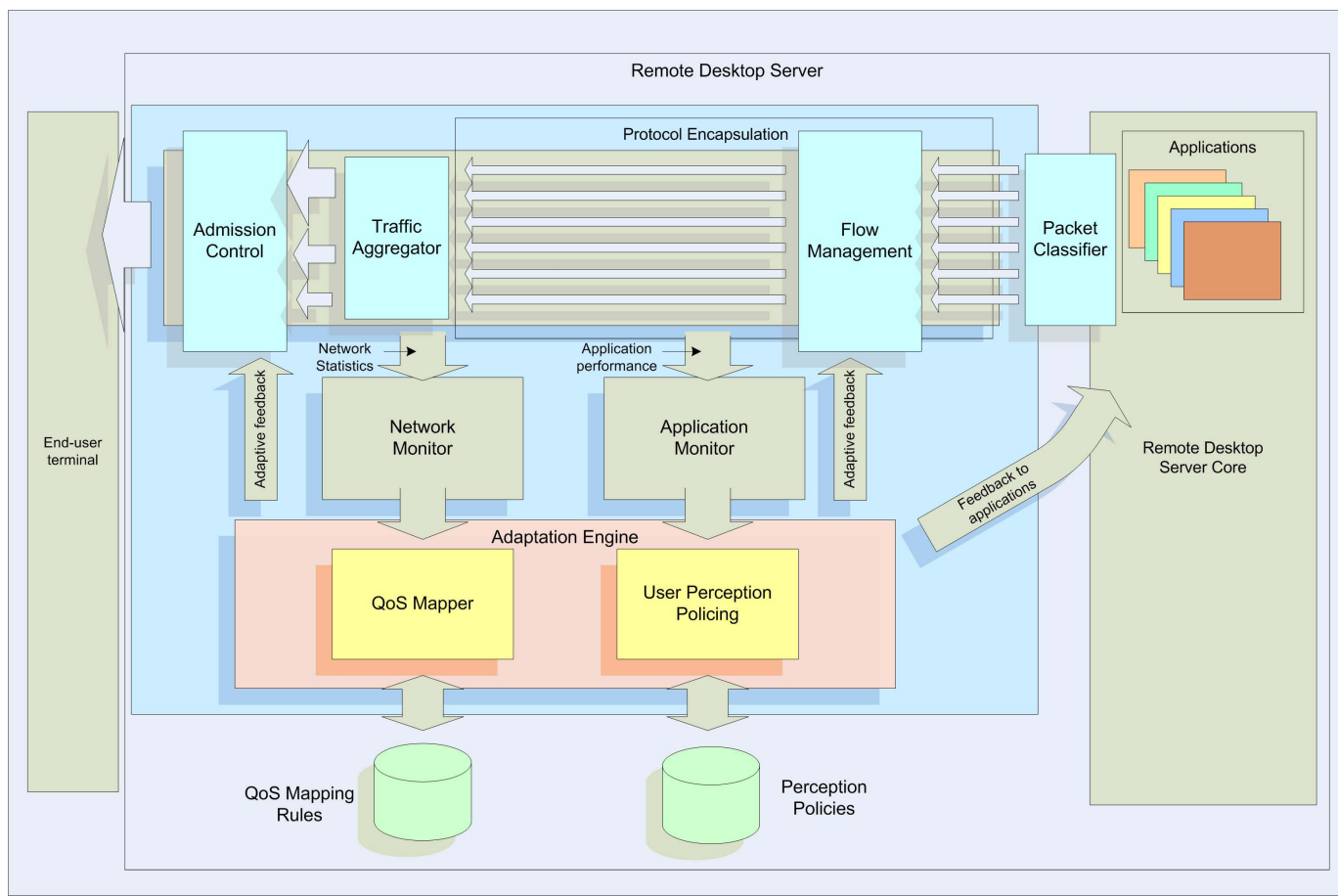


Figure 2. Remote desktop protocol adaptation - the architecture

This element should control as much outgoing as incoming traffic to avoid possible congestion. Additionally, it should monitor network usage and adaptively reuse resources allowing injection of new traffic to the network or prevention from opening a new session over WAN link if necessary. The products offered by Packeteer [23] and Expand [26] could in large extent provide required functionality for this element.

V. SERVER-SIDE COMPONENT FOR REMOTE DESKTOP PROTOCOL ADAPTATION

In our approach we focus on the resource management aspects. The end-user perception is the focal point of our proposition. We use mapping of user-perception feedback to dynamically adapt resource parameters and in result improve the overall system performance. We designed the component implementing such a mapping as a remote desktop server extension. Because of the efficiency and popularity of server-push mode of updates, we place the proposed component on the server side.

The component architecture is composed of several functional elements corresponding to the core mechanisms used

as its basis. Those mechanisms include: user perception monitoring, mapping between user-perceived and network QoS, application adaptation mechanisms as well as traffic management mechanisms – traffic shaping and aggregation and admission control.

The architectural elements can be categorized into two main groups depending on the scope of the adaptation mechanisms they implement (see Figure 2).

Per application flow adaptation, including:

- **Remote Desktop Protocol Encapsulation** (with Flow Management and Packet Classifier) – responsible for flow marking and protocol encapsulation according to the adaptation rules
- **Network Monitor** – monitoring network performance along the path from the server to the user
- **Application Monitor** – monitoring application performance and triggering application adaptation changes

- **Adaptation Engine** (with QoS Mapper and User Perception Policies) – responsible for QoS mapping

Per connection adaptation, including:

- **Traffic Aggregator** – responsible for aggregation of traffic with similar characteristics between user connections
- **Traffic Admission Control** – responsible for mitigation of mutual influence of flow aggregates with diametrically different characteristics

Additionally, the architecture includes two rules databases: **QoS Mapping Rules**, storing mapping rules between network QoS parameters and application level QoS, and **Perception Policies** containing user perception profiles.

The QoS management mechanisms, based on the user perception, benefit from the observation that some changes in application level parameters result in significant increase or decrease of user-perceived QoS while the others remain unnoticed. For instance, previously mentioned analysis of temporal and spatial parameters of video streaming applications conducted by Srinivasan and Nepal [21][27] showed significant deterioration in user-perceived performance when the number of displayed frames per second or color depth was reduced below some threshold.

This information can be used to create a user perception model which, in turn, can be stored in Perception Rules database. When a user connects to the server over remote desktop requesting access to this particular application user perception model is extracted from the database. It would be then used throughout the time of client connection as a reference for monitoring user perception changes.

During user connection to the server Network Monitor collects statistics about the network link to the client and compares them with appropriate service level rules for the particular type of user. This information would serve as a basis to User Perception Policing to verify whether the user perception requirements are met. If there is significant and persistent violation of limits of the parameters required for a particular data flow, the system would trigger the adaptation procedure.

The procedure could be as follows. In the first step the parameters of the user connection are modified; for instance allocating more bandwidth to particular data flow if possible. Failing that, the admission control would try to modify parameters of the flow aggregate to which this data flow pertains in order to meet requested requirements. Finally, if that fails the feedback would be provided to the application and if any adaptation mechanisms within application itself are available, they would have a chance to compensate for with the current network state. Within this procedure the adaptation component would verify influence of possible changes on other data flows

and assures that none of the critical applications are disadvantaged.

If the user connection consists of more than one data flow with distinctive characteristics upon a change in the network state the Adaptation Engine would try to adjust resource allocation in such a way that user-perceived QoS levels for all flows are preserved. If such an allocation is impossible, an adaptation procedure would try to compromise between QoS requirements for each flow. It would consequently distribute resources according to the information about user perception, current network state and QoS mapping rules. Depending on its priority, a particular flow might be gradually shut down until the required network resources are available again.

To perform per-flow operations there has to be means to distinguish between flows. In one of the possible implementations the traffic shaping mechanism could use information provided by Packet Classifier – responsible for the classification and marking of each application data flow - to reflect its traffic characteristics and user-perceived QoS requirements.

In the client-server model it is quite likely that there is more than one connection to server so different user sessions can influence each other. Traffic Aggregator and Admission Control are responsible for mitigation of negative effects caused by sharing the same link among different users. Their role is to aggregate the traffic into one of the predefined classes of data flow and provide fair access to available resources within the same class. For data flow aggregates with different characteristics Admission Control component would manage their interactions and optimizes network resource usage.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we addressed the problem of efficient delivery of multimedia content in the remote desktop environment. We presented the architecture of server-side component, designed to improve user-perceived QoS for multimedia applications. The design of the component provides a framework for implementation of remote desktop protocol adaptation mechanisms.

The strength of our approach lies in the mapping between the application-level (e.g.: video frame rate, frames dropped, etc.) and the network-level QoS parameters (e.g.: jitter, delay, packet loss, etc.). We use end-user feedback on user-perceived application performance of the virtual desktop to trigger resource allocation changes. Using user perception information provides the means to construct dynamic adaptation mechanisms for remote desktop protocols.

The most significant characteristics of our component architecture and its capabilities can be summarized by the following list:

- change resource allocation dynamically to compensate user perception changes

- tracks, quantifies and stores user perception information as source of remote desktop protocol adaptation rules
- bases/optimizes the resource allocation based on the mapping between this what user experiences and how the network performs
- limits the number of changes in the protocol and existing systems – ensuring backward compatibility with previous versions
- ensures that the most critical or system applications are always accessible
- optimizes resource allocation between requirements of multiple data flows in single user connection as well as between user connections to the same server
- stays independent from end-user terminal typeable to support multiple terminals
- uses resourced efficiently depending on the network link parameters
- provides interface to the existing application specific adaptation mechanisms
- uses efficiently available resources, managing new flow creation and resource recovery after the flow , important for links with lower-bandwidth or higher traffic saturation

We believe that intelligent resource (in particular network resources) in combination with extensions to the protocol definition can be a significant step forward in order to extend remote computing model to efficiently handle multimedia content.

In the next step we will focus on detailed specification of the adaptation mechanisms within the designed component. Such procedures depend on definition of mappings between user-perceived QoS to network QoS parameters. It is an open question how to represent those mappings and how to store and manage them efficiently. In order to implement adaptation for remote desktop protocols the signaling protocol has to be modified or added to existing protocols in order to provide end-user feedback on the perceived performance.

This can lead to a number of integration issues with existing remote desktop systems and requires analysis of possible impact on existing implementations.

ACKNOWLEDGMENT

We would like in this place thank Citrix Systems for the very productive cooperation and help during work on this paper. In particular we would like to thank Dr Steve Parry, Mr Tony Low and all others who contributed in any means to development of concepts presented in this paper.

REFERENCES

- [1] Limits of WideArea ThinClient Computing, Albert Lai and Jason Nieh, *Proceedings of the ACM SIGMETRICS 2002*, Marina del Rey, CA, June 15-19, 2002
- [2] Citrix MetaFrame Presentation Server - *Product Overview*, Citrix Systems, available at: <http://www.citrix.com/site/ps/products.asp>
- [3] Citrix Systems, "Citrix MetaFrame 1.8 Backgrounder", *Citrix White Paper*, June 1998
- [4] Microsoft Corporation, "Windows 2000 Terminal Services Capacity Planning", *Technical White Paper*, Redmond, WA, 2000
- [5] R. W. Scheifler and J. Gettys, "The X Window System", *ACM Transactions on Graphics*, 5(2), Apr. 1986
- [6] Virtual Network Computing, Tristan Richardson, Quentin Stafford-Fraser, Kenneth R. Wood, Andy Hopper, *IEEE Internet Computing* 1999
- [7] VNC 4.0 Documentation, Real VNC Company, <http://www.realvnc.com>
- [8] Tarantella ® Enterprise 3™ Software A Technical Overview, *A Tarantella White Paper*, August 2001
- [9] RFC 1633 - Integrated Services in the Internet Architecture: An Overview. R. Braden, D. Clark, and S. Shenker, *Informational RFC*, June 1994.
- [10] RFC 2475 - An Architecture for Differentiated Services, Steven Blake, David Black, Mark Carlson, Elwyn Davies, Zheng Wang, and Walter Weiss, *IETF*, October 1998.
- [11] QoS Roadmap, Maurizio Molina, Ilka Miloucheva, Sandra Tartarelli, *Next Generation Network Initiative IST*, Deliverable 2, October 2002
- [12] Terms and definitions related to quality of service and network performance including dependability, *ITU-T Recommendation E.800*, August 1994
- [13] QoS and queuing disciplines, traffic shaping and admission control, Marko Luoma, *Seminar at Laboratory of Telecommunications Technology*, Helsinki University of Technology, 1999
- [14] Charon Systems Inc., <http://www.futurelink.net>
- [15] Runaware Inc., <http://www.runaware.com>
- [16] The Three Components of Optimizing WAN Bandwidth, Ashton, Metzler & Associates, *Technical Whitepaper*, March 2003
- [17] Streaming video over the Internet: Approaches and Directions, *IEEE Transactions on Circuit and Systems for Video Technology*, Vol. 11, No. 1, February 2001
- [18] Admission Control for Providing QoS in DiffServ IP Networks: The TEQUILA Approach, Eleni Mykoniati et al., *IEEE Communications Magazine*, January 2003
- [19] AQUILA: Adaptive Resource Control for QoS Using an IP-Based Layered Architecture, Thomas Engel et al., *IEEE Communications Magazine*, January 2003
- [20] CADENUS: Creation and Deployment of End-User Services in Premium IP Networks, Giovanni Cortese et al., *IEEE Communications Magazine*, January 2003
- [21] Adaptive Video Highlights for Wired and Wireless Platforms, Surya Nepal, Uma Shrinivasan, *Proceedings IEEE International Conference on Multimedia and Expo*, Baltimore, USA, July 2003
- [22] Supporting Client Devices Using the Citrix Virtual Channel SDK, *Citrix Whitepaper*, Citrix Systems 2004
- [23] Controlling Bandwidth and PerformanceWith PacketShaper® Product, *Packeteer Whitepaper*, 2004
- [24] CCProxy Bandwith Controller, <http://www.bandwidthcontroller.com/>
- [25] Wondershaper, <http://packages.debian.org/unstable/net/wondershaper>
- [26] Expand Inc., "Expand Networks Accelerators, A Technical Overview", April 2004.
- [27] Video Quality for Adaptive Delivery: A Theoretical Framework, Surya Nepal and Uma Srinivasan, *CSIRO Mathematical and Information Sciences*, Internal report July 2002