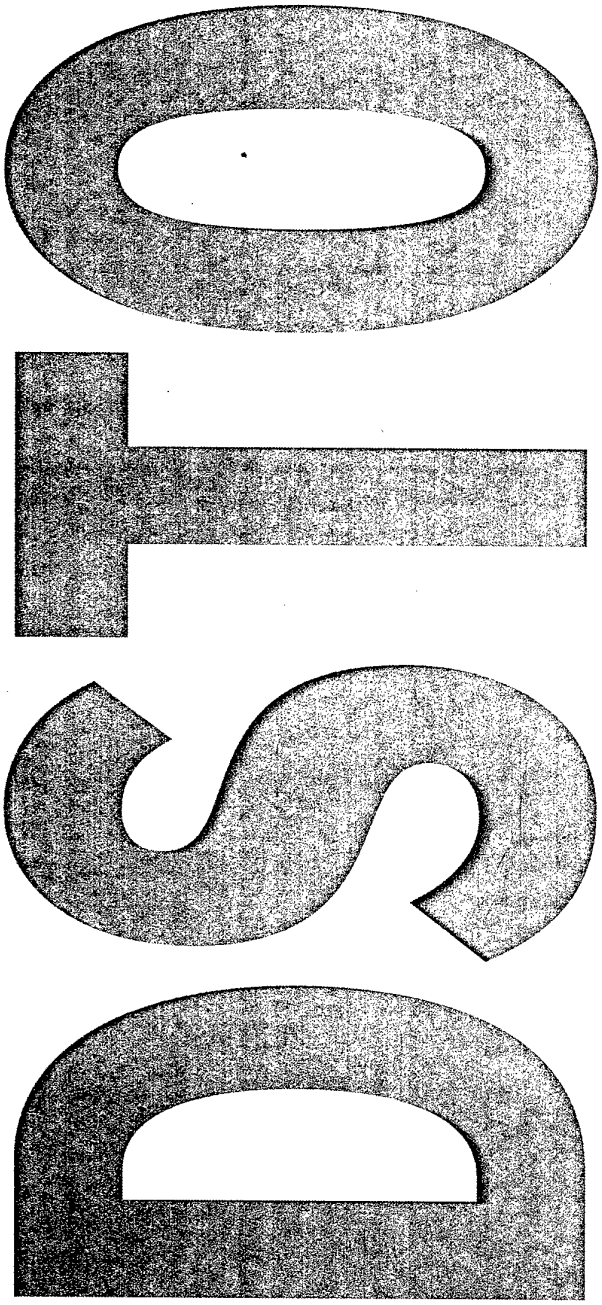
Australian Government
Department of Defence
Defence Science and
Technology Organisation

Jun 2004



Enabling Headquarters Reachback : Adaptation of Collaborative Applications for Tenuous Communications

T. Andrew Au and Cindy Tran

DSTO-TR-1588

Australian Government
Department of Defence
Defence Science and
Technology Organisation

Enabling Headquarters Reachback : Adaptation of Collaborative Applications for Tenuous Communications

T. Andrew Au and Cindy Tran

Command and Control Division
Information Sciences Laboratory

DSTO-TR-1588

ABSTRACT

The ability to access information on demand is a critical capability in the battlefield. Deployed military headquarters are often located in a challenging tactical environment, whereby major disruptions to collaborative work lead to an impairment in the operational capability. These challenges in turn stem mainly from volatile connectivity and sporadic network mobility, which are a feature of the unpredictable nature of the battlefield. A network management approach based entirely on resource reservation is especially difficult in this environment since disturbances and unanticipated events often occur with network-based applications. This paper explores the concept of network awareness in support of continued operation in an unpredictable battlefield environment. In particular, we propose a platform-independent event-delivery framework to facilitate application adaptation, and demonstrate the advantages of this approach in supporting the proactive management of applications.

Published by

DSTO Information Sciences Laboratory
PO Box 1500
Edinburgh South Australia 5111 Australia

Telephone: (08) 8259 5555
Fax: (08) 8259 6567

AR-013-122
June 2004

APPROVED FOR PUBLIC RELEASE

Enabling Headquarters Reachback : Adaptation of Collaborative Applications for Tenuous Communications

Executive Summary

The ability to access information on demand is becoming a critical capability in the battlefield. Reachback is the concept whereby the forward elements seek support from the rear to increase the survivability of the command function through increased redundancy, manoeuvrability and lower signature. The basic notion of reachback is to minimise the footprint of forces deployed in the area of operation and takes advantage of substantial information resources available at a homeland headquarters.

An essential part of the fabric of military operations is the information systems in headquarters, providing the critical capability to collect, process, and distribute relevant data to remote locations, thus allowing the military force to gain dominant battlefield situation awareness. However, deployed headquarters are often located in a challenging tactical environment, whereby major disruptions to collaborative work lead to an impairment in the operational capability. These challenges in turn stem mainly from volatile connectivity and sporadic network mobility, which are a feature of the unpredictable nature of the battlefield.

This report describes the conventional approaches to quality of service (QoS) and place them in the context of the characteristics of battlefield networks supporting a collaborative environment. A network management approach based entirely on resource reservation is especially difficult in this environment since disturbances and unanticipated events often occur with network-based applications.

Due to the inadequacy of conventional QoS approaches, this report explores the concept of network awareness for maintaining distance interaction in a collaborative environment subject to battlefield conditions. In particular, we propose a platform-independent event-delivery framework to facilitate application adaptation, and demonstrate the advantages of this approach in supporting the proactive management applications.

In many cases, an end-user has no control over the network resources, but can monitor the network QoS via appropriate middleware. The end-user can therefore adapt the application to provide the best performance possible while the network quality is varying or deteriorating.

Authors

T. Andrew Au
Command and Control Division

Andrew Au is a Senior Research Scientist in Headquarters Systems Experiementation Group with interests in performance issues in distributed computing environments. He has been involved in research into the ATM signalling and communications issues related to defence networks.

Cindy Tran
Command and Control Division

Cindy Tran was an industry-based learning student from the University of Technology, Sydney, who was assigned to undertake a research project on robust distributed environments in Headquarters Systems Experiementation Group in 2002.

Contents

1. Introduction

The rapid development of information and communication technology has brought a new level of collaboration and communication to the battlefield [1]. Reachback is the concept whereby the forward elements seek support from the rear to increase the survivability of the command function through increased redundancy, manoeuvrability and lower signature. The basic notion of reachback is to minimise the footprint of forces deployed in the area of operation and takes advantage of substantial information resources available at a homeland headquarters.

An essential part of the fabric of military operations is the information systems in the headquarters, providing the critical capability to collect, process, and distribute relevant data to remote locations, thus allowing the military force to gain dominant battlefield situation awareness. However, deployed headquarters are often located in a tactical environment that is generally characterised by volatile network connectivity due to mobility demands and hostile attacks [2]. The wireless communications links often suffer from signal attenuation due to Doppler effects and spatial fading, as well as vulnerability to weather, electromagnetic interference, interception and jamming [3]. The consequences of these phenomena are poor use of bandwidth, high latency (delay) and high error rate. These extremely unfavourable conditions degrade the capability to offer high-quality services to network applications. In particular, the viability of collaborative applications necessitates network Quality of Service (QoS).

In the following we describe the conventional approaches to QoS and place them in the context of the characteristics of battlefield networks supporting a collaborative environment. This informs later analysis of the inadequacy of conventional QoS approaches, and the description of a new approach. The strength of this method is demonstrated by its implementation for the enhancement of a particular collaborative application.

1.1 Quality of Service

QoS is generally regarded as an end-to-end network application requirement. It imposes bounds and limits on communications requirements such as end-to-end delay, data rate, error rate, and their variances in order to guarantee the performance and stability of the application. The problem of QoS has been studied from different perspectives and at different protocol layers, especially at the logical link layer [4][5]. The current data loss recovery schemes such as Go-Back-N Automatic Repeat Request (ARQ) or Selective Repeat ARQ can solve the problem of data loss at the data link layer [6] at the expense of greater variations in delay. Nevertheless, the timing requirement of real-time traffic can hardly be satisfied in these circumstances. Further, military threats such as intrusion, jamming, or physical destruction are highly synergistic. Also, the effects on a network of increased traffic resulting from a sudden increase in the tempo of combat could lead to rapid degradation of network performance.

QoS support enables a network to provide a certain level of services that are appropriate for different applications or users. At the network layer, Integrated Services (IntServ) is a service framework [7] to enforce QoS within which network resources are explicitly managed by means of resource reservation and admission control for individual data flows. Typically associated with this approach is the Resource ReSerVation Protocol (RSVP) [8], a signalling protocol that applications could request end-to-end per-flow QoS from the network, and indicate QoS requirements and capabilities to peer applications. Differentiated Services is a complementary approach [9] designed for large-scale networks that uses a simple method of classifying individual Internet Protocol (IP) data packets requiring similar QoS into a limited number of service categories based on packet marking and traffic prioritisation.

Management of tactical battlefield networks presents a multitude of challenges due to the unpredictable nature of the links. Network layer QoS mechanisms are designed to deal mainly with congestion loss and queuing effects rather than packet loss due to link errors. A dynamic environment, with potentially moving nodes connected by wireless links, presents additional challenges to providing QoS support to such a network [6]. Thus, the network dynamics can be attributed to variable link characteristics and node movement. Moreover, node movement, including the movement of end systems and routers as intermediate systems, causes network topology changes. This has a major impact on the viability of static QoS mechanisms based purely on resource reservation.

Another source of end-to-end network instability common to both dynamic and fixed environments is variable demand on network resources by end-system applications. The conventional response to variable application demand is based on resource reservation and admission control. Rather than providing an unacceptable QoS to all users, it is generally better to deny service to some users in order to grant a reasonable service to others [6]. In order to appropriately perform admission control, allocate resources, and perform other functions necessary for providing QoS, the network layer needs updated information from the link layer on the effective data rate of each interface, as well as possibly other parameters regarding the transmission quality. Further, periodic mission re-assignments can occur and may also result in frequent traffic pattern shift [2]. It may induce synchronisation of massive traffic demand, which can rapidly bring the network to a standstill.

It is clear that a system that can adapt to the dynamic network conditions is extremely important, preferably without human intervention, or, with the least amount of human intervention possible. Such a system, if feasible, will be of utmost benefit to the effectiveness of a collaborative environment.

1.2 Collaborative Environment

A collaborative environment is designed to provide integrated audio, video, document, data and application sharing. It is the capability of sharing and collaborating on various forms of electronic information and data that makes collaborative environments valuable and attractive. Collaborative environments provide comprehensive mechanisms that remove many of the potential obstacles of collaboration by offering a variety of interactive and visual tools to create, capture, present, and communicate information in the most effective way possible. Often, commercial collaborative tools are developed for moderately capable networks where bandwidth is not a substantial problem. A more challenging scenario is that of coordination between deployable headquarters in which the collaborative tool is critical during the conduct of initial planning of the deployment.

Collaboration and conferencing are not the same activities, in that collaboration need not be synchronous. Email is a simple collaboration tool that allows users to asynchronously exchange information, whereas conferencing involves all participants at the same time who are geographically separate. A key aspect of collaborative work is establishing common ground by sharing information through any given communication channel or media so that the task can be accomplished. Olson and Olson [10] argued that the quality of group work via both video and audio is not significantly different from similar group working with audio only. US Navy examined two independent variables (voice and a collaborative tool) relating to the performance measure of a situation assessment task that required collaboration among players [11]. The lack of significant differences on the quality and consistency scores indicates that voice, the collaborative tool, or both are broadly similar in information effectiveness. Each experimental condition provided adequate means to transmit tactical information to the commander.

Based on these findings, exploiting any functional communication channels for information sharing seems to be a feasible approach to maintaining collaboration among members in the face of network transport problems. For instance, a video conferencing session can be gracefully downgraded to a voice connection if the required bandwidth is not sufficient, or simply to an email conversation to keep the collaboration going. Another advanced solution is to place a format conversion server in the network at a strategic point to offer media conversion (eg, voice to email).

1.3 Contribution of this Report

This report explores the concept of network awareness for maintaining distance interaction in a collaborative environment subject to battlefield conditions. In many cases, an end-user has no control over the network resources, but can monitor the network QoS via appropriate middleware. The end-user can therefore adapt the application to provide the best performance possible while the network quality is varying or deteriorating.

Partial failure is a central reality of distributed computing, where one component can fail while the others continue [12]. Further, large-scale networks are often limited in full end-to-end QoS capability due to the diversity of networking technologies. It is likely that these networks have somewhat piecemeal QoS capability. To understand the effect of such partial support of resource reservation, the degraded performance of a QoS-enabled application under different network conditions is demonstrated in Section 2. This is done by emulating the varying degree of resource reservation coverage between a source and a destination. We conclude that the conventional resource reservation approach to enhancing application QoS does not offer a significant advantage in the context of a realistic military network.

Section 3 describes an architecture for exporting network awareness using an event-based approach, suitable for changing network conditions. This enables an application to dynamically adjust the functionality and resource usage. Through cognisance of the network capability, applications are able to proactively perform their functions with greater effectiveness. Policies can also be applied to the application domain, to adapt to changing environmental conditions for optimal performance.

Section 4 demonstrates the advantages of this architecture through the implementation of network awareness capability to enhance Microsoft NetMeeting. With user-defined policies, the enduring performance reveals the benefits of the adaptive QoS approach in a dynamic environment.

Thus, a new architecture to support the proactive management of application is described and its advantages demonstrated. Section 5 concludes the report.

2. Effects of QoS in an Overloaded Network

Simplicity is one reason for the tremendous success of the Internet where intelligence is only placed in the source and destination hosts connected over a simplistic network core. With many applications sharing the same network at one time, transient congestion is often the result, causing delivery delays or even packet loss. This is not a problem for elastic applications like email or file transfer. But delays can be fatal to mission-critical applications, which may significantly limit the usability.

QoS characteristics are certain independent quantifiable aspects of a service. The four most important network parameters for the effective transport of network traffic are bandwidth, delay, jitter, and packet loss. The Internet transmits data on a best-effort basis and is unable to provide any sort of network resources guarantee. In a QoS-enabled network, applications that need constant, high levels of bandwidth and low levels of latency, jitter, and loss can request and receive a guarantee for those network resources. A number of traffic handling mechanisms are possible to offer applications a

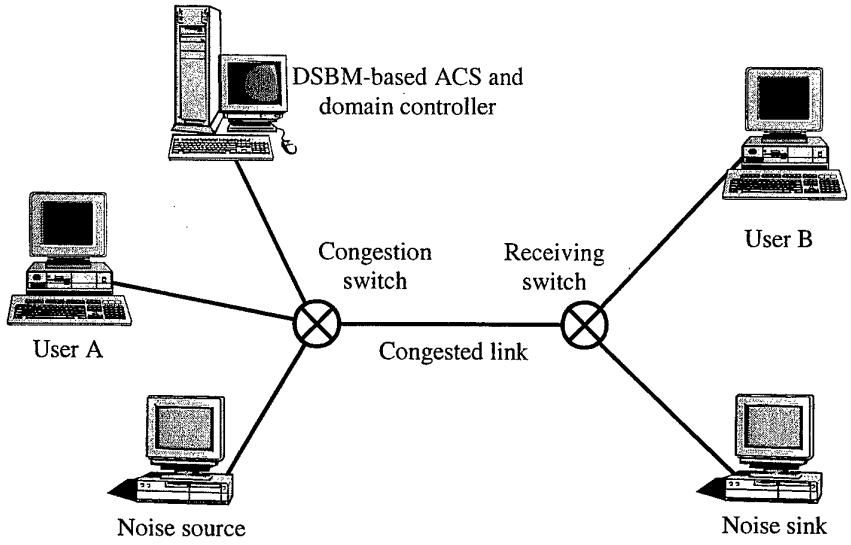


Figure 1: Configuration for the demonstration

certain level of guarantee, each of which is appropriate for specific media or circumstances. Appendix A describes further detail of network QoS mechanisms.

Microsoft Windows 2000 contains a suite of QoS components supporting several QoS mechanisms, including the RSVP, DiffServ, IEEE 802.1p, and ATM. QoS-aware applications can invoke these services to provide RSVP signalling, Subnet Bandwidth Manager (SBM) client functionality, QoS policy support, and invocation of traffic control. The latter includes marking, packet scheduling and other media-specific behaviour. As a standard part of the Windows 2000, Windows 2000 QoS can be used to accomplish prioritisation and guarantee of bandwidth for latency-sensitive or high priority network traffic, such as video conferencing. NetMeeting is one of the applications that are currently enabled to use the QoS components (See Appendix B for more detail).

2.1 Verifying QoS Operation Using NetMeeting

Figure 1 illustrates a scenario [13] that is configured to demonstrate the QoS operation using NetMeeting. The QoS Admission Control Service (ACS) is installed on a server running Windows 2000 Server. It acts as a policy configuration point for the network administrator. It also serves as a domain controller. Its function is to provide a control point for bandwidth requests for QoS users, and determines if the necessary network resources are currently available, and whether or not the user has the necessary permissions to request that amount of bandwidth. All the other computers are running Windows 2000 Professional, to allow for complete support of Windows 2000 QoS. NetMeeting is installed on Users A and B as the video conferencing application. This

scenario is typical of a NetMeeting deployment, where more bandwidth is usually available on LANs as opposed to WAN links.

The noise sink pulls noise from the source through the network to simulate network congestion. The congestion switch is the congestion point, which is 802.1p capable. This allows for the guaranteed reservation of the required bandwidth at layer 2 from User A to User B. The receiving switch is just a normal Ethernet switch, which is not necessarily 802.1p capable. All transmission links are configured to run at 100Mbps except the congested link, which is limited to 10Mbps sufficient to cause congestion at outbound interface of the congestion switch.

User A starts a NetMeeting session with User B. This triggers the RSVP service provider to send an RSVP PATH message on behalf of the sending application. Because the ACS is a Designated SBM (DSBM) on the local subnet to which the sender is attached, the message will be directed to the DSBM. The ACS verifies that enough network resources are available to meet the QoS level requested, and that the user policy allows for the requested amount of bandwidth. After verification is complete, the ACS approves the request and logically allocates bandwidth, and forwards the request through the switches toward User B. After the request arrives, User B indicates that it wants to receive the data and returns an RSVP RESV message requesting a reservation. This message passes through both switches and arrives at the DSBM, where it is approved and forwarded onto User A.

The exchange of RSVP messages actually establishes RSVP state along the route between the sender and receiver. Layer 3 RSVP-enabled devices are capable of approving and allocating the bandwidth. Upon RSVP request, any RSVP-enabled switches along the route, if present, will keep track of the resources that are requested. Each of these switches will match the receiver's response with the sender's request, and installs the reservation by granting the bandwidth. This reservation however simply passes through any layer 2 devices.

Upon receiving the reservation message, the sending QoS service provider changes the service type for the audio flow from best effort to Guaranteed, and for the video flow from best effort to Controlled Load. The traffic control on User A begins processing of marking the packets. The QoS Packet Scheduler performs the marking for both layer 2 and layer 3 devices – the 802.1p marking for prioritisation on layer 2 devices, and DiffServ marking for layer 3 devices. The tagged packets subsequently get priority in the congestion switch while waiting in the transmitting queue for the congested link. Hence video and audio quality improves after the reservation is set up.

Figure 2 shows the video output at User B in a NetMeeting session over the congested link without QoS reservation. Since User A's traffic has to compete with the noise from the source to the sink, congestion builds up at the 10Mbps link between the two switches. Only a small number of NetMeeting packets actually arrive at User B,

Figure 2: NetMeeting without QoS reservation

Figure 3: NetMeeting with QoS reservation

resulting in poor video quality. Figure 3 shows the same video output at User B in a NetMeeting session over the congested link with QoS reservation.

2.2 Effect of Partial Resource Reservation

QoS requires the cooperation of all logical layers in a network, from the application layer down to physical media and of all network elements. In particular, guarantee service assumes that every router supports resource reservation. Unfortunately, support for QoS is not always available on every element of a large-scale network. Those nodes not supporting resource reservation may simply ignore QoS requests, or even reroute packets. An example is the Internet where support for resource reservation on every network element is simply not feasible. Even in a less complex but large-scale network, it is difficult to ensure full support of resource guarantees in every network device. Besides, such a network is extremely vulnerable to mismatch of configurations [14], which could cause serious outages or performance degradation. In large-scale networks, an approach based entirely on resource reservation cannot guarantee end-to-end quality service, leaving the distributed applications in limbo.

Figure 4 illustrates an experiment to demonstrate the effect of partial QoS support in a Windows 2000 network. Various network conditions were simulated using the Cloud WAN Emulator [15]. We used *qtcp*[1] to measure end-to-end network service quality between the signal sender and the signal receiver. The noise generated across the network competes with the *qtcp* session for network resources, creating an overload of 50%.

In the experiment, resource reservation is only effective in the local networks at both the source and destination ends. The simulated link is characterised by the degree of packet loss from 0% to 10% to cover the range expected in the Internet [16] where the

[1] Qtcp measures end-to-end network integrity and service quality for QoS verification. Qtcp sends a sequence of test packets through a network and reports on the queuing delay experienced by each packet.

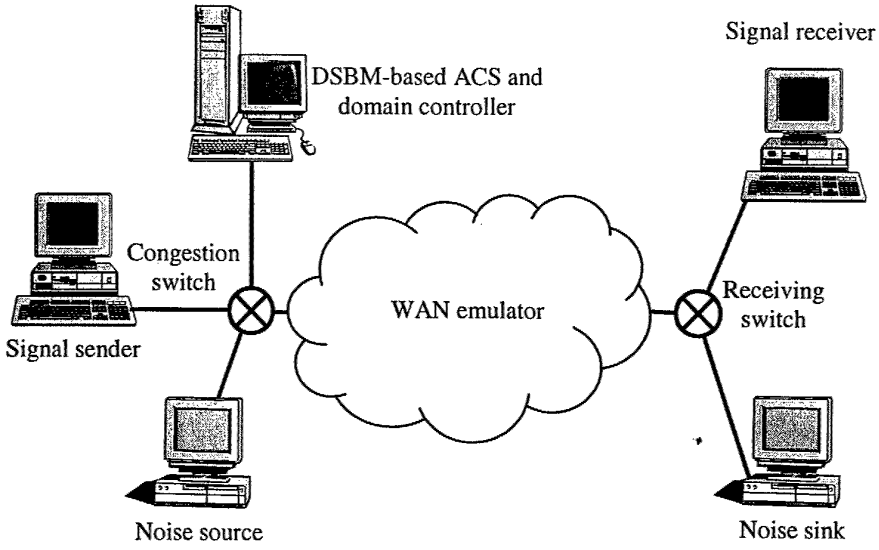


Figure 4: The experimental environment

packet loss can be attributed to congestion, bit errors, and deliberate discard. Packet discard due to corrupted data is more likely on wireless and satellite links because of poorer link quality. Figure 5 plots the 1-hop throughput versus packet loss, with and without QoS reservation. Extrapolating from this 1-hop measurement, Figure 5 also shows the throughputs for 2-hop and 3-hop links.

The throughputs are gradually reduced as the congestion of the simulated link increases. The number of hops has a more significant effect on the throughput of the QoS-enabled flow than that of the best-effort flow. Under serious congestion, the benefit of QoS reservation diminishes as the QoS-enabled flow goes along more congested hops. Even though best-effort traffic is constantly worse off under severe congestion, QoS-enabled flows are not particularly well treated. This experiment demonstrates that the performance difference between QoS-enabled and best-effort flows is less significant if QoS reservation is not recognised by most of the hops. Given that few routers in the Internet allow QoS reservation, it is by and large a best-effort network over which QoS-enabled applications are not privileged to receive priority treatment in the face of congestion.

3. An Architecture for Exporting Network Awareness in Support of Application Adaptation

In a volatile environment, an application needs to be aware of current resource availability, so as to dynamically upgrade or gracefully degrade in response. An

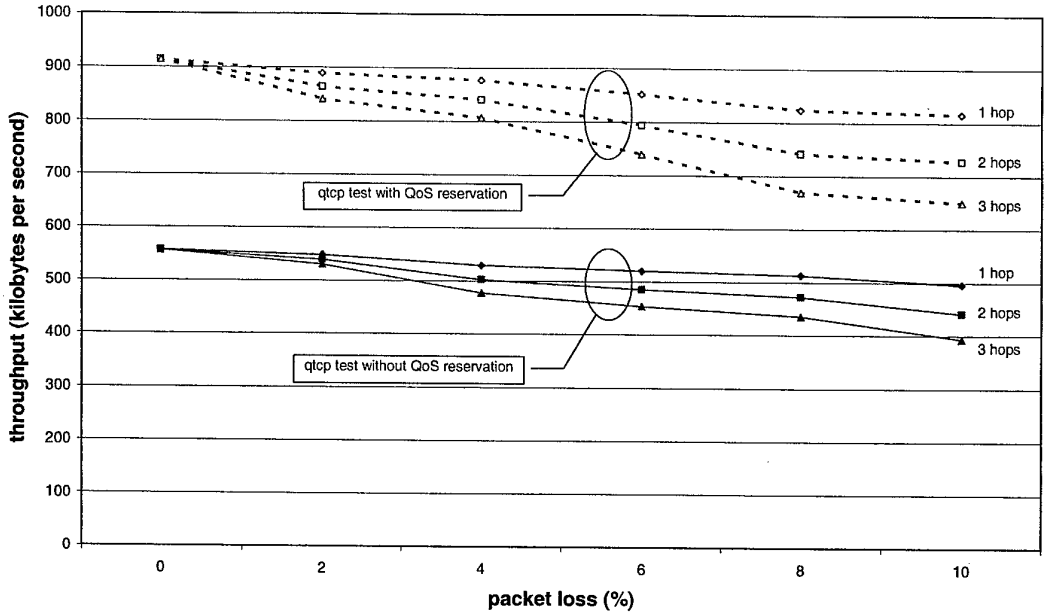


Figure 5: qtcp measurement with and without QoS reservation

architecture is required for exporting awareness of the computing and communication environment to an application, usually in terms of asynchronous events. To be network aware, an application must be notified of changes in the network state and the associated information. For instance, on detecting a current network change, an application may need to estimate the network parameters for self-adjustment like typical latency and bandwidth. Such parameters are especially useful with intermittent network connectivity of deployed headquarters. Further, network awareness can be found useful when an application is first activated. The latest network state can help select the appropriate modules for an optimal configuration.

Network detection and adaptation strategies can be constructed directly at the application level, but such ad hoc implementation restricts its reusability by other applications. It also violates separation of concern between functional and environment-specific aspects, and leads to unwieldy and custom code. A powerful means to achieve application portability is to encapsulate platform peculiarities in middleware runtime support, which lies between the application and system resources. Integration of an adaptation mechanism with the middleware brings the awareness and control closer to the application and at the same time, functional aspects of the application are kept separate from the environmental peculiarities.

3.1 The Event Delivery Framework

Figure 6 illustrates a loosely coupled mechanism where a change in the network state is modelled as an asynchronous event. Information is advertised and delivered to interested parties who subscribe to particular events, without prior knowledge of their

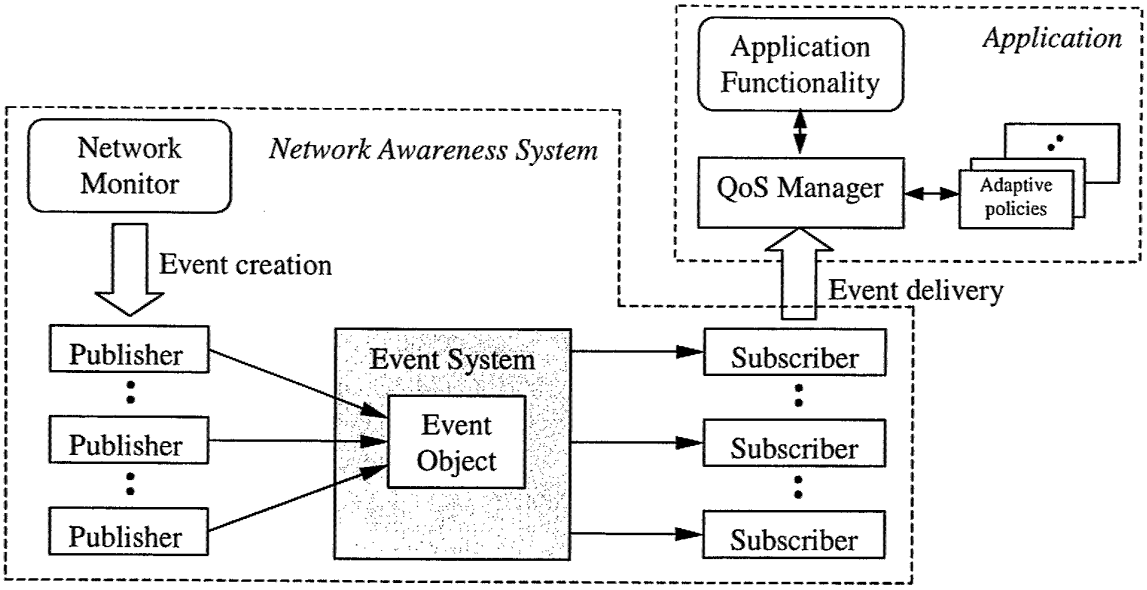


Figure 6: Architecture for network awareness

identities. An event object encapsulates information related to the occurrence of a network change. Upon publication of an event, the event system notifies any subscribers who have registered to receive notification of that event. Each application can implement appropriate adaptive behaviour based on the occurrence of specific events. Examples are events associated with the battery status and the network connectivity in mobile computers [17].

3.2 Adaptive Applications

Network awareness facilitates the development of adaptive applications. By definition, an adaptive application is able to operate at different operating points during its lifetime, each having different resource requirements. Ultimately the functional constraints have to make decisions on which service-level is functionally possible. Performance sensitivities of an application determine what adaptation algorithm is most appropriate. For example, video streams can be degraded through reduction of different parameters [18]: the frame rate, the quality of individual frames, the size of individual frames or colour components. Whenever a received event indicates that resource conditions have changed, the QoS Manager is able to dynamically reconfigure the application functionalities to match available network resources according to user-defined policies for adaptation [19].

3.3 The Network Monitor

The network monitor performs certain tests on the quality of network service. It consists of individual software modules that can conduct measurement of network conditions, record statistics such as latency, available bandwidth, and round trip time

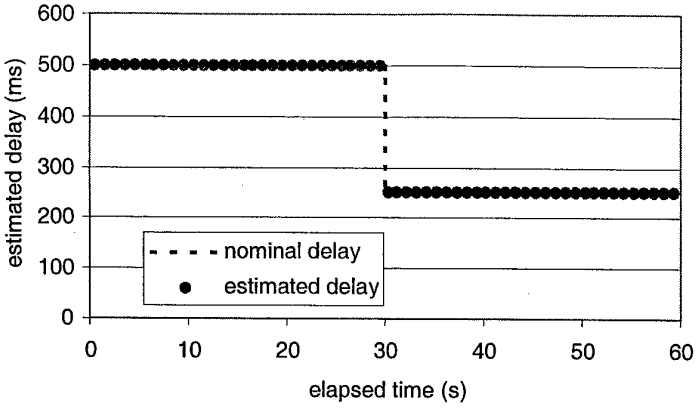


Figure 7: Detecting decreased delay

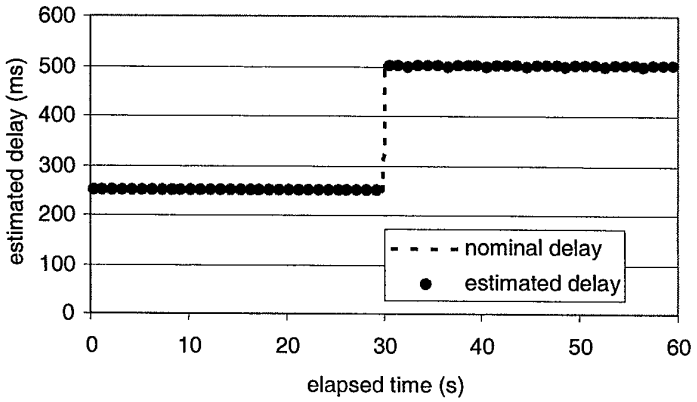


Figure 8: Detecting increased delay

(RTT). Actual available bandwidth or throughput is frequently used as a realistic estimate of the real network performance that one can achieve. It is typically measured by timing the transfer of a large bulk of data. Poor throughput indicates network congestion but it depends also on the configuration of the transport protocol, such as TCP window size.

There are numerous useful network-monitoring tools [20] to determine link state, functionality and capacity. Several of these tools are described in Appendix C. In our experiment, we selected *ping* and *iperf* but these components can be replaced without affecting the rest of the architecture. *Ping* can measure point-to-point RTT and estimate packet loss. Assuming symmetric paths, the one-way point-to-point latency is half of the RTT. *Iperf* [21] calculates available bandwidth using the amount of bytes received and the time it takes to receive all the packets. We configure *iperf* to measure the TCP throughput as the available bandwidth, and to produce the datagram loss of UDP, which is taken as the measure of packet loss.

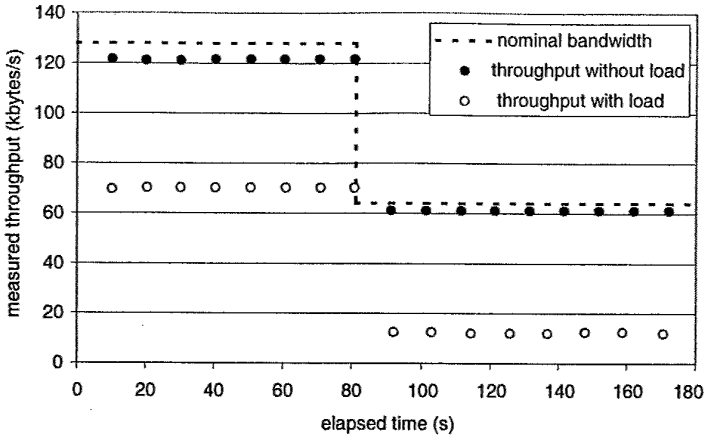


Figure 9: Detecting decreased throughput

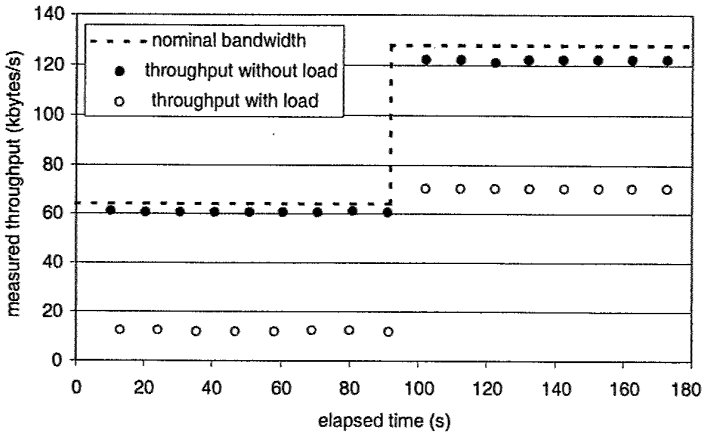


Figure 10: Detecting increased throughput

3.4 Estimation of Network Quality

The network monitor must be able to faithfully react to true changes in the network. To determine its accuracy, we created a testbed in which we can emulate changing link characteristics. We subjected it to various network conditions simulated using the NIST Net Emulation Package [22]. Similar to the control-theoretic technique of impulse response analysis [23], we synthetically generate two waveforms for each parameter of interest. The first waveform is called step-down that instantaneously decreases an initial value halfway through the trace. The second is called step-up that increases the value.

The results for step-down and step-up waveforms of the RTT are shown in Figure 7 and Figure 8, respectively. Since all estimates observed by *ping* match accurately with the nominal values, it confirms that increases and decreases in delay are easily detectable. The results for step-down and step-up waveforms of the available bandwidth are shown in Figure 9 and Figure 10, respectively. At no load, *iperf* correctly measures the available bandwidth with some discrepancy possibly attributed to

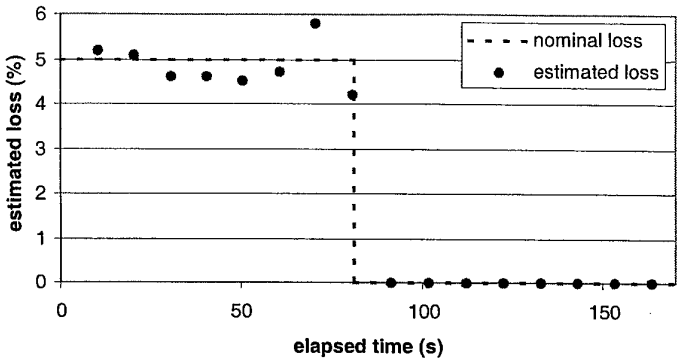


Figure 11: Detecting decreased packet loss

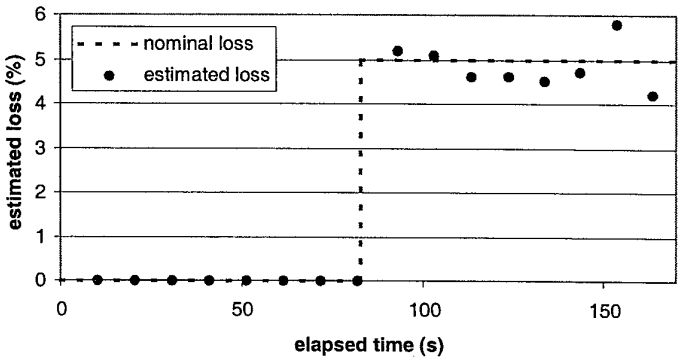


Figure 12: Detecting increased packet loss

overheads for Windows 2000 network. When an artificial load of 51kbytes/s is introduced, *iperf* is able to observe changes in available bandwidth. It confirms that increases and decreases in available bandwidth are detectable. Next, the results for step-down and step-up waveforms of the packet loss are shown in Figure 11 and Figure 12, respectively, at 1 Mbyte of UDP test messages. This estimate is considered reasonably adequate in detecting increases and decreases in packet loss. The test requires a series of UDP datagrams and its accuracy depends on the message length. A better estimate results at the expense of measurement overheads and turnaround time.

3.5 Implementation Details

The event system is implemented based on COM+[2] Event service where the publishers and subscribers are registered components. A publisher fires an event by calling a method on the event object. COM+ Event service then notifies the subscribers. Appendix D discusses the three methods of distributing COM+ Events. Persistent subscription to the COM+ Event service is required since it can withstand system reboots and is not bounded by the lifetime of the subscriber application. Each

[2] COM+ provides enhancements to the Microsoft Component Object Model (COM), which enables programmers to develop COM objects more easily. COM+ is the next logical progression of COM and Microsoft Transaction Server (MTS), fused into one seamless suite of services integrated into Windows 2000.

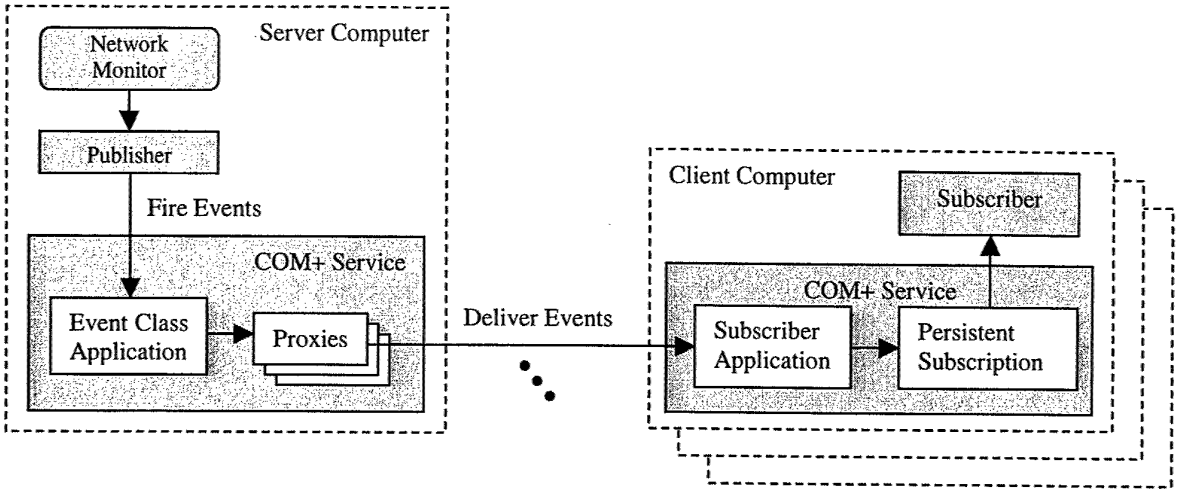


Figure 13: Implementation of multiple persistent subscriptions

persistent subscriber resides in the COM+ database and is built as a dynamic link library containing a class that implements the interface of the event (See Figure 13). When new estimates are available from the network monitor, the publisher fires an event to the COM+ service in the server computer. The COM+ service then delivers the event to each subscription through its proxy. Once an event arrives at a client computer, the COM+ service loads and instantiates the subscriber, then the event is delivered to it. This event is recorded into the registry in the client computer for future reference and becomes available for use should an application require. If an application is already active, the event is also passed to the QoS Manager (See Figure 6).

4. Enhancing NetMeeting with Network Awareness

Collaborative environment is an important application in the battlefield. The knowledge of network conditions allows the users to exploit the use of resources rather than aggravate the network problem with aggressive re-tries. For demonstration, we chose to enhance Microsoft NetMeeting with the capability of network awareness. Depending on the prevailing network conditions, users are advised to appropriately alter the functionalities of NetMeeting for optimal performance. The network monitor component on the server computer implements *ping* and *iperf* to retrieve the network conditions of a destination. *Ping* returns the RTT, while *iperf* calculates the available bandwidth based on TCP. *Iperf* also estimates the loss using the feature of UDP. These measurements are collected as an event to be fired off by the publisher on the Event Service. The subscribers receive the event regularly and are informed of any changes. These results are written into the registry and a message is displayed to advise the user. According to the network conditions and the QoS policies, the user is advised to take adaptive actions if required. Figure 14 shows the user interface containing four entities: a ListView to display network conditions, an ActiveX Control NetMeeting, a

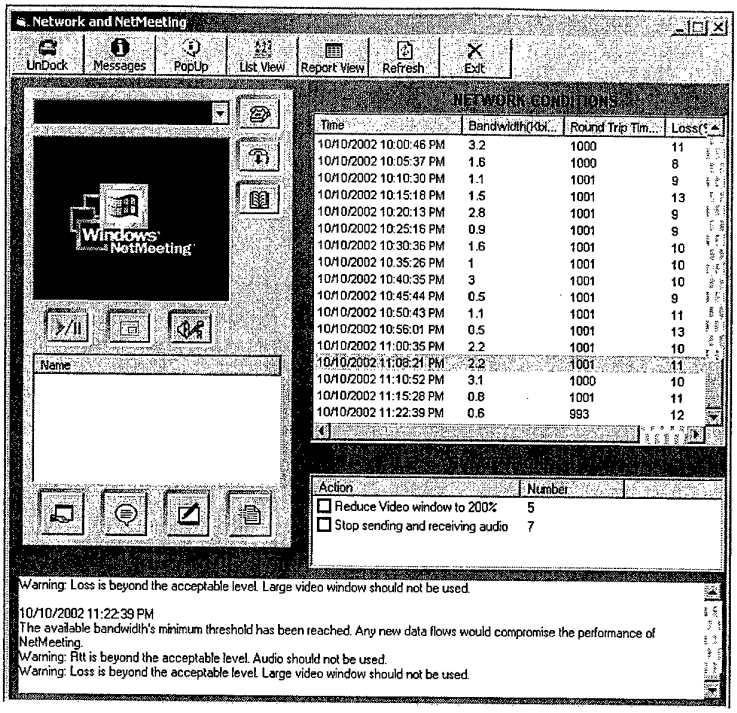


Figure 14: User interface to NetMeeting for network awareness

suggestion box to recommend possible actions, and a log to record all messages and actions.

4.1 QoS Manager

The QoS Manager receives information about network conditions (available bandwidth, delay, and packet loss), and in consultation with QoS policies, advises NetMeeting users to take particular actions. In the face of extreme network constraints, it helps achieve robust operation through adaptive self-configuration in an ongoing dynamic balance.

Enhancing an application with adaptation may also compromise its user-friendliness. Hence, we implemented the adaptive decisions in a semi-automatic fashion so that the user is always consulted before an action is taken. Messages about network conditions are displayed, providing warnings to users and suggesting an optimised configuration.

The drawback of a dynamic application is information overload to users. It may become another burden if the user interface is not well designed. A better approach is using an intelligent agent (such as Microsoft Office Assistant), who is able to assist the user about all the network applications within the operating environment. The agent can be set to operate so that it meshes with the way the user works. Users will automatically get suggestions on how to configure the applications for optimal performance within the network constraints.

4.2 QoS Policies

The QoS policies capture the rules for network adaptation that dictate optimal modes of operation for any given environmental conditions and user requirements. These are versatile user preferences for usage monitoring and application configuration. We implemented a set of lightweight policies to allow self-configuration when NetMeeting is started, and to allow automatic reconfiguration when NetMeeting is already running.

The measure of available bandwidth reveals the current network usage, indicating a degree of possible congestion. The rules for bandwidth adaptation were set according to reference measurements [24]. Users are able to predict whether it is feasible to start NetMeeting or to add a new information flow (voice, video, whiteboard, chat, application sharing, or file transfer). When NetMeeting is first activated, the latest network state is retrieved from the registry that is used to advise the user of the optimal configuration of NetMeeting. When a new information flow is initiated in a bandwidth-constrained environment, the user is advised of the possibility of network congestion.

From the QoS point of view, timing parameters of interactive applications are sometimes more important than the data loss ratio. When the round-trip delay is more than 50 ms, echo becomes a problem without echo cancellation [25]. When the one-way delay is greater than 250 ms, talker overlap is made worse [25]. One-way delay of more than 400 ms is unacceptable for voice [26], whereas video is generally regarded as a supplementary medium in video conferencing. Our policy enables voice only if the RTT is less than 800 ms. Packet losses greater than 10% for audio and video data are generally intolerable [25] in multimedia applications unless the encoding scheme implemented provides extraordinary robustness. Our loss policy enables voice only if loss is less than 10%. With these policies, the enhanced NetMeeting demonstrated its enduring performance compared to the standard version, revealing the benefits of the adaptive QoS approach in a dynamic environment.

5. Conclusions

The success of Reachback depends on seamless data flow from the forward location through the entire support pipeline. It is vital that deployed elements are able to reach back to the rear elements for the follow-on support necessary to conduct sustained operations. If the information systems do not work properly, failure of communication and information services may result in loss of life.

To facilitate adaptation in an unpredictable battlefield environment, we have proposed a platform-independent event delivery framework so that collaborative applications

and their users can be made aware of changing network conditions. As middleware, it allows concerns about the application functionality and adaptation to be kept conceptually separate from concerns about network awareness. For example, a video conferencing session can be gracefully downgraded to a voice connection if the required bandwidth is not sufficient, or simply to a message conversation to keep the collaboration going.

In the application domain, policy-based management can optimally adapt applications to changing environmental conditions. According to user preferences and application specific policies, adaptive applications are able to adjust their behaviours appropriately through cognisance of the communications capability, and perform their intended functions with greater effectiveness. We chose to enhance NetMeeting to demonstrate the viability and salient features of the proposed design. We have evaluated our implementation in an emulated network environment. Depending on the prevailing network conditions, users are advised to appropriately configure NetMeeting for optimal performance. The results have shown that this approach can be an important component in graceful performance degradation and in maintaining network-based system availability.

In summary, we have described and shown promising results on the use of proactive adaptation by the end-users in response to poor and variable network quality. With adaptive applications, users with little or no formal training can control their own use of available resources. Knowing the prevailing network conditions, users can help prevent further network deterioration. For better usability, an intelligent agent can be configured to handle all system information and to provide only timely advice as necessary to the users.

References

1 B. T. Robinson, "Who Goes There?" IEEE Spectrum, October 2003.

2 L. Kant, C.-H. Zhu, D.K. Hsing, and M. Lee, "Design of an Adaptive Configuration Management Architecture for Tactical Battlefield Networks," Proceedings of IEEE 2000 Conference on Military Communications, MILCOM 2000.

3 H. Zheng, S. Wang, and J. A. Copeland, "QOS Aware Mobile Video Communications," Proceedings of IEEE 1999 Conference on Military Communications, MILCOM 1999.

4 A. Ohta, M. Yoshioka, and T. Sugiyama, "PRIME ARQ: A Novel ARQ Scheme for High-Speed Wireless ATM – Design, implementation and performance evaluation," Proceedings of IEEE 1998 Vehicular Technology Conference, VTC 1998, pp. 1128-1134.

5 O. Nakamura, H. Matsuki, T. Oono, and T. Tanaka, "Pre-Repeat Selective-Repeat ARQ in Fading Channel," Proceedings of IEEE 1998 International Conference on Universal Personal Communications, 5-9 October 1998, Florence, Italy, Vol. 2, pp. 1253-1257.

6 M. Mirhakkak, N. Schult, D. Thomson, "A New Approach for Providing Quality-of-Service in a Dynamic Network Environment," Proceedings of IEEE 2000 Conference on Military Communications, MILCOM 2000.
7 R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture," IETF, RFC 1633, June 1994.
8 R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, "Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," IETF, RFC 2205, September 1997.
9 S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "An Architecture for Differentiated Services," IETF, RFC 2475, December 1998.
10 G.M. Olson, J.S. Olson, "Technology Support for Collaborative Workgroups," in Coordination Theory and Collaboration Technology, edited by G.M. Olson, T.W. Malone, and J.B. Smith, Lawrence Erlbaum Associates, 2001.
11 "Review and Recommendations for Development and Experimentation with Collaborative Environments," TTCP C3I TP-10 Report, The Technical Cooperation Program, US Air Force Research Laboratory, Jan 2000.
12 J. Waldo, G. Wyant, A. Wollrath, S. Kendall, "A Note on Distributed Computing," TR-94-29, Sun Microsystems Laboratories, Nov 1994.
13 Y. Bernet, "Networking Quality of Service and Windows Operating Systems," New Riders Publishing and Microsoft Corporation, 2001.
14 L. Kant, D. Hsing, and T.-H. Wu, "Self-Configuration Management System Design for Tactical Battlefield Networks," Proceedings of IEEE 1998 Conference on Military Communications, MILCOM 1998.
15 The Cloud WAN Emulator User Guide, Shunra Software Ltd, 2000.
16 Internet Traffic Report, available at http://internettrafficreport.com/main.htm
17 G. Welling, and B.R. Badrinath, "An Architecture for Exporting Environment Awareness to Mobile Computing Applications," IEEE Transactions on Software Engineering, Vol. 24, No. 5, May 1998, pp. 391-400.
18 B. Noble, "System Support for Mobile, Adaptive Applications," IEEE Personal Communications, Feb 2000, pp. 44-49.
19 V. Witana, M. Fry, M. Antoniades, "A Software Framework for Application Level QoS Management," 1999 Seventh International Workshop on Quality of Service, IWQoS'99, IEEE, 1999, pp. 52-61.
20 Network Monitoring Tools, available at http://www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html
21 Iperf 1.7.0 – The TCP/UDP Bandwidth Measurement Tool, available at http://dast.nlanr.net/Projects/Iperf/
22 NIST Net, available at http://snad.ncsl.nist.gov/itg/nistnet/
23 B. Noble, "System Support for Mobile, Adaptive Applications," IEEE Personal Communications, Feb 2000, pp. 44-49.
24 Microsoft Windows NetMeeting 3 Resource Kit, Microsoft Corporation, 2001.
25 P. Mehta, S. Udani, "Voice over IP: Sounding good on the Internet," IEEE Potentials, vol 20, issue 4, October/November 2001, pp.36-40.
26 ITU-T Recommendation G.114: Transmission Impairments; Transmission Delay.

Appendix A: Network QoS Mechanisms

This appendix discusses a number of traffic handling mechanisms possible to offer applications a certain level of guarantee, each of which is appropriate for specific media or circumstances.

A.1. Integrated Services (IntServ)

IntServ is a service framework to enforce QoS within which network resources are explicitly managed by means of resource reservation and admission control. An IntServ-capable network provides services to accommodate requirements of individual traffic flows as well as link-sharing requirements for aggregations of traffic flows. Two services are defined to provide quantifiable QoS to a specific quantity of traffic: the Guaranteed Service and the Controlled Load Service. The Guaranteed Service promises to carry a certain traffic volume with a quantifiable, bounded latency. The Controlled Load Service agrees to carry a certain traffic volume with the appearance of a lightly loaded network.

When a traffic flow is to be established in an IntServ-capable network, a set of QoS parameters is passed to the network, including bandwidth, packet delay and packet loss. The network tries to reserve sufficient resources in routers in order to satisfy the requirements. IntServ is typically (but not necessarily) associated with the resource ReSerVation Protocol (RSVP). The RSVP is a layer-3 protocol by which applications can request end-to-end, per-flow, QoS from the network, and indicate QoS requirements and capabilities to peer applications. The RSVP serves to invoke and to coordinate the admission control and reservation set-up processes in individual network elements along the traffic path of a flow. Transmitting hosts send RSVP messages through the network to receiving hosts, which respond by sending resource requests back toward the transmitters. The RSVP is required to periodically update path states in all the routers along the path of the flow. The reservation remains valid until an application explicitly requests its termination or the network signals the application that it is not able to maintain the reservation.

IntServ allows per-flow management, in which each router perceives the individual traffic flow and allocates network resources to each flow. However, this per-flow approach has problems with scalability. To establish IntServ on a network, core backbone routers need to manage a huge amount of flows simultaneously. In large networks, simpler, less fine-tuned methods are needed, and that is what DiffServ provides.

IntServ emulates the resource allocation concept of circuit switching to apportion resources according to requests each host makes.

A.2. Differentiated Services (DiffServ)

IntServ is a general approach to QoS on any sort of network; DiffServ is an approach for large-scale networks. As opposed to IntServ, DiffServ uses a simple method of classifying packets requiring similar QoS into a limited number of service categories, so as to eliminate the overhead associated with fine-grain, per-flow traffic handling in the core of an IntServ-capable network. Based on packet marking and traffic prioritisation, DiffServ aggregates flows that require similar preferential treatment at the core backbone routers.

DiffServ is a layer-3 QoS mechanism that defines a field in the layer-3 header of IP packets, called the DiffServ CodePoint (DSCP)[3]. Typically, hosts or routers sending traffic into a DiffServ network will mark each transmitted packet with the appropriate DSCP. Routers within the DiffServ network use the DSCP to classify packets and apply specific queuing or scheduling behaviour (known as per-hop behaviour or PHB) based on the results of classification. The two PHB types are Expedited Forwarding (EF) and Assured Forwarding (AF). The EF PHB provides high-quality, low-latency service, whereas the AF PHB provides a level of service lower than EF PHB but better than the present best-effort Internet service.

PHBs are individual behaviours applied at each router, which alone make no guarantees of end-to-end QoS. However, by concatenating routers with the same PHBs and with careful admission control, it is possible to use PHBs to construct an end-to-end QoS service. IntServ and DiffServ are complementary technologies in the pursuit of end-to-end QoS. Whilst IntServ offers fine-grain guarantees for quantitative applications, DiffServ offers a simple model for coarse service differentiation to address the needs of qualitative applications.

DiffServ classifies per-hop behaviours on the basis of a DSCP attached to the type of service byte in each packet's IP header. The DSCP approach represents a form of soft QoS that rather coarsely classifies services through packet marking. The expedited forwarding (EF) class minimises delay and jitter and provides the highest level of aggregate QoS.

A.3. IEEE 802.1p

The IEEE 802.1p is a specification that works in tandem with 802.1Q specification for VLAN tagging. It defines a three-bit field in the layer-2 header of IEEE 802 frames for prioritisation, which allows frames to be grouped into eight traffic classes. Typically, hosts or routers sending traffic into a local area network (LAN) may mark each transmitted frame with an appropriate priority value. LAN devices, such as switches, bridges and hubs, are expected to treat the frames accordingly by making use of underlying queuing mechanisms. The scope of the 802.1p priority mark is limited to

[3] The DSCP is a six-bit field, spanning the fields formerly known as the type-of-service (TOS) fields and the IP precedence fields.

the LAN. Once packets are carried off the LAN, through a layer-3 device, the 802.1p priority is removed.

IEEE 802.1p invokes an aggregate traffic handling mechanism in layer-2 devices in accordance with the requested user priority. This is similar to the DiffServ, which invokes traffic handling mechanisms in layer-3 devices through the DSCP. However, 802.1p traffic handling mechanism is often considered less important than its layer-3 counterpart supporting wide area networks (WANs). With the increasing usage of multimedia applications on LANs, delays through LAN devices do become problematic and 802.1p is ready to tackle these delays.

Appendix B: Microsoft NetMeeting

Microsoft NetMeeting is a collaborative support tool, which has a limited conferencing capability allowing two users at a time to exchange audio and video. It also provides the ability for several users to share a whiteboard, text chat, file transfer, and Windows applications across a network.

NetMeeting creates individual audio, video and data streams for network transmission at their own rates. The H.263 and G.723 standards are used as the default video and audio codecs, respectively. Audio creates a stream at a fairly constant rate when speech is being sent. Video produces a stream at a widely varying rate that depends on motion, quality, and size settings of the video image. Data produces a stream at a widely varying rate that depends on a number of factors, including the use of file transfer, file size, the complexity of a whiteboard session, and the complexity of the graphic and update information of shared programs. However, the chat window provides lightweight chat capabilities, which makes it very easy to start up informal, text-based conversation.

Different types of data for collaborative applications require support for multiple levels of QoS. NetMeeting uses intelligent bandwidth management and control, together with the optimisation of data through compression, caching, and other tools to make best use of the networking resource available [24]. In a connection, the highest priority is given to the audio stream, followed by the data stream, and then the video stream. The nature and type of network connection between conferencing partners can affect the performance. Higher bandwidth allows more video data to flow, resulting in faster and better video quality.

Other NetMeeting processes can affect audio quality and general performance if the network is not able to fully support QoS reservation. For example, sharing several programs while performing a large file transfer and simultaneously running whiteboard and chat, affects available bandwidth for audio. Also, full-duplex audio requires more bandwidth and more processing power than half-duplex audio, which may result in poor voice quality.

Appendix C: Network Monitoring Tools

Network monitoring tools are used to measure network conditions and to record statistics such as latency, bandwidth, loss, and round trip time (RTT). These parameters are essential to gain a reliable measure of network quality. A number of Windows-based tools are listed below that are available from the public domain.

C.1. Ping

Ping is a simple well-known tool to retrieve the RTT and the packet loss. *Ping* uses the Internet Control Message Protocol (ICMP) to send small packets to a destination address, which then replies with other ICMP packets. The RTT is the time taken for the packet to reach the destination and a reply received by the source. The loss is calculated as the proportion of replies that are not returned by the destination address. *Ping* is compatible with any platform and is readily available on every computer. The *ping* tool can be downloaded from http://www.codeproject.com/ website. However, *ping* alone is not sufficient to provide a complete analysis of the entire network, as it does not measure the bandwidth between a source and a destination.

C.2. Traceroute

Traceroute is a similar tool to *Ping* that uses the ICMP to measure the RTT at each hop along the route to the destination. It pings the first hop and retrieves its RTT. It then goes on checking the next hop until the destination is reached. Certainly more ICMP packets are required to travel back and forth between each hop. Thus, *traceroute* generates more network traffic and it takes time to retrieve the results.

C.3. Iperf

Iperf originates in Linux to measure maximum TCP and UDP bandwidth, which is now ported to Windows platform. It requires both a server to run a receiver on one end and a client as the sender to run on the other end. *Iperf* can only use UDP or TCP protocol at a time. Using the reliable TCP protocol, *iperf* establishes a connection and transfers packets of a certain size. The available bandwidth is calculated according to the amount of bytes received and the time taken to receive all the packets. Measurement using unreliable UDP generates a number of parameters: jitter, RTT, loss, and bandwidth. The drawback is using UDP in bandwidth measurement does not truly reflect the existing network conditions due to the aggressiveness nature of UDP. *Iperf* and its documentation can be downloaded from http://dast.nlanr.net/Projects/Iperf/ website. However, suitable source code of *iperf* is not available.

C.4. Windows 2000 performance tool

A system monitor is available in the Windows 2000 Performance tool. It enables users to collect and view real-time data about memory, disk, processor, network, and other activities. The performance counters related to network activities include bandwidth and bytes sent or received per second. However, these statistics are only valid for the local computer and not for the whole network. The bandwidth is actually the one configured on the network card, not reflecting the conditions on the network.

C.5. Bing

Bing is another Linux program ported to Windows that measures the point-to-point total bandwidth based on *ping*. The program can run at any computer. It basically pings the computers of each end of the link with ICMP packets of different sizes. The RTTs for each computer are used to calculate the RTT, loss and bandwidth of the path between the two computers. The latter actually measures the raw bandwidth rather than the available bandwidth. The program can be downloaded from http://www.cnam.fr/reseau/bing.html.

Appendix D: Distribution of COM+ Events

COM+ can be used either locally on a single machine or distributed among several machines. COM+ Events can be distributed in three different ways to be discussed below. The method of distribution has minimal effect on the publishers and the subscribers. The only possible changes in the implementation are updating the Globally Unique Identifier (GUID) in the subscribers and updating each subscriber with a new event class, depending on the method used. Persistent subscriptions have minimal limits on distributing COM+ Events, while transient subscriptions are constrained to two options only.

The methods of distribution are limited by these constraints:

- A publisher can fire an event class or proxy event class locally (both components residing on the same machine).
- A subscriber can subscribe to an event class or a proxy of an event class locally.
- Each subscriber and their proxies must have a unique GUID if they are to be stored on a computer.
- Transient subscribers require the event class to reside on the same machine.

D.1. Simple single remote subscriber structure

This method of distribution applies to one-to-one or many-to-one publisher and subscriber relationship, as illustrated in Figure 15. A single subscriber results in a simple structure for communicating the events. The publisher collects all the data from the network component and fires an event to a proxy. The COM+ service locates the owner of the proxy (the client machine) and passes the event to the subscriber. It is required to export the publisher application to a server computer while the original publisher application remains on the client computer. This structure is simple to construct and maintain, but the method is limited for single subscribers only.

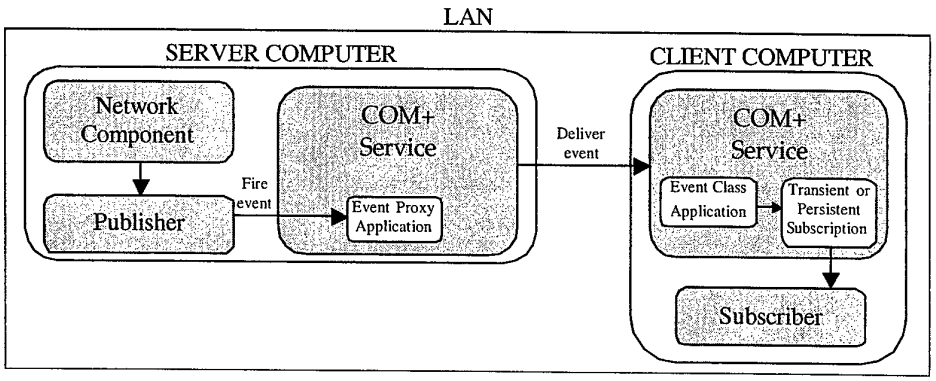


Figure 15: Simple single remote subscriber for transient and persistent subscriptions

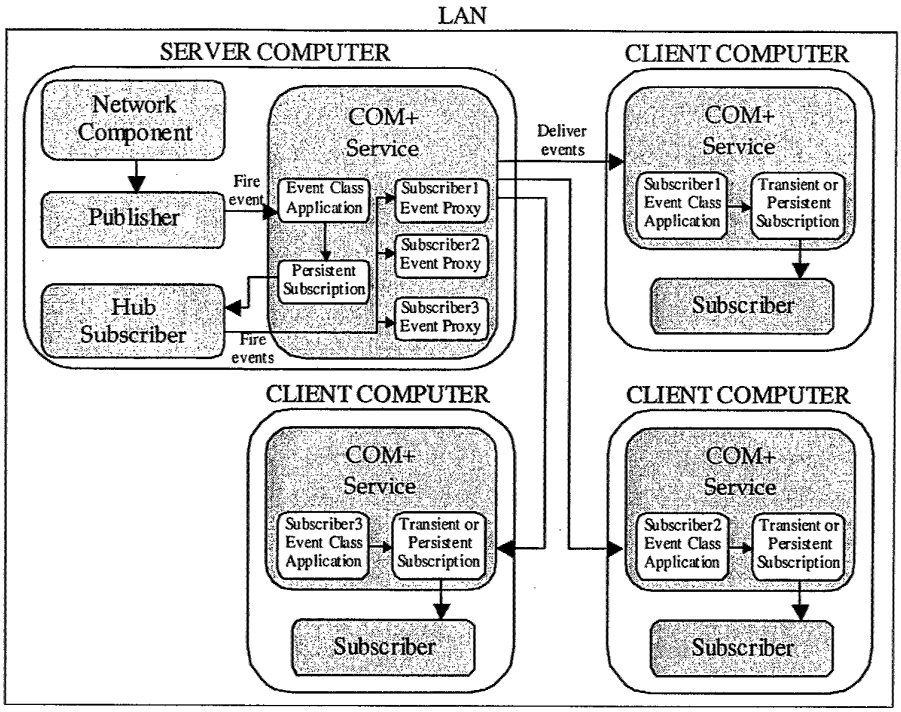


Figure 16: Multiple subscribers for transient and persistent subscriptions

D.2. Multiple subscribers for transient and persistent subscriptions

In this method, an extra subscriber is used to act as a hub to distribute events to each individual subscriber. It supports a many-to-many publisher and subscriber relationship. As shown in Figure 16, the hub subscriber is situated in the server computer with the publisher. The publisher fires the event to the COM+ service, which locates the hub subscriber through a persistent subscription. The hub subscriber then fires all the events to the proxies of the event classes for the client machines. The COM+ service in each client machine receives the event and passes it to the subscriber.

This method requires each client computer to host its own event class with a different GUID. These event class applications are exported and their proxies installed to the server computer, while the original application remains on the client computers. It is required to create the hub subscriber on the server computer as a persistent subscription to the original event class. The hub subscription is able to handle the event and fire the event to the proxy of each client machine. This method is flexible as it can provide support for both persistent and transient subscribers. The drawback is the hub subscriber must be rebuilt each time a subscriber on a new client machine is added, which may become a maintenance nightmare. Further, the hub subscriber has to fire the events in a sequence since firing events in parallel is not available.

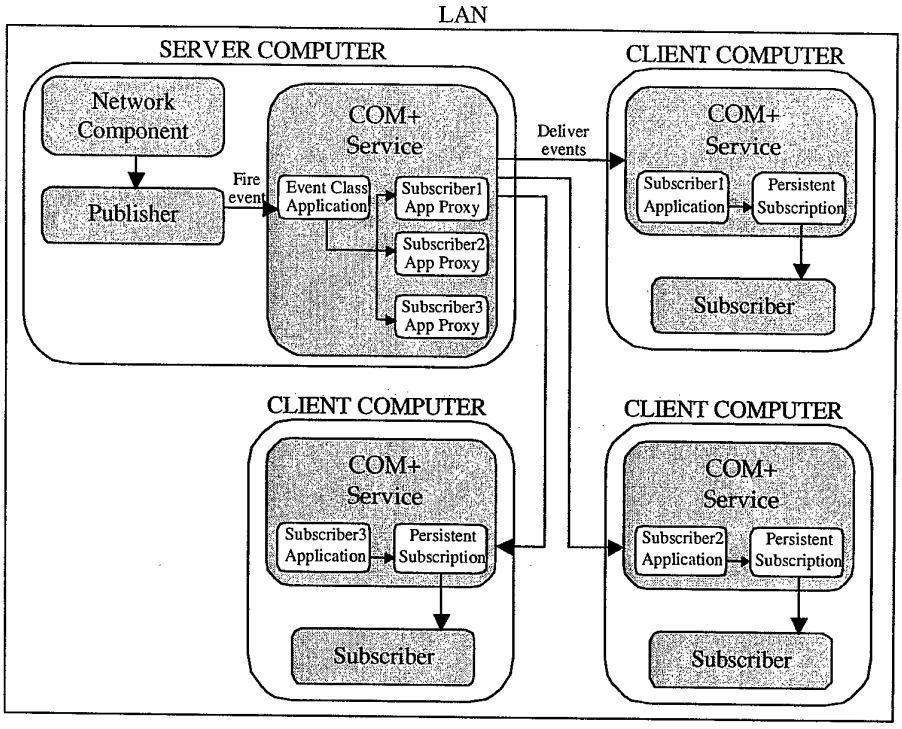


Figure 17: Multiple subscribers for persistent subscriptions

D.3. Multiple subscribers for persistent subscriptions

In this method, each subscriber is assigned a different GUID. As illustrated in Figure 17, the publisher fires the event to COM+ service in the server computer, which is able to locate the persistent subscriptions from the client computers through their proxies. The COM+ service then fires each subscription in turn. The COM+ service in each client computer receives the event, loads the subscriber and passes it the event.

Supporting persistent subscribers only, this method requires all subscribers to be rebuilt, each with a different GUID. The subscribers are installed on the client computers, while their applications are exported and the proxies are installed on the server machine. Subscriptions can be added before or after installation of their proxies. It therefore requires less maintenance and configuration, but rebuilding of subscribers is unavoidable. Due to its simplicity and capability, this method appears to be suitable to our requirements and is selected in our implementation.

DISTRIBUTION LIST

Enabling Headquarters Reachback : Adaptation of Collaborative Applications for Tenuous Communications

T. Andrew Au and Cindy Tran

AUSTRALIA

DEFENCE ORGANISATION

	No. of copies
Task Sponsor	
Deputy Director Theatre Command, Capability Development Group	1
S&T Program	
Chief Defence Scientist	shared copy
FAS Science Policy	
AS Science Corporate Management	
Director General Science Policy Development	
Counsellor Defence Science, London	Doc Data Sheet
Counsellor Defence Science, Washington	Doc Data Sheet
Scientific Adviser to MRDC, Thailand	Doc Data Sheet
Scientific Adviser Joint	1
Navy Scientific Adviser	Doc Data Sht & Dist List
Scientific Adviser - Army	Doc Data Sht & Dist List
Air Force Scientific Adviser	Doc Data Sht & Dist List
Scientific Adviser to the DMO M&A	Doc Data Sht & Dist List
Scientific Adviser to the DMO ELL	Doc Data Sht & Dist List
Information Sciences Laboratory	
Director, Information Sciences Laboratory	Doc Data Sht & Dist List
Chief of Command and Control Division	Doc Data Sht & Dist List
Research Leader, Command & Intel Environments	Doc Data Sht & Dist List
Research Leader, Theatre Command Analysis	1
Research Leader, Military Information Enterprise	1
Head, Headquarters Systems Experimentation	1
Head, Information Systems	1
Task Manager: T.A. Au	1
DSTO Library and Archives	
Library Edinburgh	1
Australian Archives	1
Library Canberra	Doc Data Sheet
Capability Systems Division	
Director General Maritime Development	Doc Data Sheet
Director General Aerospace Development	Doc Data Sheet
Director General Information Capability Development	Doc Data Sheet

Office of the Chief Information Officer

Deputy CIO	Doc Data Sheet
Director General Information Policy and Plans	Doc Data Sheet
AS Information Strategies and Futures	1
AS Information Architecture and Management	Doc Data Sheet
Director General Australian Defence Simulation Office	Doc Data Sheet

Strategy Group

Director General Military Strategy	Doc Data Sheet
Director General Preparedness	Doc Data Sheet

Navy

Director General Navy Capability, Performance and Plans, Navy Headquarters	Doc Data Sheet
Director General Navy Strategic Policy and Futures, Navy Headquarters	Doc Data Sheet

Air Force

SO (Science) - Headquarters Air Combat Group, RAAF Base, Williamtown NSW 2314	Doc Data Sht & Exec Summ

Army

ABCA National Standardisation Officer, Land Warfare Development Sector, Puckapunyal	e-mailed Doc Data Sheet
SO (Science), Deployable Joint Force Headquarters (DJFHQ) (L), Enoggera QLD	Doc Data Sheet
SO (Science) - Land Headquarters (LHQ), Victoria Barracks NSW	Doc Data & Exec Summ

Intelligence Program

DGSTA Defence Intelligence Organisation	1
Manager, Information Centre, Defence Intelligence Organisation	1 (PDF version)
Assistant Secretary Corporate, Defence Imagery and Geospatial Organisation	Doc Data Sheet

Defence Materiel Organisation

Head Aerospace Systems Division	Doc Data Sheet
Head Maritime Systems Division	Doc Data Sheet
Chief Joint Logistics Command	Doc Data Sheet
Head Materiel Finance	Doc Data Sheet

Defence Libraries

Library Manager, DLS-Canberra	Doc Data Sheet
Library Manager, DLS - Sydney West	Doc Data Sheet

OTHER ORGANISATIONS

National Library of Australia	1
NASA (Canberra)	1

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy	
Library	1
Head of Aerospace and Mechanical Engineering	1
Serials Section (M list), Deakin University Library, Geelong, VIC	1
Hargrave Library, Monash University	Doc Data Sheet
Librarian, Flinders University	1

OUTSIDE AUSTRALIA

INTERNATIONAL DEFENCE INFORMATION CENTRES

US Defense Technical Information Center	2
UK Defence Research Information Centre	2
Canada Defence Scientific Information Service	1
NZ Defence Information Centre	1

ABSTRACTING AND INFORMATION ORGANISATIONS

Library, Chemical Abstracts Reference Service	1
Engineering Societies Library, US	1
Materials Information, Cambridge Scientific Abstracts, US	1
Documents Librarian, The Center for Research Libraries, US	1

SPARES	5
Total number of copies:	**33**

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE

Enabling Headquarters Reachback : Adaptation of Collaborative Applications for Tenuous Communications

3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)

Document	(U)
Title	(U)
Abstract	(U)

4. AUTHOR(S)

T. Andrew Au, Cindy Tran

5. CORPORATE AUTHOR

Information Sciences Laboratory
PO Box 1500
Edinburgh South Australia 5111 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-TR-1588	AR-013-122	Technical Report	June 2004

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
E9505/25/80	00/256	DC2ISD	26	26

13. URL on the World Wide Web

http://www.dsto.defence.gov.au/corporate/reports/DSTO-TR-1588.pdf

14. RELEASE AUTHORITY

Chief, Command and Control Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS Yes

18. DEFTEST DESCRIPTORS

Information access
Digital battlefield
Network centric warfare
Operational intelligence

19. ABSTRACT

The ability to access information on demand is a critical capability in the battlefield. Deployed military headquarters are often located in a challenging tactical environment, whereby major disruptions to collaborative work lead to an impairment in the operational capability. These challenges in turn stem mainly from volatile connectivity and sporadic network mobility, which are a feature of the unpredictable nature of the battlefield. A network management approach based entirely on resource reservation is especially difficult in this environment since disturbances and unanticipated events often occur with network-based applications. This paper explores the concept of network awareness in support of continued operation in an unpredictable battlefield environment. In particular, we propose a platform-independent event-delivery framework to facilitate application adaptation, and demonstrate the advantages of this approach in supporting the proactive management of applications.