Handling Multimedia Data for Mobile Computers

Charles E. Perkins
Communication Systems
IBM T.J. Watson Research Center
Hawthorne, NY 10532
perk@watson.ibm.com

Abstract

Mobile clients have unusual needs for sending and receiving multimedia data. The traditional client/server paradigm does not take into account variable network conditions; thus traditional client/server applications have no capability to respond to intermittent connectivity, changing cost structures, or the kind of fluctuations in available bandwidth which are characteristic of wireless links. Two approaches towards allowing such adaptations are presented for comparison: proxies, and modifications to client and server.

1 Proxies

There are two approaches evolving to handle the new problems presented by mobile computers when they need to send or receive multimedia data. One approach is to create an intermediary known as a *proxy*. This is especially popular for applications such as Web browsers, where there is no particular need to maintain lengthy connections between client and server [1]. With a proxy, the client can essentially post a request and return at some later time to acquire the results.

Once the proxy is available, however, many other techniques become available. The mobile node can use the proxy to filter its incoming data, perform retransmissions, perform compression and/or encryption, and any of a wide variety of other useful operations [5].

One of the advantages of using a proxy service in this manner is that, in some configurations, neither the client nor the server has to be aware that a proxy is involved. Other gradations of service are possible with mobile-IP [3]. For instance, with the natural operation of mobile-IP, a mobile node could easily have a proxy associated transparently with incoming data, without affecting its output stream. This would be done if the home agent performs packet filtering functions similar to those performed by a firewall, but relays packets to a proxy service before being encapsulated and sent to the mobile node.

Proxy services allow offloading functions away from mobile stations, which can make the mobile stations lightweight and able to sustain longer battery life. For instance, the Infopad project [2] architecture offloads everything from X11 protocol processing to handwriting and voice recognition to a set of *type servers* operating on the other side of a wireless gateway. This necessarily creates a more intimate relationship between the mobile unit and its proxies, and implies that the mobile unit cannot operate very well without continuous network connectivity.

2 End-to-End solutions

Another approach is to modify the client and server parts of the applications so that they can adapt to changing network conditions. For example, researchers at CMU have attempted to identify the dimensions along which data can degrade in lossy channels [4]. If an application can specify the *fidelity* (along a certain dimension) which it requires to be able to use the incoming data, data which supplies unnecessarily high fidelity can be filtered or discarded. A simple example is selecting text-only mode for Web browsing when the network interface is too slow. Since such browsers already measure effective bandwidth, it would be a simple matter for the user to specify preferences that depend on the available bandwidth. At 9.6kbit/s and below, text only might be desired; at higher speeds, only graphic images that load in less than a second might be considered loadable by the browser unless specifically requested by the user. Other aspects of the network connection should also be made visible to the application. For instance, instead of an amount of time, the user might wish to say that any image that costs more than $.02 should not be loaded without interactive approval. Applications that deal with information of varying sensitivity labels should be able to use information about the nature of the network link to decide what style of security assurances to employ.

Obtaining information about the changing nature of network connectivity is a new requirement. One way to meet this requirement is to use a callback mechanism with mobile-IP; another way involves continuous polling of the network interface adapter parameters. These methods are not mutually exclusive.

3 Summary and Conclusions

It will be interesting to watch how mobile devices change the way multimedia applications interact with the network. It is, of course, particularly the high volume of data required by multimedia applications that highlight the need for programmatic solutions to the variability of network conditions. Either of the above approaches will go a long way towards enabling mobile devices to be full-fledged members of the Internet multimedia community. I tend to favor the second approach, just because proxies inherently insert delays and a certain rigidity into handling data for their clients. There are simply too many ways that applications might wish to adapt. Proxies cannot be reasonably expected to offer services in so many different ways. However, they may be very useful in many particular circumstances.

Acknowledgments

Thanks to Carl Tait, Henry Chang, and Dan Duchamp for sharing with me interesting ideas about proxies. Thanks to Satya (at CMU) for interesting ideas on modifying clients and servers according to specified characteristics of the incoming data.

References

[1] C. Brooks, M. Mazer, S. Meeks, and J. Miller. Application-Specific Proxy Servers as HTTP Stream Transducers. In *4th International World Wide Web Conference*, 1995.

[2] S. Narayanaswamy et. al. Application and Network Support for InfoPad. *IEEE Personal Communications*, 3(2):4–17, April 1996.

[3] C. Perkins, Editor. ietf-draft-mobileip-protocol-16.txt - work in progress. IPv4 Mobility Support, April 1996.

[4] M. Satyanarayanan. Mobile Information Access. *IEEE Personal Communications*, 3(1):26–33, February 1996.

[5] B. Zenel and D. Duchamp. Intelligent Communication Filtering for Limited Bandwidth Environments. In *Proceedings Fifth IEEE Workshop on Hot Topics in Operating Systems*, pages 28–34, May 1995.