Active Information Networks

Ian Marshall

BT Labs, Martlesham Heath, Ipswich, IP5 3RE

Ian.W.Marshall@bt.com

Abstract Future requirements for a broadband multimedia network are discussed and a vision of the future network is presented. Three key needs are identified; rapid introduction of new services, dynamic customisation of services by clients, and minimal management overhead. Application layer active networking, perhaps the most pragmatic and immediately realisable active network proposal, is introduced as a potential solution to all three.

Introduction

The characteristics and behaviour of future network traffic will be different from the traffic observed today, generating new requirements for network operators. Voice traffic will become another form of data, most users will be mobile, the amount of traffic generated by machines will exceed that produced by humans, and the data traffic will be dominated by multimedia content. In the medium term the predominant multimedia network application will probably be based around electronic commerce capabilities. Operators will therefore need to provide a service, which provides an advanced global trading environment for buyers and sellers of any commodity, at a low cost. The e-trading environment will be equipped with all the instruments to support the provision of a trusted trading space. Most important is the ability to support secure transactions over both fixed and mobile networks. Networks will thus need to be robust and contain built in security features, but also sufficiently flexible to address rapidly evolving demands as, in the longer term, other unforseen applications become predominant.

Existing networks are very expensive, and the deployment of new communication services is currently restricted by slow standardisation, the difficulties of integrating systems based on new technology with existing systems, and the overall system complexity. The biggest cost is management. The network of the future will need to be kept as simple as possible by using as few boxes as possible, removing duplication of management overheads, minimising signalling, and moving towards a hands off network.

The simplest (and cheapest) current networks are multiservice networks based on powerful ATM or IP switches. New transport networks are designed on the basis that nearly all applications will eventually use internet-like connectionless protocols. The difficulties of adding services such as multicast and QoS to the current internet demonstrate that even these simpler ip based networks will require additional mechanisms to enhance service flexibility. The simple transport network will thus need a flexible service surround. The service surround will provide a trusted environment, with security features (for users, applications and hosts), QoS support, application specific routing, automatic registration and upgrade for devices connected to the transport network. It will also enable Network Computing facilities such as secure gateways, application layer routers, cache/storage facilities, transcoders, transaction monitors and message queues, directories, profile and policy handlers. Such a service surround will likely be based on some form of distributed middleware, enabling the features to be modular and interoperable. The service surround will enable rapid introduction of new features by the operator. In order to minimise the management overhead clients will directly control which features should be used for a particular session, without operator intervention. The network will thus need to know nothing of the semantics of the session. To achieve this a middleware based on some form of active services or active networks will be required.

Active networks and services

Active networking was originally [1] a proposal to increase flexibility by adding programmes to packet headers which are intended to run on network devices that the packet encounters. However service operators will never permit third party programmes to run on their equipment without prior testing, and a strong guarantee that the programme will not degrade performance for other users. Since it will be extremely hard to create interesting programmes in a language which is simple enough to enable termination guarantees, we regard this approach as unrealistic.

A somewhat different flavour of active networking, in which the packets do not carry programmes but flags indicating the desirability of running a programme, has also been proposed [2] based on a dynamic interpretation of programmable network ideas. This would enable the service operator to choose a programme to load, which matches the indicated need, but is sourced from his own database of tested programmes. We believe that this second approach may be valuable in the long term, however in order not to degrade router performance the number of flags will need to be small and the number of possible programmes will thus also be small. In addition it will not resolve the immediate need for increased flexibility, as it will require standardisation, which will take time.

We therefore propose a third alternative, which we call application layer networking [3]. A similar proposal [4] was described as active services. In this system the network is populated with application level entities we refer to as service nodes, or dynamic proxy servers. These could be thought of as equivalent to the http caches currently deployed around the internet, but with a hugely increased and more dynamic set of capabilities. This approach relies on redirecting selected packets into the application layer device, where programmes can be run to modify the content, or the communication mechanisms. The programmes can be selected from a trusted data source (which is normally a cache and has minimal restoration overhead), and can be run without impacting router performance or requiring operator intervention. Packets are redirected on a session by session and protocol by protocol basis, so there is no need for additional flags or standardisation. Programmes are chosen using the mime type of the content (in the application layer header), so again no additional data or standards are required.

In our initial implementation, programmes to be run on the dynamic proxy server are implemented as java beans for portability reasons, and the proxy server itself is a java application. The beans can be loaded on demand and are referred to as proxylets (an analogy with applets and serverlets). A proxylet is a specific type of bean, which supports the management operations Load, Run, Stop and Modify. The management operations are normally invoked by clients (through operating system filters). Stateless proxylets are preferred since the management overhead is lower. If stateful proxylets are required it is necessary to add further operations to the list such as Suspend (stores state pending restart – possibly at another location), and implement network based state management schemes. The redirection of packets in a session carrying a particular protocol (such as http) to the proxy is initiated either by the client of the session or by the server. This avoids the issues associated with transparent redirect, and in our view is preferable, but transparent redirect would be possible.

We have shown, by implementing appropriate proxylets, that our approach offers performance benefits for streaming of retreived audio and video files, for link compression, for enabling multicast content to propagate over unicast links, and for bridging tcp sessions. The management overhead is minimal. We are currently developing proxylets which perform authentication (for mobile access), content transcoding (for caching dynamic content and for adapting to low functionality clients), dynamic redirection, and message queuing. We anticipate demonstrating benefits for all the proxylets under development and evolving further proxylets to supply all the middleware functions required in the network of the future. We therefore believe that application layer active networking has a crucial role to play in the broadband multimedia network, both relatively soon and in the long term.

References

[1] D.Tennenhouse, D.Wetherall, "Towards an active network architecture" Computer communication Review, 26, 2 (1996), pp5-18

[2] Alexander, Shaw, Nettles and Smith "Active Bridging" Computer Communication Review, 27, 4 (1997), pp101-111

[3] M.Fry and A.Ghosh "Application layer active networking" HIPPARCH '98 Workshop

[4] E.Amir, S.McCanne, R.Katz "An active service framework and its application to real time multimedia transcoding" Proc SIGCOMM '98 pp178-189