Active networks and high speed content delivery

Eva Hladka and Ludek Matyska

Faculty of Informatics, Masaryk University of Brno

1 Introduction

The best effort approach is typical for Internet data transmission since the early days. New kind of services, different application requirements, and real time data transport asked for a change in this model. ATM networks introduced several years ago included a Quality of Service model aimed to solve at least some of these requirements. However, the theoretical results in this field were too difficult to implement and nowadays ATM technology is used in backbones (if at all), but definitively not to user's end machines and the QoS must be achieved by other means.

The "end to end" argument used in computer networks till now limits more complicated network services. It is time to find architectures not limited by this approach and one of them may be the active (or programmable) networks approach. If we simplify the computer network as a set of end stations, network elements (routers or switches) and passive links among them, active network allows transported data to actively change functionality of network elements. This is not as revolutionary idea as it seems. More complicated network services requested network element activity more earlier then active networks idea emerged. Typical example of these protocols is multicast - data packets are replicated inside the network on network elements - or MPLS, with egress routers labeling different dataflow packets.

Very important feature of active networks is their flexibility. New network services and protocols can be implemented easily and quickly and problem with differences between simulator results and real production network, which was fatal in deployment of quality of service in ATM networks may be eliminated. The role of the prototype system and environment for understanding new network protocols and services by its implementation makes active networks very attractive for research, as well as for educational purposes.

2 Architectures

The basic model of active network is that network is created by active nodes (routers, switches) connected by passive links. Active node is a node with possibility to execute program code of the third party (user, application).

The basic criterion lies in the way how active code is spread to the active nodes[1]:

- architecture of active nodes - the program code is injected to node separately of data packets,
- architecture of active packets - the program code is included in data packets,
- architecture of active packets and active nodes - combination of both previous architectures.

2.1 Active nodes

The code is injected to the nodes separately of the data packets. The code can be realized as built-in functions or during the opening phase of data transfer. Advantage of this architecture is that the code is injected only once and it implies that the length of program code is not critical. Disadvantage is the necessity to inject the code before data transmission, which means larger delays and lesser flexibility.

2.2 Active packets

Each data packet contain the program code, which is extracted on active node and executed on the data part of packet. This access is flexible and clear, the data packets in one data transmission can be processed by different programs, the node must be able "only" to extract the code and execute it. Disadvantage is in the number of transmitted data (large overhead), the total length of packet is increased and the length of program in packet is strictly limited (must be very short).

2.3 Active packets and active nodes

This combination of both previous architectures makes possible to use more complex programs and be flexible enough thank to program command or parameters in data packets.

3 Models and implementation

For every model, every implementation, there are two basic challenges-to be so safe and so fast as traditional networks. It is not too little, if we accept the fact that the code executed on routers/switches migrate with data through the network and may have nontrivial demands on network resources.

Since 1995, when the concept of active networks started to be developed, lot of models and implementations for each of the above mentioned architectures were created and implemented. For active packets special languages were designed, e.g., PLAN [2] or Netscript [3]. PLAN (Packet Language for Active Packets) became a base for several models and implementations.

The first of them, ANTS [4] was developed on MIT by SDS group. The goals of ANTS developers were in fact the same as our goals. ANTS works with capsules and active nodes, capsules are generalization of packets and include routines for processing (data transfer, application)-it is similar to the connection message in our model and active nodes for protocols execution with strict described node activity. System is implemented in Java and runs as user-level process under Linux.

The second, DAN [5] model is dedicated to be realized on high-performance hardware optimized for the purpose of this model. This is in fact the source of the main differences. Moreover, the DAN model is based on distributed functions from code server, capsules can obtain function identifiers, which will be realized. The code server with function database is important part of this system.

The third, PRONTO [6] model developed ad AT&T Research laboratories is a system of active nodes built on Linux operating system, i.e., the active node is based on noncommercial operating system. Facilities in Pronto are divided into generic facilities, execution environment, and interface between them. The Pronto platform consists of a set of kernel modules and a user level library, with defined API to access these facilities.

4 Our active network model

We have developed an active network architecture using active nodes and we are working on implementation of this architecture using software routers based on NetBSD (Unix) operating system. The software routers are widely used in our metropolitan area network and represent a general programmable platform. We are looking for a model which permits not only new network protocols tailored to new applications, but gives us possibility to implement new QoS capabilities in the environment of connectionless networks. Highly variable demands to the network functionality were the main argument for the development of this new model. The proposed network architectures uses a connection oriented approach. While a full scale connection-oriented approach for all individual streams crossing the wide area network is a fiction, as proved by problems of ATM QoS model, the premium quality of service requirements cannot be achieved without using at least a little bit of connection oriented approach.

Active router plays a key role in our model. It is a network element which is able to recognize active packets and to process them. The processing of active packets has two disjoint phases. The first one is connection establishment and management. This phase includes initial load of user functions onto the routers along the path between source and destination address or addresses and execution of bookkeeping functions. The second phase, initiated by the first one, represents the data packet processing itself. The active router model extends the router functionality through a family of new protocols. The protocols can be divided in two groups relating to two phases of the active packet processing - protocols for network connectivity and program loading and those for processing loaded programs. Security of this AN model is based on authentication and authorization protocols and security properties of other used protocols.

The major distinction with respect to other connection-oriented approaches, where the connection either cannot be dynamically re-established, or the parameters cannot be dynamically changed lies in the support of flexible changes in connection parameters. Within our model, it is possible to change parameters as an atomic operation, thus overcoming a danger of two separate steps in the highly parallel environment: it may not be possible to construct a new path after the tearing of the already established one because a completely independent connectivity request may consume all the just released resources. This type of connectivity could be characterized as dynamic connection-oriented, changing itself in time depending on the network behavior and the user changing requirements. For some applications the programmable connection is all that is required and no active program is needed.

Adapting a PC router to an active router means to introduce new features on the kernel and user levels. On the user level, the authentication and authorization module must be added, together with the full environment for the connection and active programs including parts of the resource management and statistics collection. On the kernel level, it is necessary to change functionality of packet classifier (standard part of PC router) which recognizes data packets of active connection, and also to add the active program scheduler. Our implementation uses the C language to implement both the user and kernel level modifications and extension of the PC router.

5 Applications

Active networks provide an excellent development environment for new network protocols and applications. The main areas of applications whose performance can be dramatically increased via active network support are:

- Reliable multicast
- Multimedia applications
- Network and traffic management
- Mobile IP services

5.1 Multimedia applications

Multimedia applications, especially in real time (e.g. videoconferencing) use multicast and real-time services. Combination of both these services poses challenge in current Internet. For video and audio, the datastreams are sent from sender to receiver(s) and in active networks, the data can be processed on active node with knowledge of link bandwidth to the next hop and with the knowledge of data character and data coding. The paradigm of reliability and quality of service is substituted by the paradigm ”do as well as possible”. With the use of knowledge of protocols and coding of transmitted data, active router can make the multimedia data flow to keep pace, i.e., to satisfy the delay and jitter specifications. When congestion occurs and it is necessary to drop packets, the knowledge of coding (i.e,, the semantics of individual packets and data within them) it is possible to minimize perceived loss of transmission quality (e.g., visual problems in MPEG videostreams can be reduced if I-frames are dropped first) [7]. This strategy can be used for any data transmission which does not require absolute reliability and where some loss of data can be tolerated. When transmitting sound, for example, the possible solution is to buffer (cache) packets which represent e.g., 100 ms of sound. If more packets should be buffered (e.g., for high congestion on the outgoing link), the oldest one can be dropped as humans are much more adapted to tolerate some interrupts in transmitted sound then to tolerate desynchronization (which must naturally occur if data re-transmission takes too long time). Again, the decision when to drop packets can be made locally by an active element within the network with knowledge of local transmission problems.

5.2 Reliable multicasting

Reliable multicast over the Internet is very difficult problem. One of the possible solutions requires active role of routers in loss recovery [8]. By suppressing multiple NACKs routers control the NACK implosion problem. To reduce wide-area recovery latency and to distribute the retransmission load, routers cache multicast data on best effort basis. Combination of this techniques with multicasting data characteristics and coding, the active reliable multicast can give us flexible and robust solution for reliable multicast problem.

5.3 Network and traffic management

Network and traffic management are very important activities whose correct and timely deployment can heavily influence the overall network behavior and performance. However, both are problems practically unsolvable without at least some support included within the network itself. Active routers give us possibilities to solve both problems inside network and to find local optimal values for parameters influencing network behavior. Smart packets project [9] introduced experimental technology for network management. Congestion Control is a special case of network management, when a local action is usually necessary to quickly react to changes in individual data flows. Active nodes can monitor the available and consumed bandwidth and control the rate of individual data flows, transforming data at congestion points, selectively dropping packets or multistream them. Another are is active caching, where instead of placing large caches at specific points within the network, the self-organizing wide area ”intelligent” network caches were suggested [10]

6 Exemplary application

Our research within the area of active networks was inspired by the need to find protocols and networking arrangement for collaborative environment. Collaborative environment can be seen as three multimedia applications together - video, audio, and shared workplace. For simplicity, we will suppose that this environment connects only two persons, with logical point-to-point connection. We have three data streams with different requirements to bandwidth and QoS (e.g., jitter, latency, ...). In current networks, all the data will be processed across the network in the same way. In active network, third party (e.g. user, application) could arrange different conditions for transmission of any of these three dataflows through the network.

Let the first dataflow be video. Video is application which needs high bandwidth, only small variations in jitter are allowed, and the datastream may have specific coding.

The second dataflow is made of audio. The specific feature of audio is that the delay between packets must be lower than 20 ms for uninterrupted and smooth hearing. As already noted above, one of possible active transmission models for audio is caching data packets on active nodes. 20 ms is too strict requirement in many cases for retransmissions of loosed packets directly from source, but if the retransmission may be satisfied from previous active node (with respect to the logical active network topology), the 20 ms requirement could be satisfied. Caching may thus make the audio data stream more smooth and it may increase a perceived quality of sound especially in network with fast changing congestion patterns.

In case of shared workplace it is necessary to transmit all data packets, however, the required bandwidth is usually rather low and there are no additional special requirements (like delay or jitter). The whiteboard data packets are obviously the lowest priority packets, but the overall network goodput can be again increased if caching similar to the scenario introduced above for audio streams is used.

Additional complexity may be considered when taking into account that esp. the audio and video streams are not fully independent. They must be somehow (weakly) synchronized, otherwise the final impression at the destination may be completely unacceptable. Naive idea may be based on suggestion that when we have audio and video data streams and packets on the video stream must be dropped, the synchronized audio packets should be dropped as well.. While being possible, the result would not be really good - ears are more sensitive than eyes and loosed packets in audio are more appreciable. More sophisticated protocols describing mutual inetraction of audio and video streams are necessary, but the primary point here is the fact that even such complex behavior may be programmed within the active element, leading thus to even better use of networking resources.

All the scenarios presented split the end-to-end (virtual) connection to several shorter ones. The first virtual connection goes from the source to the first active element, then from this first to the second and so on up to the last virtual connection between the last active node and destination point. Each individual virtual connection can react to the changes in network quality independently to the situation on other links and these changes in transmission throughput and pace are much faster then when only the source can react to changing network conditions. Active networks, whose elements are responsible for the local fast reactions, can thus achieve much better utilization of available network resources and contributes substantially to the perceived network performance.

When the collaborative environment is used by more than two users, it needs some kind of point to multipoint transmission. Scalable solution used within current Internet for this case is multicast, less scalable can be provided be one or several mirrors. In classical networks, routers spreading multicast data make two (or more) copies of data packets and try to transmit them without any respect to available bandwidth on individual output interfaces. Active multicast can respect the fact of heterogeneity of interfaces, with knowledge of data characteristics and coding can make optimal choice which packets should be transmitted (and with which priority) and which can be easily dropped. It can even make a re-coding of individual data streams: e.g., when the in-going data stream arriving on the high capacity link is high definition TV (which requires up to 100 Mb/s of available bandwidth) and there are two outgoing links, one with the same capacity as the ingoing link and the second one with much smaller bandwidth (e.g., 10 Mb/s Ethernet). In this case, the data stream to high capacity outgoing link is simply transmitted directly from the ingoing stream, while for the second outgoing link the high definition TV stream is re-coded to e.g., lower quality MPEG stream which could be easily transmitted on 10 Mb/s link without the danger of continuous congestion (and if a congestion occurs, the knowledge of MPEG data stream semantics can be used to drop the lest important packets).

7 Security and safety

Active networks are open systems for implementing intelligence to the network nodes. Adding computing elements into the network implies a danger of substantial safety decrease. It is necessary to reduce risk of mistakes in active code, to protect privacy, integrity, and availability, and to protect active elements from unintended or intended abuse or malicious use. Programs on active nodes consume network resources and there is a question who can and who cannot use these resources and to which extent. The techniques of authentication are used [11]. Features of security and safety must be present already in design of active network. The important question is the security and safety price in terms of processing power required. Executing and transmitting packets on active nodes must be as fast as possible and economical to node resources and it bounds possible safety and security techniques. Some kind of simple pre-authentication is used, which can quickly distinguish between definitively non-authorized users and those whose credential should be inspected by more rigorous methods. Fast pre-authentication can reduce the risk of denial of service and similar attacks to acceptable minimum, especially if it is accompanied with active protection from attack. It was suggested to use active networks to actively discover sources of denial of service attacks and to close directly the discovered sources, reducing thus impact to the whole network (in theory, you can discover the source of such an attack whenever the number of attacking packets is equal to the number of hops (active elements) between source and attacked system. Active networks can thus not only use passive protection schemes, they can ultimately increase the safety of the Internet.

8 High speed delivery context

In the previous text, the review of active network with their benefits and problems were described. Increasing functionality and software complexity on core network elements is used as very strong argument against their deployment for high speed networks, as opponents correctly points out that for bandwidths above gigabit per second even simple routing represents serious performance problem without support of specialized and therefore very costly hardware.

Network nodes hardware is currently represented by wide spectrum of devices ranging from PCs with routing daemon, small dedicated routers and switches to powerful gigabit backbone routers. These different models and architectures provide yet another argument against wide use of active networks. However, using more detailed analysis, it is not too difficult to find places, where active nodes can fulfil their beneficial role. The core of gigabit backbones evidently is not the proper place, because of its homogeneous nature. On the other hand, places where the network is heterogeneous, where different branches have different bandwidth, are the best points for active element placement. Specialized local networks, network infrastructure for mobile IP (esp. On egress where the change of network bandwidth is the most prominent) and places on strategic network point for traffic and network management are next natural candidates for inclusion of active network elements.

The exemplary application introduced earlier had shown how individual dataflows can be processed differently. The suggested processing leads to optimized data flow and the whole amount of actually transmitted data may be substantially greater (due to the congestion reduction or even elimination) than in classical network. Unfortunately, the load on active elements (routers) will be much larger. We suggest to eliminate this drawback by using high number of active elements, creating active node farms directly connected to the neighbor dumb networking elements. Each member of such active node farm would process only limited number of data streams, but together they can keep pace with even the highest data rates. And if the active elements are built on top of ordinary PCs, we see the same pattern which currently draw out specialized supercomputers and replaces them with clusters of PC. The same may happen with the advent of active networks, where such farms can take over almost all functionality of traditional dumb network elements which will remain only within deep core of super high speed networks.

The active node farms imposed within dumb computer networks can create logical active supernetwork which can process high number of active data flows. Processing of each flow can be programmed, the flows can be processed separately or in bunches (where some synchronization between flows should be used, like with video and audio streams from the same source). Total achieved throughput will be very high while the processing (and throughput) power of each individual active element could be kept in cost sensitive range. Active node farms can support aggregate bandwidth at the same level as the current most advanced core networking elements.

9 Conclusion

Two seemingly contradictory approaches are gaining more interest of both research and commercial community-the super high speed networks (tens of gigabits per second and more) and active networks. It is usually claimed that the really high speed can be achieved only on networks where the core elements (e.g., the optical routers) are just transmitting packets without any attempt to analyze them. On the other hand, new services with defined content-the (e.g., real-time voice or audio transmission, remote visualization, remote access to special experimental devices or robots) require support from the core network to be really efficient (and reliable within the time constraints they have). The paper presents an idea of overlaid networks, where active node farms create an overlay on top of the basic high speed "dumb" networks. Individual content based streams do not require super high speed and can be independently combined within the high speed network where only very simple quality of service can be assured. The active network can be specifically tailored for a particular application and in any given time many such specific active networks (e.g., reliable multicast for audio only, specific support for a videoconference, a real-time video delivery network or active network supporting multipoint access to large-scale distributed computational experiment on the GRID) may overlay the high speed backbone.

References

[1] K. Psounis. Active networks: Applications, security, safety and architectures. IEEE Communication Surveys, 1999.

[2] Official PLAN web site: http://www.cis.upenn.edu/ switchware/PLAN

[3] Y. Yemini and Sushil da Silva, Towards Programmable Networks, Proc. IFIP/IEEE Int'l Workshop on Distributed Systems, Operations, and Management, L'Aquila, Italy, 1996.

[4] http://www.sds.lcs.mit.edu/activeware

[5] Dan Decasper, Bernhard Plattner, DAN: Distributed Code Caching for Active Networks in IEEE INFOCOM'98

[6] G. Hjalmtysson, The Pronto Platform: A Flexible Toolkit for Programming Networks using a Commodity Operating System in OpenArch2000

[7] R. Keller, S. Choi, M. Dasen, D. Decasper, G. Fankhauser, B. Plattner, An Active Router Architecture for Multicast Video Distribution in IIEE NFOCOM2000

[8] L. H. Lehman, J. Garland, D.L. Tennehouse, Active Reliable Multicast

[9] B.Schwartz, W. Zhou, A. W. Jackson, Smart Packets for Active Networks, BBN Technologies, 1998

[10] S.Bhachttacharajee, K. L. Calvert, E. W. Zegura, Self-Organizing Wide Area Network Caches, IEEE INFOCOM'98

[11] P. Bellavista, A. Corradi, R. Montanari, C. Stefanelli, Security in Programmable Network Infrastructures: The Integration of Network and Application Solutions in Active Networks, IWAN 2000,LNCS 1942

[12] A. Juhola, K. Ahola, J. Molsa, S. Lehoten, Mobile Active Overlay in Active Networks, IWAN 2000,LNCS 1942