Quality of Service Oriented Active Routers Design

Poster Session, Informatics Seminar – 18th May 2006, Brno, Czech Republic

Tomáš Rebok
xrebok@fi.muni.cz
Faculty of Informatics
Masaryk University Brno
Botanická 68a, Brno 602 00
Czech Republic

Active Networks

Contemporary computer networks behave as a passive transport medium which delivers—or in case of best-effort service tries to deliver—data sent from the sender to the receiver. The whole transmission is done without any modification of the passing user data by the internal network elements.

We believe that the future-generation networks may be extended beyond that paradigm and behave as an active transport medium, which processes passing data based on data owners or data users requests. Multimedia application processing (e.g., video transcoding) and security services (data encryption over distrusted links, etc.) are a few of possible services which could be provided.

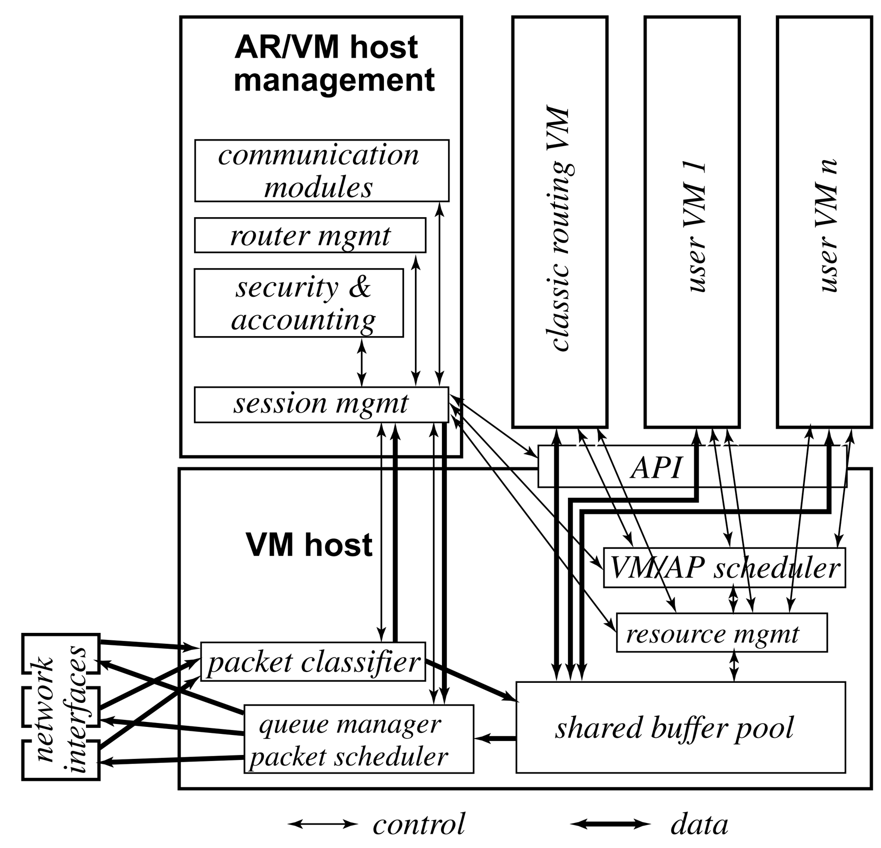
The principle called *Active Networks* or *Programmable Networks* is an attempt how to build such intelligent and flexible network using current "dumb and fast" networks as an overlay network. The inner nodes of such active network are directly programmable by users and so provide computations up to the application level. These inner elements are called *active nodes*, *active routers*, or *programmable routers* (all three with rather identical meaning). Users and applications have thus the possibility of running their own programs inside the network using these active nodes as processing elements.

An application of software programmable routers in multi-user environment pose new challenges in the design of router operating systems and especially in the design of resource management system. Since more resources are shared among the users of the active router—router CPU cycles, state storage capacity, data storage together with traditional networking components like packet queues on network interfaces. To enable sharing of all these resources within the active node by its users in a secure and effective manner, much more complex Quality of Service (QoS) architecture needs to be deployed, including sophisticated resource accounting and resource scheduling algorithms that respects characteristics of individual resources.

VM-ready Active Router Architecture

We have extended the generic active router (AR) modular architecture proposed by E. Hladká and Z. Salvet to support the complex QoS and also slightly modified the scheme in order to facilitate implementation based on virtual machines. This approach enables users not only to upload the active programs, which runs inside some virtual machine, but they are allowed to upload the whole virtual machines with its operating system and let their passing data being processed by their own operating system running inside uploaded VM.

The bottom part of our VM-ready active router is the VM-host layer where the core of the proposed VM-ready router is located. The core includes packet classifier, shared buffer pool, and packet scheduler modules (the other modules relevant to resource management—resource management module and VM/AP scheduler module—are described in more detail below). Packet classifier module classifies all the incoming packets whether they belong to any active session running on the router. It also extracts packets destined to the session management module and sends them directly to that module. The shared buffer pool module operates as the buffer space where all the incoming packets are stored before further processing and also all the outgoing packets before the packet scheduler module sends them onto the network.



VM-ready active router architecture

The VM-host management system is located in user space. Besides the other functions it has to manage the whole router functionality including uploading, starting and destroying of the virtual machines, security functions, session accounting and management. The virtual machines managed by the session management module could be either fixed, providing functionality given by system administrator, or user-loadable.

The VM-ready AR architecture uses a connection-oriented approach similar to the one used in active router proposed by E. Hladká and Z. Salvet before. In terms of our active architecture, the connection is also called *(active) session*, but each active session consists of one or more active programs/virtual machines, one or more network flows and potential QoS requirements.

When the session is established with the required resources, the data flow through the router could be briefly described in the following way: when a packet arrives to a network interface, the packet classifier module decides, whether the incoming active packet belongs to the given AR or not, based on information from the security and accounting module. If the packet is accepted, depending on resource allocations and actual scheduling algorithm, the classifier module forwards packet to the proper VM running on the AR or the new session establishment takes place. Depending on resource management, the active packet is processed in the VM and sent into the network through the shared buffer pool.

QoS Support for VM-ready Active Router

As obvious from VM-enabled AR architecture described above, there are the two main modules concerned with resource management: (1) resource management module and (2) VM/active program scheduler.

Resource management module. This module implements the crucial resource management scheme and has the following functionality:

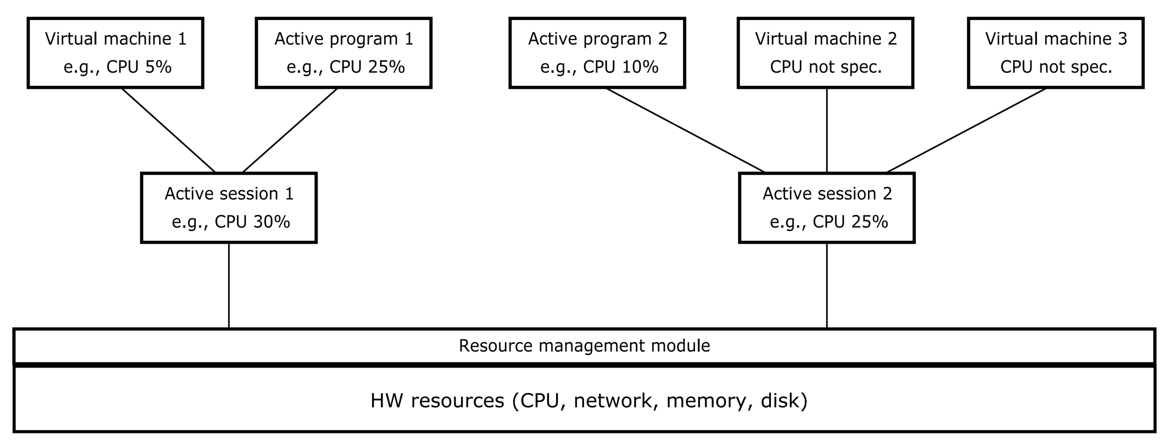
- Possessing all the information about the resources in the AR.
- Providing necessary information to the session management module.
- Monitoring and adjusting the resources used by each active session and sending notifications to the active sessions through the session management module to inform them about the actual resource status of the AR.

VM/active program scheduler module. This module schedules the execution of the applications and the transmission of the packets to the next node. It implements scheduling algorithms for different classes of resources to enforce the active sessions allocations of the AR resources. Besides that, the accounting and resource limit checking functions are also the part of this module:

- It checks whether the active sessions are permitted to request given resources.
- It logs active sessions requests and replies from resource management system about allocating given resources.

Resource Management System

Due to the structure of active sessions where each session consists of one or more virtual machines (simply active programs) and one or more network streams, the fine-grained hierarchical design of resource allocations is very desirable. I.e., when the session possesses allocated resources, it is possible to split these resources held by the session in a way the user of the given active session wants.



Hierarchical resource assignment

Schedulers

Scheduling algorithms are the most important part of the whole resource management system in our active router because they affect both overall performance and keep all required resources in desired limits. Since resource characteristics vary, scheduling algorithms must be designed in a resource specific manner. For example, CPU context switching is more expensive compared to switching between flows in network scheduling. Therefore, efficiency of CPU scheduling improves if active programs can receive a minimum CPU quantum before being preempted. Disk scheduling, unlike both CPU and network, must consider request locations to limit seek time and rotational latency overheads. Memory schedulers, in order to match actual memory use, must estimate the current working set of active programs. All the schedulers must therefore examine relevant resource states (e.g. disk state, whether it is spinning or parked) in addition to QoS specifications. Besides that, the resource requirements are often related to the others. For example, when requesting high network bandwidth while having only a small amount of CPU time, it is not possible to reach required bandwidth, because there is insufficient CPU time to send all the packets. Thus the scheduling algorithm's design must be sophisticated enough to take such inter-dependencies into the account.

Acknowledgements and References

Acknowledgements

This project has been supported by a research intent "Optical Network of National Research and Its New Applications" (MŠM 6383917201), "Parallel and Distributed Systems" (MŠM 0021622419), and "Integrated Approach to Education of PhD Students in the Area of Parallel and Distributed Systems" (No. 102/05/H050).

References

- E. Hladká and Z. Salvet, *An Active Network Architecture: Distributed Computer or Transport Medium*, ICN 2001: First International Conference Colmar, France, 2001.
- T. Rebok, P. Holub and E. Hladká, *Quality of Service Oriented Active Routers Design*, MIPRO 2006, Croatia, 2006.