AMNET: EFFICIENT HETEROGENEOUS GROUP COMMUNICATION THROUGH RAPID SERVICE CREATION

Till Harbaum, Anke Speer, Ralph Wittmann, Martina Zitterbart
Technical University of Braunschweig
[harbaum|speer|wittmann|zit]@ibr.cs.tu-bs.de

Keywords: Active and Programmable Networking, Heterogeneous Group Communication, Rapid Service Creation, Active Node Design

Abstract AMnet provides a framework for flexible and rapid service creation. It is based on Active and Programmable Networking technologies and uses active nodes (AMnodes) within the network. These AMnodes execute so called service modules to make user-tailored data streams available to individual receivers. This paper focuses on the goal of AMnet to design a platform for AMnodes which is shaped for efficient execution of flexibly loadable service modules.

1. INTRODUCTION

One paradigm of new evolving applications on the Internet like distributed computation, tele-collaboration or distance learning is that they are based on group communication which raises the need of efficient and scalable group services for proper communication support.

One special challenge for group communication, especially over the Internet, results from the heterogeneity caused by different network technologies, different applications, end systems and the like. Heterogeneity causes individual service requirements of different end users and applications which can not be met with a single group service. In most current approaches for providing group communication the service provided to individual group members is penetrated by the group member with the lowest service capabilities. Such an approach, however, is not acceptable for multimedia and collaborative applications in heterogeneous networking environments.

AMnet is a framework for flexible and rapid service creation [**?**] and provides primarily heterogeneous group communication services. One goal of AMnet is to make user-tailored data streams available to individual receivers. Following the concept of Active and Programmable Networking technologies [**?**, **?**]

AMnet uses active nodes – so called *AMnodes* – located within the network to execute *service modules* for the provision of individual group communication services. AMnet is not limited to the application of specific service modules. The AMnodes rather provide an open architecture for the dynamic incorporation of new services. They form the core building blocks of AMnet and operate on the network path between sender and receivers [**?**]. AMnodes provide an execution environment for efficient processing of flexibly loadable service modules. Dedicated hardware support with dynamically programmable FPGAs can be included transparently.

The paper is organized as follows. Section 2 introduces the AMnet approach including the service concept as well as service location and announcement. In section 3 the design of an AMnode is discussed. Section 4 reviews related work and section 5 closes with a summary and an outlook on ongoing research.

2. AMNET – ACTIVE MULTICASTING NETWORK

Tele-collaboration is a typical application of AMnet. Figure 1 shows a scenario where the participants differ in their requirements on Quality of Service (QoS). The AMnodes perform the adaption of the original data stream in accordance with the individual desired QoS-levels. To support service heterogeneity transparently to the source of a data stream as well as to the receivers, the AMnet service establishment follows a receiver oriented approach. Loadable service modules with out-of-band signalling are used for service creation, since

- capsules would have to be provided by the sender, i.e., the desired sender transparency and receiver orientation could not be achieved,
- for performance improvement of AMnet service provision a combined HW/SW solution is envisioned.

Both software-based service module execution and programmable hardware support for dedicated service modules is available. As the structure of the hardware dependent modules differs significantly from software-based modules, a capsule-based approach is not easily applicable.

Service Model. An *AMnet session* or *session* describes a communication scenario where a designated sender issues a data stream which can be received from several communication participants without or after adaptation, thus, different service levels are provided. Service heterogeneity within a session needs to be bound to a manageable degree of diversity. Therefore, one concept of AMnet is to logically group receivers with similar service demands into distinct multicast receiver groups – the *service level groups*. Service level groups are distinguishable by their multicast addresses. The receivers join the corresponding group on demand through IGMP [**?**].

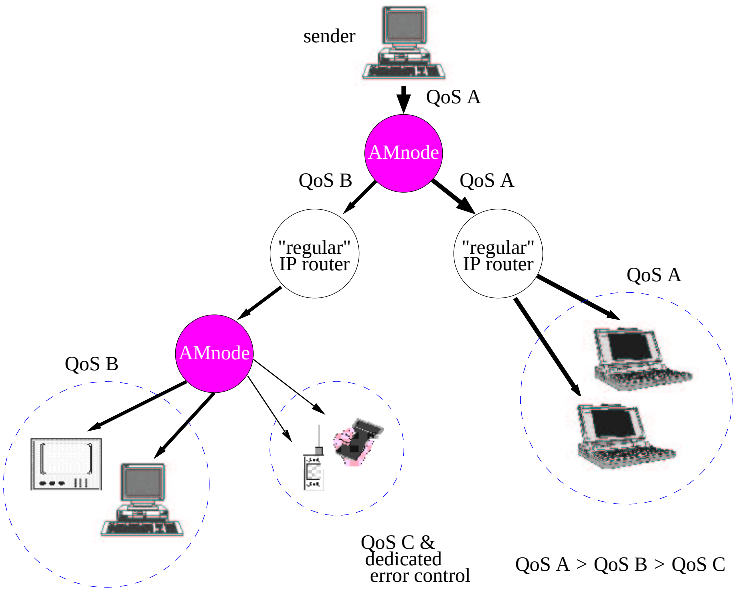

Figure 1 Basic AMnet scenario: Tele-collaboration

Each service level group within a communication session represents those receivers whose service demands can be resolved with a single group service. Therefore, each group represents a different view on the same original data corresponding to the adaptation within the AMnodes. AMnet specific signalling procedures are defined to locate and announce as well as to establish and maintain the corresponding services [**?**].
The communication service offered by AMnet can be described by a hierarchically ordered tree of service level groups (e.g., Figure 2). The service of a group is supported by an AMnode through the use of appropriate service modules. The actual service may be derived from the processing of the original data stream (see AMnode 1 in Figure 2) or from the service of another service level group (see AMnode 2 in Figure 2). The scope of an AMnet service level group is limited by the actual TTL value assigned to packets issued by the corresponding AMnode.
The establishment of service level groups permits the provision of different service qualities within one region without the services interfering with each other. Data streams with different media formats or individual error control must sometimes coexist on a communication link and have to be distinguishable by the appropriate receivers. All packets are explicitely assigned to corresponding service level groups by their multicast destination adresses.

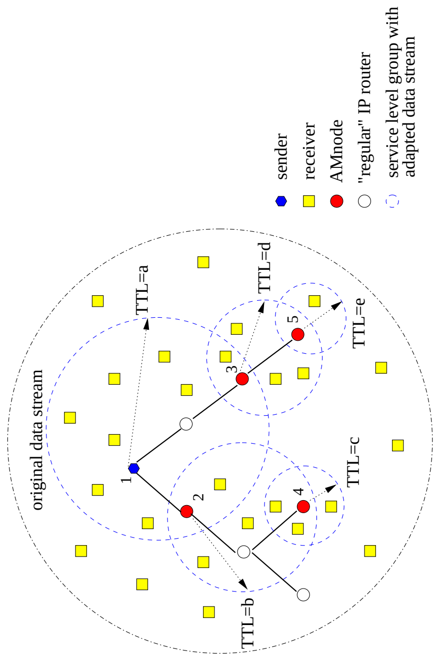

Figure 2 Example multicast tree

The hierarchical order of service level groups allows for an efficient establishment of different quality levels within a session. One distinct service quality might be easily derived from another already available service level, if, for instance, only a different (weaker) error control policy for network overload conditions has to be inserted into the already adapted data stream. However, the hierarchical ordering does not automatically imply hierarchical degradation of all service parameters. Some parameters can be provided unchanged, other parameters even improve. As an example, the insertion of a new service can improve media playback quality due to less jitter at the cost of higher, but uncritical delay. This could be useful for video distribution, for example.

Service Location and Announcement. Each AMnet session is announced within a global *session announcement group* (see Figure 3). Within this group, the description of the session, including information such as bandwidth and delay requirements as well as content specific information, e.g., data format and compression scheme, is exchanged. Based on this information the receiver decides whether to join a session. In the case of participation, the receiver joins the session specific *service announcement group* (see Figure 3) to learn about available services [**?**].

If the receiver wants to use an AMnet service to adapt the original data stream to an appropriate quality level, it examines the service announcements whether the desired service is already provided by an AMnode. If already available, the receiver joins the appropriate service level group of the selected AMnode. The participation can depend for example on metrics like distance or workload of the specific AMnode. If the service is not available, a procedure is started where – according to the propagated metrics selected – an AMnode is elected to provide the service. Therefore, the AMnode has to load the appropriate service module out of the distributed *service module repository* (see Figure 3), which maintains all accepted and verified AMnet service modules [**?**].

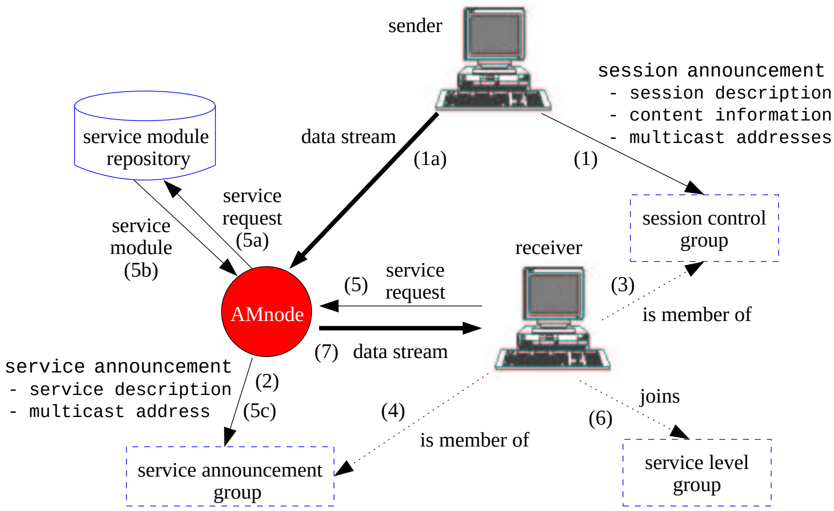

Figure 3 Service Control Architecture

3. AMNODE DESIGN

AMnodes form the core building blocks of AMnet. They are part of the data path and, therefore, an efficient design is critical. Research on Active Networking showed the importance of active node design for service flexibility, security and its performance [**?**].
An AMnode is build from a flexible execution environment and a set of fixed functional blocks (see Figure 4) which are responsible for the management of active node functionality and for the AMnet control protocols. The fixed path of the node provides abstract end-to-end management and local node control whereas the flexible component establishes the environment for the execution of active service modules. Each functional block will be briefly introduced in the following.

AMnode Service Management. The service management functionality is common to all AMnodes. It is represented by the AMnet signalling entity which implements the service control protocols and is responsible for the an-

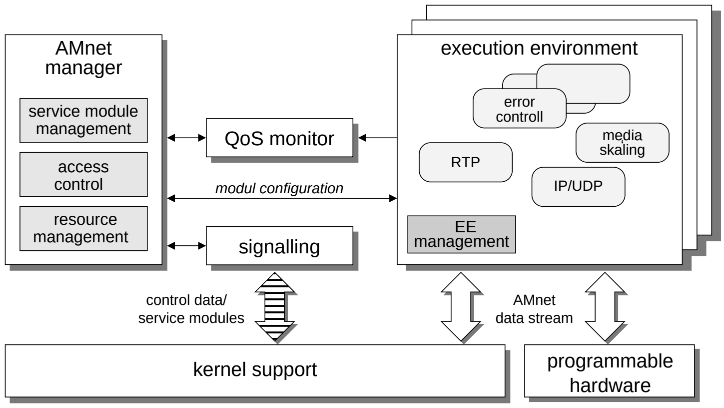

Figure 4 Architecture of an AMnode

nouncement of provided services and handling of service requests. AMnet signalling contacts the local AMnet manager which implements access control and resource management.

AMnode Node Management. Node management is the task of the local AMnet manager. The AMnet manager loads and/or configures service modules to provide requested services. Therefore, it receives needed information from AMnet signalling. The configuration of a service is complemented with appropriate local resource management.

Service Environment. Within the execution environment pre-loaded or on-demand-loaded service modules are executed. It is designed for a flexible establishment of receiver-tailored AMnet services. Loading and activation of service modules is controlled by the AMnet manager of an AMnode.
If the AMnode is equipped with dedicated programmable hardware, processor consuming service modules can be offloaded. This way, complex services, such as media translation, can be provided more efficiently.

QoS Management. After service establishment, a QoS monitor observes ongoing local operations and collects actual information for the AMnet manager. This functionality is intended to provide the AMnet manager with knowledge on the current performance and resource consumption of the node. Furthermore, service violations due to local resource shortage are reported to the AMnet manager. The information can be used for service and access control.

Service Modules. Service modules are loadable on demand and, thus, allow for dynamic adaptation to heterogeneous application requirements. For performance reasons, AMnet supports three implementations of service modules:

- Java: These modules are coded in Java and, due to Java's portability features, are executable on every AMnode. It is assumed that each service module is at least available as a Java code.
- Native: Modules of this type are coded in a compiled language, most probably C, and compiled for a specific architecture, operating system, etc. Obviously, with these modules a better performance can be achieved compared to Java modules. However, they salvage a security risk, since only limited measures to protect the node from malicious service modules exist.
- Happlets: A happlet is a service module coded in a specification language for hardware. It is executed on a dedicated programmable hardware device, which is part of an AMnode. These kind of services are expected to accomplish even complex operations very fast, thus, helping to overcome performance constraints.

For the transparent provision of service heterogeneity two types of service modules are currently considered: modules that perform media translation (such as media filters) and protocol adaptation modules as they are used in [**?**]. In several experiments we used so-called QoS-filters [**?**] to reduce bandwidth requirements of compressed video streams to enable receivers to participate in a video conference over low capacity links, e.g., wireless links. The results have shown that improved perception quality can be achieved with the AMnet approach [**?**]. In order to reduce the processing delay a library of filter functions for block-based video compressions in Java was developed. The library decomposes the steps necessary to scale various video streams. Often it is not possible or efficient to move all required processing to programmable hardware, since the service modules can be very complex, as in the case of media filters. With the decomposition of certain functions, however, it is possible to defer costly operations to specialized hardware to improve performance.

Hardware Support. With software-based service modules the achievable performance of time critical or CPU consuming operations such as on-the-fly media recording is limited. To overcome these limitations flexible hardware-based service modules have been introduced. The developed hardware platform FHIPPs (Flexible High Performance Platform) combines both DSP- and FPGA-based flexible service module execution with HW/SW co-design [**?**]. The control of the dedicated hardware is realized via the happlets explained above. They contain the necessary information to reprogram the hardware to provide a requested AMnet service and to dynamically install drivers for the interaction between host operating system and hardware units. Happlets can be retrieved on demand from the service module repository (see Figure 3) as it is done with the Java and Native modules.

4. RELATED WORK

Currently, different approaches to signal and provide heterogeneous communication services are developed. A possible action of the sender is hierarchical coding of the output stream. Receivers can select different media qualities by joining appropriate multicast groups related to the coding levels (e.g., [**?**]). In [**?**] a communication model is described where hierarchical coding into one output stream allows for lossy decompression at the receiver. Dependent on the loss ratio the original signal can be restored on a corresponding quality level.

The model of integrated services within the Internet [**?**] is a step towards heterogeneous services. Using the RSVP protocol heterogeneous resource reservations are possible. However, current flow specifications limit heterogeneity support to network performance parameters.

An efficient design of the intermediate systems or active nodes is another key feature in current research. In [**?**] an Active Bridge is introduced which can be reprogrammed "on the fly" with loadable modules. [**?**] introduces PANTS a dynamically extensible active network architecture for an active node which provides the flexibility to perform dynamic changes to the node. The Darwin Project [**?**] focuses on appropriate resource management mechanisms embedded in a new node architecture and [**?**] presents a solution which implements automatic protocol deployment and application specific processing and is also suitable for a gigabit environment.

5. CONCLUSION

In this paper the AMnet framework for the provision of heterogeneous group communication is presented. AMnet is based on Active and Programmable Networking technologies and uses *service modules* for the provision of individual services within a heterogeneous multicast group. These service modules can be dynamically loaded and activated within the network. Innovative steps of AMnet are the design of an active intermediate system which allows for efficient service provision and flexible service adaption.

Current work focuses on the signalling of heterogeneous services. Protocols for the announcement and control of heterogeneous services have been designed and are currently investigated with respect to their performance. The hardware integration will focus on the implementation of additional video compression functions and on performance measurements as well.

References