HEART : A Software Architecture for Distributed Multimedia Multiuser Applications

T. Kwon, S. Park, Y. Choi
Department of Computer Engineering
Seoul National University
Seoul, Korea

H. Kim, S. Kwon
Hyundai Information Technologies
Seoul, Korea

Abstract

The distributed multimedia multiuser applications are generally composed of multiple individually-controlled application service elements, which require various complex communication services. In this paper, we present HEART; a software architecture to support distributed multimedia multiuser applications in high-speed networks. The proposed HEART software architecture consists of three layers; the MMCP(Multimedia Multiuser Communication Platform) layer, the CASE(Common Application Service Elements) layer and the target application layer. The MMCP provides structured and dynamically-configured communication services to the individual applications of CASE layer. And the CASE layer provides various types of basic application service elements which can be flexibly configured and integrated for the development of target applications. The proposed HEART architecure will help to develop various distributed multimedia multiuser applications.

1 Introduction

With rapid advances in the computer technology on multimedia and in high-speed networks, various DMMA(Distributed Multimedia Multiuser Application)s are emerging widely and rapidly. This application area includes CSCW(Computer Supported Cooperative Work), remote education, tele-medicine, entertainment, and so on[2, 5].

In general, abovementioned DMMA is composed of individually controlled multiple application service elements in the OSI sense. Typical examples of application service elements are voice conference, video conference, remote display of live video, shared manipulation of data, tele-pointing and so on. A set of application service elements may be integrated for the development of a target DMMA.

Because multimedia requires various QoS(Quality Of Service) from the network communication services and multiuser environment makes the communication complex, the application service elements in DMMA require more diverse and complex communication services from the underlying network.

We will discuss about the related works on DMMA in section 2. The model of the HEART software architecture is reviewed in section 3. The proposed call model and the control of the communication services are briefly presented in section 4. Example application service element in typical DMMA is mentioned in section 5. A prototype based on the HEART software architcture; Multimedia Presentation System[8] is presented in section 6.

2 Related Works

Most of the existing works[4, 9] for the development of DMMA are application-oriented, in the sense that each application service element includes the management functions of communication services. And the application service elements of the target application are statically-configured. With this approach, every application must handle the complexities of multimedia multiuser communications, and it is very difficult to re-use the constituent application service elements for the development of other DMMA.

In platform-oriented approach, on the contrary, the services commonly used by the DMMAs are provided by the underlying platform via well-defined application programming interfaces. With this approach, the management overhead of each application service element can be reduced, and the reusability of the component application service elements is enhanced. This approach has been mainly applied to develop the multimedia multiuser communication platforms which provide complex communication services in a flexible way[1, 6, 7]. The Touring Machine System[1] is a software platform developed at Bellcore to provide an infrastructure in which applications requiring complex multimedia communications can be developed, independent of the actual network fabric used to provide transport. Heijenk and et. al.[6] enhanced the call model of the Touring Machine System to support the relationships such as multimedia synchronization and audio-activated-video among the related data transport objects. Intensive research on the development of multimedia multiuser transport protocols[7] is also aimed to support common communication services for multimedia multiuser applications.

To satisfy the design guidelines in the most flexible and efficient way, HEART software architecture is platform-oriented and layered. The bottom layer provides enhanced communication services for the application service elements of the DMMA. For this, we have developed a new multimedia multiuser call model which provides more powerful structuring capability than the existing call models. Since it provides rich set of communication control services, the individual

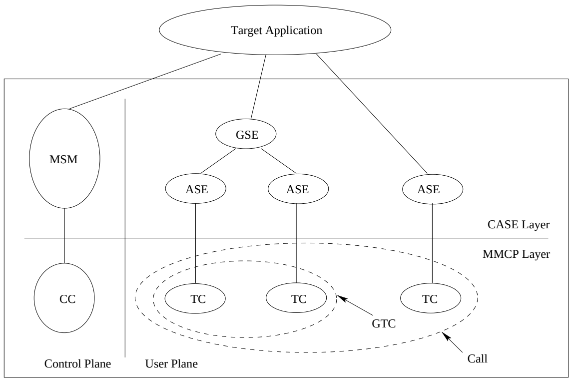

Figure 1. Functional Model of HEART

application's burden to manage complex communication services can be greatly reduced. The middle layer defines multiple common application service elements over the bottom layer, which may be commonly used by the target DMMA. In the proposed end system architecture, a target application can be developed by integrating the predefined common application service elements and additional application-specific service elements. This approach will greatly increase the reusability in the development of the various DMMAs.

3 HEART Software Architecture

3.1 Overview

HEART(Hyundai Electronics Advanced Research Testbed) is three-layered software architecture; target application layer, CASE(Common Application Service Element) layer, and MMCP(Multimedia Multiuser Communication Platform) layer. Target application is one of DMMA such as desktop multimedia conferencing. In the CASE layer various basic ASE(Application Service Element)s such as video conference application, audio conference application, whiteboard application are defined. A typical target application is realized by combining a set of service elements in the CASE layer. MMCP layer provides various complex types of communication services to the application service elements in CASE layer and/or to the target application directly.

3.2 Functional Model of HEART

The functional model of HEART in Figure 1 is divided in two planes; control plane and user plane.

In control plane, there are MSM(Multi-Service Manager) and CC(Call Controller) in CASE and MMCP layer respectively. MSM manages and controls ASEs and GSE(Grouped Service Element) in the user plane. GSE is a service element which controls a group of application service elements. A target application is composed of MSM, ASEs, and GSEs.

CC controls and manages the TCs(Transport Connections) in the user plane. TC provides communication services for each ASE. There is one-to-one correspondence relation between ASE and TC. TC consists of basic transport objects such as ATM bearer connection[3], TCP/IP connection, ST-II stream[11] according to TC type . MMCP provide communication services to each ASE by means of TC. A set of TCs can be grouped into a GTC(Grouped Transport Connection) for layer control granularity. For a target application, a corresponding call is defined at MMCP layer. A call consists of GTCs and TCs. CC can control and manage communication services at different granularity levels : TCs, GTCs, and call.

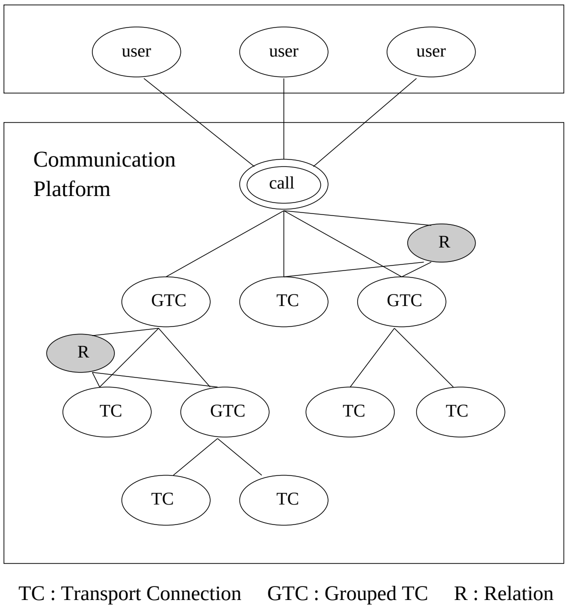

Figure 2. Structure of the HEART Call Model

4 MMCP(Multimedia Multiuser Communication Platform) Layer

In the context of the communication platform, the call model defines data transport objects required for the communication platform, and how the underlying data transport objects can be structured to support various and complex communication services for the applications. And it defines how the structured object, named as call in this paper, can be managed in a consistent manner by the applications. In this section, we present our call model[10] in HEART for DMMA.

Figure 2 shows an example of the general call structure of the proposed call model. In the call model, TC, a basic control unit, can be grouped as an integrated control unit, GTC. A GTC can be a member of another GTC. That is, a GTC groups one or more GTCs and/or TCs so as to allow applications to be able to control the GTC as well as to control the constituent TCs and/or GTCs individually.

The relation objects which specify the relationships such as synchronization and voice-activate-video among the related GTCs and/or TCs can be created as members of the GTC. And the relation objects should have two or more related objects to exist.

This nested grouping of TCs provides more powerful structuring capability than the existing call models. In our model, a call is a special GTC which represents the dynamic association between the communicating users and the communication platform, and it provides the communication capability for the application. The call is the root of a grouping hierarchy of TCs, and it is an abstraction of the communication platform instance. The global view of a call can be dynamically changed during lifetime of the call. The subordinate TCs and/or GTCs can be dynamically added and/or deleted by the permitted users. And, the communicating users of the call can be also dynamically changed(i.e., users can join and leave the call).

In our call model, the TC is the basic control unit which can not be divided into more fine-level control units. One TC is allocated for one basic service element of an application. Actual data interactions among the communicating users occur through the TCs. Further explanation of our call model and TC is in [8, 10].

5 CASE(Common Application Service Element) Layer

5.1 Overview

Various GSEs and ASEs can be pre-defined in the CASE layer. After analyzing the popular DMMAs, we choose ASEs and GSEs. And we can add ASEs and GSEs according to the DMMA requirements. The GSEs and ASEs pre-defined in HEART are : AVCS(Audio/Video Conferencing Service element), SWSS(Shared WorkSpace Service element), RPSV(Remote Processing of Stored Video service element). And MSM is an indispensable application service element because it controls and manages all other GSEs and ASEs.

5.2 MSM(Multi-Service Manager)

MSM manages the communication services of the MMCP and other common application service elements according to target application requirements. The MSM allocates TCs and/or GTCs to the common application service elements according to the registered ASE configuration information. The allocated TCs and GTCs belong to the call of the target application.

MSM consists of three functional modules as follows.

- Application Connection Function : MSM establishes and releases the connections between participating users for exchanging control information.
- Service Management Function : Activation/deactivation of ASEs and GSEs, control of the access right of each participant to ASE and GSE, and management of token in floor control are typical examples of service management functions by MSM.
- Participant Management Function : A participant can be added or deleted to a DMMA by MSM's initiative or by a user's request. MSM invite and drop the participant. And the users can join and leave the conference. MSM also supports the function of changing chairman.

5.3 AVCS(Audio/Video Conference Service element)

AVCS provides audio and video conference service to the target application. This GSE has two subordinate ASEs; AuSE(Audio Service Element) and VSE(Video Service Element). Activation/deactivation of AuSE/VSE is requested by MSM. Audio data is transmitted between participants by AuSE through the TC established by MMCP. Audio control function is executed by AuSE. Video is handled by VSE in the same manner.

AVCS consists of three functional modules as follows.

- Service Management Function : AVCS activates and deactivate AuSE and VSE.
- Stream Management Function : Stream means path between TC and audio/video service endpoint which handles the audio/video device. AVCS establishes, releases and manages the stream.
- Floor Control Function : This function enables participants to pass the token. The participants who hold the token can transmit the audio/video data.

5.4 SWSS(Shared WorkSpace Service element)

SWSS provides application sharing and workspace sharing. The shared application includes whiteboard, tele-pointer and other cooperation-aware or cooperation-unaware applications. SWSS in our architecture has three subordiante ASEs : WBSE(WhiteBoard Service Element), TPSE(TelePointing Service Element), ASSE(Application Sharing Service Element). SWSS activates and deactivates WBSE, TPSE, and ASSE when request from MSM arrives. WBSE provides telepointing, telemarking, and telewriting within the shared work space. TPSE enables the participants to share the telepointer. ASSE provides the participants with the cooperation-aware and cooperation-unaware application sharing service.

SWSS has four modules as follows.

- Service Management Function : SWSS has the function of activating and deactivating WBSE, TPSE, and ASSE.
- Connection Management Function : SWSS can request the connection to its subordinate applications.
- Event Handling Function : This function broadcasts the event of the local application or the request from the local application to the corresponding module of other participants.
- Floor Control Function : This function is similar to that of AVCS.

5.5 RPSV(Remote Processing of Stored Video)

RPSV is the application service element which provides the video multicasting service to the participants. It displays the video data stored in a video server or local disk of some participant's computer, to all participants. Authorized participants can control video data display; play, fast forward, rewind and so on.

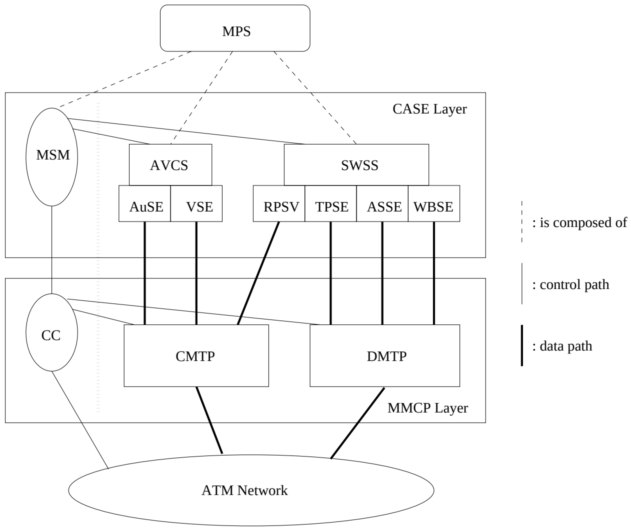

Figure 3. MPS Architecture

6 MPS(Multimedia Presentation System)

MPS(Multimedia Presentaion System) is developed as the first DMMA based on the proposed HEART architecture. In this application, multimedia data is shared by participants using presentation sharing.

MPS has the following GSEs and ASEs of CASE layer as shown in Figure 3 : MSM, AVCS, SWSS, and RPSV. The user requests are transferred to MSM for execution. AVCS is responsible for the management of audio/video data. SWSS enables the participants to share the application and workspace. CC establishes and releases the TC. CC also adds or deletes a user to/from corresponding TC. DMTP(Discrete Media Transport Protocol) and CMTP(Continuous Media Transport Protocol) are specified at the MMCP layer to transmit the non-real-time data and real-time audio/video data respectively.

Every ASE and GSE is implemented as a UNIX process. We adopt socket and message queue for interprocess communication mechanism. MPS is implemented over laboratory-level ATM network. The software modules are implemented in C using X Windows and Motif when applicable.

7 Conclusion

In this paper, we presented three-layered HEART software architecture to support distributed multimedia multiuser applications such as computer supported cooperative working, desktop multimedia conferencing, and tele-medicine. Our architecture is composed of the MMCP(Multimedia Multiuser Communication Platform) layer, the CASE(Common Application Service Elements) layer, and the target application layer. The MMCP provides structured and dynamically-configured communication services between participants to the individual applications of CASE, and the CASE layer supports various types of basic application service elements which can be flexibly configured and integrated for the development of target applications. The target application layer is the final application which has a direct interface with the user. The proposed HEART architecure will help to develop various and general multimedia multiuser applications flexibly by the layered architecture. MPS(Multimedia Presentation System) is developed as our first target application.

References

[1] M. Arango and et. al., "The Touring Machine System," Communications of the ACM, Vol.36, No.1, pp. 68-77, Jan., 1993.

[2] H. Armbruster and K. Wimmer, "Broadband Multimedia Applications Using ATM Networks : High-Performance Computing, High-Capacity Storage, and High-Speed Communication," IEEE JSAC, Vol.10, No.9, pp. 1382-1396, Dec. 1992.

[3] The ATM Forum, ATM User-Network Interfaces Specification Version 3.0, Prentice Hall, 1993.

[4] M. Chen and et. al., "Software Architecture of DiCE : A Distributed Collaboration Environment," IEEE Multimedia '92, 1992.

[5] W. J. Clark, "Multipoint Multimedia Conferencing," IEEE Communications Magazine, Vol.30, No.5, pp. 44-50, May, 1992.

[6] G. J. Heijenk and et. al., "Communication Systems Supporting Multimedia Multi-user Application," IEEE Network Magazine, Vol.8, No.1, pp. 34-44, Jan./Feb., 1994.

[7] L. Henckel, "Multipeer Transport Services for Multimedia Applications," Proceedings of HPN '94, 1994.

[8] MPS(Multimedia Presentation System) Service Requirement Specification, Technical Report HEART-MPS100-2.0, Multimedia Lab., Dept. of Computer Engineering, Seoul National University, 1995.

[9] T. Ohmori and et. al., "Multiparty and Multimedia Desktop Conferencing System : Mermaid," IEEE Multimedia '92, 1992.

[10] S. Park and Y. Choi, "A Call Model for Multimedia Multiuser Communication Platform," IEEE ICNP(Intern. Conf. on Network Protocols), pp.305-312, 1995.

[11] C. Topolcic, "Experimental Internet Stream Protocol(Version 2)," RFC 1190, NIC, SRI International, Oct., 1990.