The Application of TINA in the MESH Project

Marten van Sinderen and Luís Ferreira Pires

Centre for Telematics and Information Technology, University of Twente, PO Box 217
7500 AE Enschede, the Netherlands
{sinderen,pires}@cs.utwente.nl

Abstract. This paper discusses the application of TINA concepts, architectures and related design paradigms in the MESH project. MESH adopted TINA as a means to facilitate the design and implementation of a flexible platform for developing and providing interactive multimedia services. This paper reports on the activity in which the TINA specifications have been studied and implementation options for the TINA components have been selected. This paper also discusses the role and position of protocols and objects in this context.

1 Introduction

User demands with respect to telecommunication services are becoming increasingly stringent. These demands are not only related to functionality, performance and capacity, but also to reuse, adaptability and composability. That is, telecommunication systems should not only satisfy current requirements, but they should be designed in such a way that they can be easily maintained and modified to satisfy future requirements as well.

This challenge was taken up by the MESH[1] (Multimedia-services for the Electronic Super Highway) project. MESH aims at the development of a flexible distributed services platform to support some specific business domains, including education and health care. MESH is the continuation of the PLATINUM (Platform providing Integrated-services to New Users of Multimedia) project [3]. In PLATINUM a platform consisting of an ATM-based transport system with multiparty connection support, a middleware layer providing multiparty/multimedia sessions, and a set of application components has been developed. While re-using these results as much as possible, MESH should achieve higher adaptability and flexibility than the PLATINUM system, and should link up with current standards developments.

After surveying current developments in standards for multimedia telecommunication services, we have concluded that the requirements above can be properly addressed by following an approach such as the one taken in TINA (Telecommunica-

[1] MESH is carried out by a Dutch consortium of research institutes, industry and end-users and is partially sponsored by the Dutch Ministry of Economic Affairs. The project started in December 1996 and will end in December 1998.

tions Information Networking Architecture) [4, 9]. The TINA approach assumes the availability of a Distributed Processing Environment (DPE), which supports the development, deployment and operation of interacting objects, independent of the operating system, software and hardware of the different computing nodes. In line with the TINA approach, we decided to use CORBA (Common Object Request Broker Architecture) [7, 8] as the DPE for the MESH platform.

The particular combination of ingredients (ATM, TINA and CORBA) makes it necessary to consider two different paradigms for distributed system design, viz. a protocol-centred and an object-centred paradigm. The protocol-centred paradigm has traditionally been used by the telecommunications community. It focuses on the rules for message exchange between (protocol) entities that allow the support of some defined cooperative behaviour or service. The object-centred paradigm originated in the computing community. This paradigm is primarily concerned with the operations that can be invoked on (remote) objects and their possible returns. Since the two paradigms represent different cultures and comprise different, but partially overlapping concepts, their combined use in a single project may lead to confusion and inconsistent (intermediate) designs, unless the paradigms are precisely characterized and mutually related [2].

This paper discusses the application of TINA in the design of the MESH platform. It also discusses the role and position of protocols and objects in a TINA-based platform.

The remaining of this paper is organized as follows: Section 2 discusses the high-level decisions taken in the design of the MESH platform and their implications; Section 3 presents the MESH platform architecture; Section 4 discusses the role of objects and protocols in this architecture; and Section 5 draws some final conclusions.

2 Design Decisions and Implications

The Platinum system was limited in the sense that it was based on a single network technology, did not conform to open standards, did not provide a clean separation of network and middleware/application concerns, and was not easy to extend or modify. The main architectural challenge for the designers of the MESH platform was to provide an open and more flexible architecture through the adoption of TINA concepts, principles and components, and to re-use as much as possible of the Platinum results.

TINA is an open architecture consisting of a set of concepts, principles and components for the development, deployment and operation of telecommunication systems. Re-use of specifications and software, high availability, interoperability and flexibility are crucial requirements for advanced telecommunication systems. For this reason, TINA is based on object-orientation, distributed computing and other standards from the telecommunications and computing industries.

TINA-compliant systems are distributed computer systems, consisting of interconnected *computing nodes*. Different computing nodes in a TINA system may be produced by different manufacturers, which implies that they may use different hardware and software technology. *Telecommunication applications* therefore can assume the availability of a *Distributed Processing Environment* (DPE), which allows one to

develop and deploy a collection of interacting objects that implement the applications, independent of the operating system, software and hardware of the computing nodes, and the network technology used to interconnect these nodes.

TINA identifies different sub-architectures for different aspects of telecommunication systems, and uses different models to completely define these sub-architectures. For the development of the MESH platform, the crucial sub-architectures were the service architecture (covering telecommunication applications) [5] and the network architecture (covering data transport) [6]. The most important models were the information model and the computational model. These architectures and models were also used in the evolutionary path from the PLATINUM system to the MESH platform. In this path we had to find a proper mapping from the PLATINUM components and models (e.g., multiparty/multimedia sessions) onto TINA architectures and models.

The MESH platform uses CORBA as DPE. When this choice was made, the available commercial CORBA implementations were not capable of supporting interactive multimedia services[2]. Therefore, multimedia support in the MESH platform had to be realized outside CORBA, using PLATINUM components.

3 Platform Architecture

The MESH platform architecture basically consists of the TINA service and network architecture, limited by some specific choices. This section illustrates the application of TINA, the use of CORBA and the encapsulation of PLATINUM components.

Fig. 1 illustrates a multipoint-to-multipoint relationship (service session) between service users (parties) that can be supported by the MESH platform, according to the TINA service architecture and using the information model concepts. Components are able to establish, change and dissolve relationships through the manipulation of information model concepts.

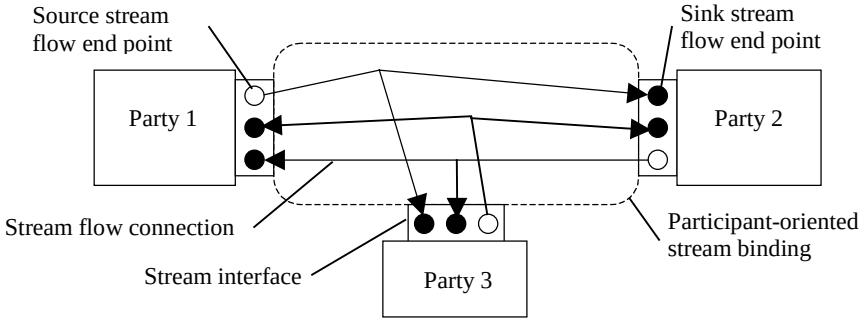

Fig. 1. Example of mp-mp service session

[2] IONA Technologies recently announced an extension of Orbix, called OrbixMX, which supports multimedia and is ATM-aware. OrbixMX is expected to be available in Q3 98.

Fig. 2 illustrates the distribution of service components supported by the MESH platform, according to the TINA service architecture and using the computational model concepts. The distribution is based on the distinction between access to services and usage of services, and the existence of domain boundaries. Component interactions that cross domain boundaries require conformance to the external interfaces specified in the TINA service architecture. Although TINA also specifies component interfaces that are used by other components in the same domain (internal interfaces), conformance to these interfaces is not mandatory.

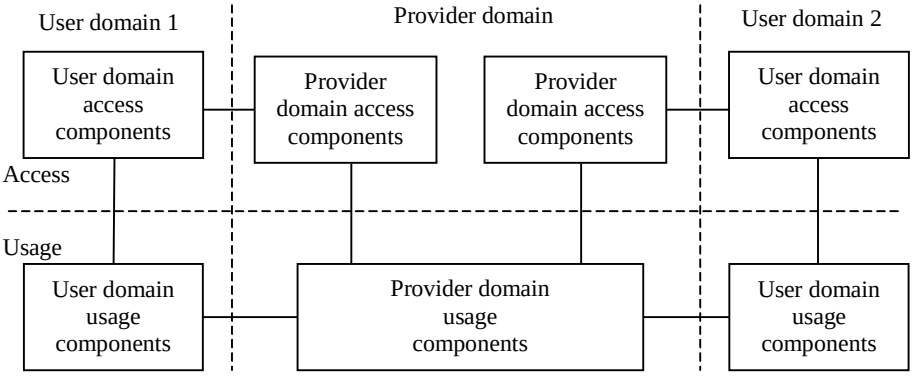

Fig. 2. Distribution of service components

One of the user domain usage components is the *service session User Application* (ss-UAP). The ss-UAP of the MESH platform consists of three sub-components (see Fig. 3): a generic component for interfacing with the provider domain usage components, a service specific component that contains the user applications and also interacts with the user domain access components, and a medium realization component that provides stream interfaces for the exchange of continuous data (audio and video).

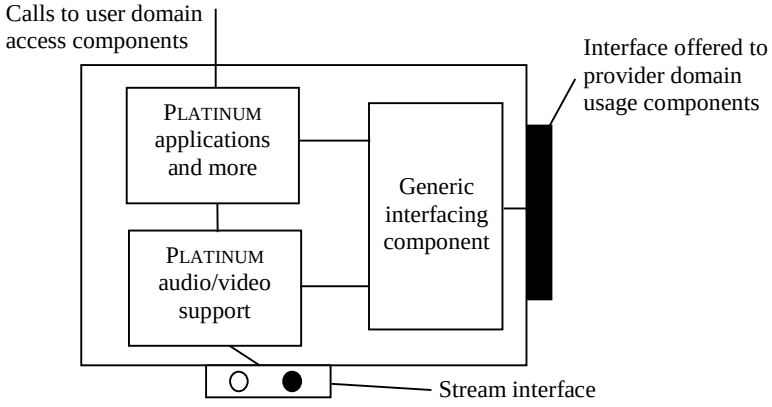

Fig. 3. Decomposition of the service session User Application component

The medium realization component is necessary since the CORBA implementation that we use does not support stream interfaces. This component re-uses the PLATINUM middleware layer functionality for linking network devices and multimedia user interface devices within a terminal. An alternative medium realization component, based on the H.323 series of standards [1], is under consideration. The service specific component also re-uses part of the PLATINUM system, viz. the application components (e.g., a conference management application with shared whiteboard). These applications need the stream support provided by the medium realization component.

The MESH platform uses CORBA to support the interactions between its service components, provided these interactions are based on operational interfaces. CORBA, in turn, uses the Internet transport system (TCP/IP) to exchange messages between different ORBs (see Fig. 4). The PLATINUM transport system is used for the exchange of continuous data through stream interfaces.

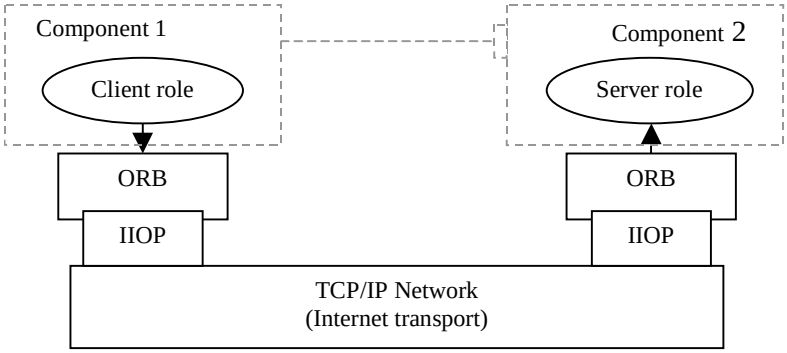

Fig. 4. Component interaction using CORBA

4 Protocols and Objects

The TINA architecture isolates service issues from network issues, such that telecommunication applications can be designed and deployed independently of the network technology used to support these applications. The interactions between the application (service) components, and their constituting objects, are supported by the DPE. The DPE, or actually its ORB part, hides the protocols that provide the interaction support. This allows interactions to be specified at the application level in terms of (patterns of) invocations of methods on interfaces of components.

The DPE boundary reflects the state-of-the-art with respect to middleware technology: below this boundary, functionality is provided out-of-the-box, while above the DPE boundary, functionality is user-defined, although application frameworks may facilitate the development of user-defined functionality. We expect the DPE boundary to shift in time. For example, next generation DPEs will probably support stream interfaces and multipoint communication. In general, recurring patterns of interaction are potential candidates for future DPE (ORB) extensions, and complex coordination

of components may be replaced by simpler interactions if protocols that provide richer interaction support are incorporated in the DPE.

5 Final Remarks

The MESH platform architecture presented in this paper is an intermediate result of the MESH project. The platform is currently being implemented. This implies that the architecture may still be modified, extended or simplified, depending on the problems and opportunities encountered in the implementation process. For example, tools and technologies for WWW programming will be used to create a wide range of additional applications, without losing the benefits of TINA and TINA compliance.

TINA turned out to be useful in the achievement of the project goals. However, the interpretation and application of TINA specifications is not straightforward: the TINA specifications are quite complex and general, forcing designers to make choices and interpretations, such that the resulting system complies with the intended requirements in a reasonable time scale. Furthermore, TINA was still under development during the design of the MESH platform, and consequently the specifications were not always complete and consistent. In retrospect, however, we feel that it is important to exercise the TINA concepts in order to get a better understanding and control of complex telecommunication systems and services.

References

1. International Telecommunication Union. Audio/video communication. Recommendation H.323v2 (forthcoming), ITU-T, 1998.
2. M. van Sinderen, L. Ferreira Pires. Protocols versus objects: can models for telecommunications and distributed processing coexist? Sixth IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'97), IEEE Computer Society Press, 1997, pp. 8-13.
3. M. van Sinderen, P. Chimento, L. Ferreira Pires. Design of a shared whiteboard component for multimedia conferencing. Third International Workshop on Protocols for Multimedia Systems (PROMS'96), Dpto. de Publicaciones de la ETSIT, Ciudad Universitaria, Madrid, Spain, 1997, pp. 1-16.
4. Telecommunications Information Networking Consortium. Overall concepts and principles of TINA. Version 1.0. February 1995.
5. Telecommunications Information Networking Consortium. Service architecture. Version 5.0. June 1997.
6. Telecommunications Information Networking Consortium. Network resource Architecture. Version 3.0. February 1997.
7. S. Vinoski. CORBA: integrating diverse applications within distributed heterogeneous environments. IEEE Communications Magazine, 35(2):46-55, Feb. 1997.
8. http://www.omg.org/
9. http://www.tinac.com/