AN ENGINEERING PLATFORM FOR MULTIMEDIA SERVICES

Paul L Reynolds and Peter W Sanders
Network Research Group
School of Electronic, Communication & Electrical Engineering
University of Plymouth
Drake Circus
Plymouth, Devon PL4 8AA
UK
E-mail: Reynolds@sc.plym.ac.uk

ABSTRACT

Multimedia services carried by telecommunication networks have been the target of many research projects, and different approaches have been followed by different research consortia over recent years. The ETSI - Intelligent Networks (IN), [1] the ITU - Open distributed Processing (ODP) [2] and the Telecommunication Information Networking Architecture Consortia (TINA-C) [3] are a few of the different groups involved at present in the research to support multimedia services in telecommunications.

This paper details some of the work currently in progress in a European Union RACE project [4] that will, once implemented, support multimedia services within a global environment. An engineering platform, based on an object oriented approach, is proposed. The main objective of this platform is to support an engineering viewpoint of telecommunications, with the aim of providing the infrastructure required to allow the design, implementation and testing of generic multimedia services. This will provide facilities for service creation, service deployment, service maintenance, service operation and service management. The structure of this paper commences with an introduction to define the boundaries of the platform and definition of terms. An implementation mechanism and possible methodology are then identified. The distillation of application objects is discussed and a mechanism for their decomposition into engineering objects is detailed. Finally, a possible engineering platform for multimedia services is given and conclusions are drawn.

1: INTRODUCTION

1:1 Multimedia Services Engineering (MSE)
MSE is the engineering of integrated multimedia services. It involves the integration of different types of information for a wide range of services, the integration of the management and control of these services, their integration to provide more complex arrangements, the development of user interfaces that support and co-ordinate the service provision and its use, and the integration of objects and components during the service development process. The key to MSE is the efficient deployment of multimedia services in an open market environment. This is achieved by the decomposition of these services into reusable components, which are themselves an abstraction of the telecommunication network resources necessary to form part of the services.

1:2 Engineering Platforms
An engineering platform is required to support MSE in the designing, implementing and testing of multimedia services, and is a tenable expansion of the abstracted engineering viewpoint. It is concerned with the types and interaction of objects identified as required by the service for their realisation. The platform requires a development environment which supports multimedia interfaces, a database of reusable service components, and a test environment that comprises network simulations plus test data. Any simulated service that is produced can then be bound to real world components and thus into the network to create actual multimedia services. The process of service development can be performed on a centralised computational platform, however, the binding from generic software components, of a simulated model, to the specific physical components of the telecommunication network, requires further consideration. Here the physical components are more likely to be distributed, probably heterogeneously and comprise a mixture of hardware and software. Some transformation objects, that bridge the gap between generic and specific components, may also be required. The distribution of physical components is also likely to require some software components at appropriate remote locations to provide the proper communication support

between these locations. When a service is deployed and operational, there is still a need for maintenance, which may require changes to the design of the components that implement the service. A service also has to be run and controlled which will require interactions from users, customers, as well as operators.

1:3 Telecommunications

Telecommunications networks are by their ubiquitous nature more global than the traditional locally distributed IT systems. Object oriented distribution is a natural step forward, where objects represent an abstraction of actual telecommunications resources, either software, hardware or a combination of both. In addition, these networks, because of their complexity introduce many transparency problems that must be solved to provide flexible and user friendly multimedia services. These transparencies include: naming, protection, sharing and translation issues resulting from network object heterogeneity; accounting and resource management created by object autonomy; and process synchronisation, consistency, error recovery and control, resulting from object distribution. The complexity of federated telecommunication network control may restrict the full power of object orientation. For example the benefits of inheritance, which facilitates reuse, cannot be controlled without a significant degradation of network performance.

Telecommunication networks can be considered a distributed collection of interaction components, composed of objects, communicating with each other using messages addressed to defined interfaces. The telecommunication network autonomy is maintained because components can change independently and transparently provided they maintain their interfaces. These interfaces accommodate heterogeneity because messages sent to distributed components depend only on their interfaces and not on the internal set of objects.

As telecommunications tends to be conducted in a process synchronous mode, the ODP mechanisms for unwanted distributed transparency can lead to an unacceptable performance overhead. Therefore, in such environments, it is desirable that the support should provide for selective use of the transparencies [5]. In addition, the telecommunications management system may want to maintain control over distribution or participate directly in its provision, e.g., by specifying a recovery procedure for certain failures. Transparency in telecommunications must, therefore, be 'translucent' and allow for such influences by way of a generic mechanism.

Systems based upon ODP-RM will become more and more used for multimedia applications. Multimedia brings with it special problems that telecommunications platforms must cope with. These include factors on how to handle:

- complex data types;
- synchronisation between streams of voice, data and video;
- the integration of the switching of these different streams;
- the trading and binding processes for objects comprising these systems.

2: TELECOMMUNICATION APPLICATION OBJECTS

In ODP terminology an object is a unit of computation while in telecommunications an object is considered a unit of system, i.e. includes hardware. Also, in ODP terminology an object has one interface that is the point of provision and use of the object, while in telecommunications, management is a prime issue not directly addressed by ODP. Here a managed object is defined as any hardware or software component whose behaviour can be controlled by a management system. From an engineering viewpoint the interfaces give a view or way into an object. The interfaces are an access to a set of operations or functions concerning that service object. The interface to a monitoring object measures state changes and telecommunication monitoring functionality, e.g. account metering (start, stop, duration, etc).

However, while the model fits the definition for normal computation objects it does not include management. This exists in telecommunications as a separate network to control network elements and multimedia services. The interfacing to a management object controls resource allocation, cost, violation control and conflict resolution. Thus the model must be improved by the integration of the management into the service object definition. .

In this case, a telecommunication object (t-object) can be said to have two interfaces as in figure 1:

- An operational interface that supports the normal information-processing operations, fulfilling the main purpose of the service provided by the object.

- A management interface that supports monitoring and control interactions with the management system.

administration
billing
control commands
status request
monitoring information

Management Interface TELECOMMUNICATION MANAGED APPLICATION OBJECT Operational Interface

information
processing
operation

Figure 1. A MANAGED OBJECT

In extending the ODP model it is possible that through the definition of a suitable repertoire of transparency objects the ODP infrastructure can be made sufficiently flexible to meet the unique demands of telecommunications. Such transparency objects provide a mechanism to deal with organisation, time, technology etc. A further way in which ODP can be developed is that the 'user' could be allowed to select their own required transparencies. A platform should detail the mechanism of providing user selection of transparencies and provide appropriate presentation techniques to allow the tailoring of such transparencies by the user.

The components so far have defined what is seen at an Operational Interface. This is equivalent to the operational interface in ODP's Computational terminology that describes the services offered to, and required from its environment. A computational operational interface thus describes a service offered by the object in telecommunications as well. It has four components:

- A set of signatures;
- Quality of service and transparency attributes the object requires;
- Set of actions resulting from activity at the interface;
- Role, i.e. client or server.

ODP describes how these application objects can be supported by an engineering platform, i.e. the application object can be mapped into a set of engineering objects.

3: ENGINEERING PLATFORMS

A multi stage process can be used to develop the engineering platform. The first stage is to define the object model which captures the static structure of the system. In defining the object classes it is important that they should contain terms familiar to both the application (telecommunications) and the viewpoint (engineering). The decomposition of the application object into object classes depends largely on judgement. Rumbaugh et al [6] provides one mechanism for the distillation of object classes and object associations by the distillation of nouns and verbs from a textual definition of a problem. This process has proved to be successful in distilling a number of telecommunication object classes for RACE EURSAF [4] but these did need to be augmented by knowledge of the domain. The next stage comprises the grouping together of object classes with common functionality and the rejection of redundant and irrelevant classes into components or a single unit of telecommunications network. It is then necessary to define the transparencies to support these components on actual pieces of network, i.e. switches and transmission equipment, and abstract these pieces of network as nucleus objects and thence identify the communications necessary to support cluster to component communication.

The resultant object model can be further refined into levels, each with its own job function. To complete the platform definition it is necessary to define the dynamic relationships between the objects' classes in the object model.

An engine responsible for the selection of the objects that comprise the service can take the form of a service window. Here the window is responsible for the selection of the required objects to compile a new service request as shown.

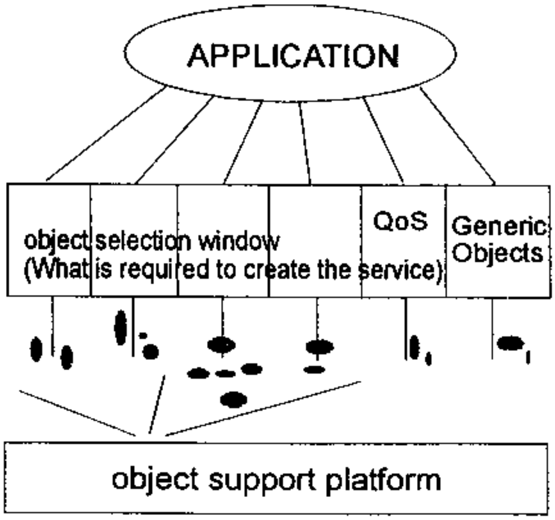

Figure 2. OBJECT SELECTION ENGINE

The service window may then operate as follows. A service application can be considered to comprise a fixed number of objects, some of which are used to make up individual services. Simple multimedia services would therefore not need a full function capability set and hence some of the selected objects must be termed null objects. Management of such a system would be localised and hence the speed of service provision could be increased. Different window profiles can be envisaged dependant upon the class of service user or provider.

4: PLATFORM OBJECTS

Each service can be considered to be a sequence of events between t-objects and a number of sequences dependant upon conditions, and can be divided.

There should be a number of t-object events common to a service, or it is possible to develop sequences within a group of multimedia services. For example:

- A request for service using one or a number of communication forms from the user including:
Retrieve (EDI, FTP, Documents & Files);
Receive (EFT);
Speech, (PSTN, ISDN);
Send mail, (Messaging);
Visual (Video conference), etc.
- A general response - the service is available on that station insert authentication smart card;
- Authentication of user - PIN, visual, speech, thumb;
- Access control response.

An application service, e.g. e-mail, video conferencing, etc., will comprise a number of telecommunication objects and object interactions. A service machine takes a request for a service and determines the optimum way of satisfying the request. In the process, objects will be created or destroyed. Examples of generic telecommunications requirements and t-objects are:

- Availability
- Business
- Connection
- Cost
- Control
- Database
- Flexibility
- Identification
- Legal
- Language
- Security
- Statistics
- Process
- Transparency

Each generic t-object has attributes accessible by their interface and selectable and changeable. Examples of the attributes are as follows:

- AVAILABILITY:
Immediate
Priority
Pre-emption
- CONNECTION:
Capacity
Number
Location
- COST ACCOUNTING:
Time
Capacity
- FLEXIBILITY:
Broadcast
Communication
- SECURITY:
Integrity
Confidentiality
Survivability
- PROCESS
Real
Store & Forward

Thus a model of a t-object will be as shown below:

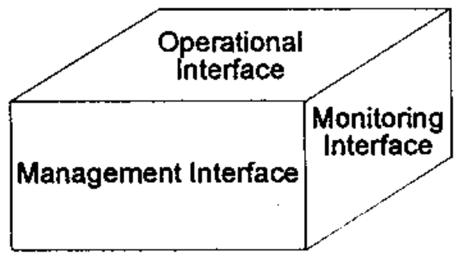

Figure 3. t-OBJECT

The t-object can now be used with other objects defined by ODP, and modified for the management, as in the prescriptive model selection below:

- generic t-objects applicable to arbitrary service requests
- t-objects decomposable and rebuildable to give new multimedia services
- every t-object must provide distribution transparency
- t-objects are not dependable on technology and service application
- t-object can have one to one service request
- t-objects can contain objects
- t-objects are location independent.

In developing the platform it is important not to constrain unnecessarily the technology of telecommunications by specifying irrelevant local detail of their implementation or by prescribing unnecessary features that are internal to them.

The t-objects or components that comprise the platform should contain the following:

- The behaviour of the t-object and the way this behaviour must be achieved.
- A list of the primitive terms used in its specification when making statements of its behaviour.
- A conformance statement.

A hierarchal layering of the object into an Engineering Platform is shown in Figure 4.

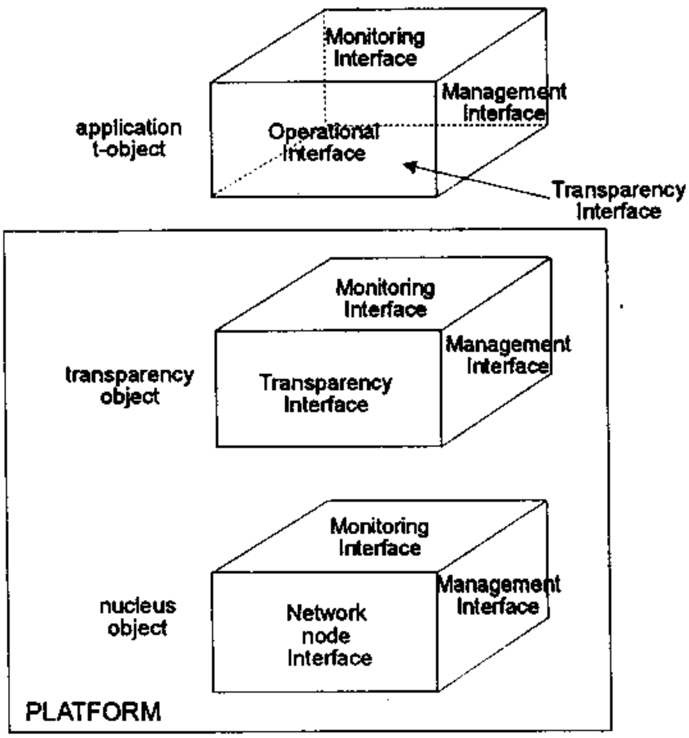

Figure 4. OBJECT VIEW OF A MSE ENGINEERING PLATFORM

There will be a number of specific activities for the individual service, but some common to other multimedia services. By the analysis of existing multimedia services it will be possible to determine a common set of events and sequences from which a standard interface for the user can be developed. The user agent will take information from this interface and set up a sequence of events using the management and the appropriate selection of objects. These will be developed to provide a group of events to provide the overall service required at the time of invoication.

Additional multimedia services can be added or developed by adding appropriate object to the service unit, but making use of existing ones if possible. The network agent, operating to the standard network interface will select the appropriate network facilities necessary to carry out the required communication. The most economical will be selected. The necessary terminating and control units for the individual facilities will be provided as and when that facility is provided to the interface. All activities will be transparent to the user. Different communications facilities will be patched if the user changes the service requirement during a session.

5: PLATFORM IMPLEMENTATION

The realisation of a platform for multi-media services carried by telecommunication networks need to be cognisant of both those underlying heterogeneous networks and the service applications that need to be deployed onto the networks.

The heterogeneous networks will comprise a mix of technology ranging from 64 kbit/s digital circuit switched channels that form the basis of the public telephone network, B-ISDN up to 2 Mbit/s being a building block of the Intelligent Network, and Asynchronous Transfer Mode networks capable of switching and transporting at 155 Mbit/s. These technologies sit uncomfortably with store and forward data networks including X25 packet switching and the INTERNET. The applications environment will comprise of a multitude of offerings generated by providers who have no knowledge of the underlying technology supporting telecommunications.

An implementation of an engineering platform for multimedia service thus will comprise a minimum of three layers, as shown in figure 5.

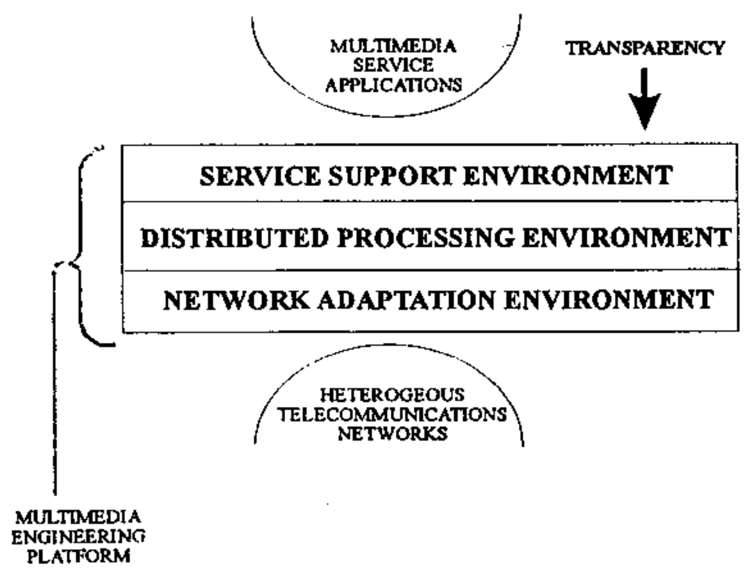

Figure 5. MULTIMEDIA ENGINEERING PLATFORM

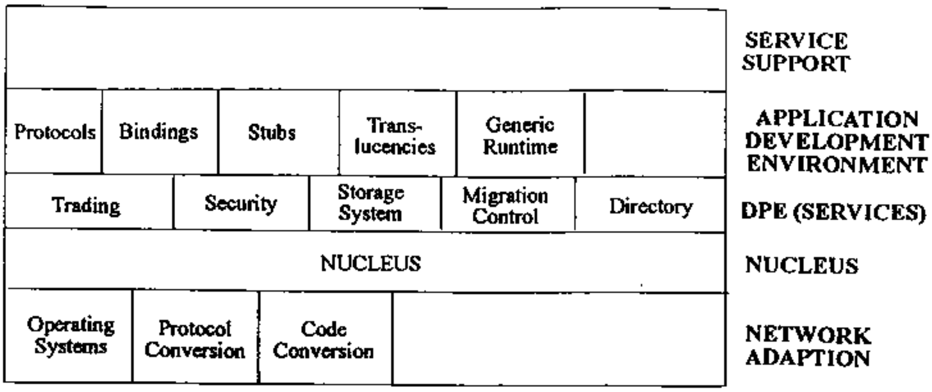

Figure 6. MULTIMEDIA PLATFORM (MINIMAL)

The Network Adaptation layer forms the first layer and makes the heterogeneos telecommunications environment transparent to the service environment. The second layer is a Distributed Processing Environment (DPE) which supports distribution and the transparencies necessary for distribution. The third and highest layer is the Service Support Environment which helps in the translation of application into deployed service objects. A minimal platform is defined as one which needs to beprovided at each node within the telecommunications networks forming part of the transport system for multimedia services. This minimal platform based upon this three-layer model is shown in figure 6, showing the Nucleus and Application Development Environments. The platform layers give the following functionality:

Network Adaptation
As a minimum there is a need to support differing operating systems, protocols and codes and provide a common interface to the nucleus.

Nucleus
Ties the platform to the local operating system and provides the mechanisms for inter nuclei communication.

DPE Services
Trading, that is the finding and combining of t-objects to invoke a service. Security, both of platform access and of t-object access. Storage, maintaining data integrity of the distributed database. Control of migration of t-objects. Directory system which maintains a list of currently supported t-objects.

Application Development Environment
Provides the tools necessary for service creation.

Service Support
Providing a 'common interface' and hides the complexity of the applications development environment to the application providers.

6: MULTI-MEDIA PLATFORM ISSUES

The underlying telecommunications network is unique in terms of scale and complexity; handling more than 1000 billion calls every year plus an immense amount of data; 800 million terminals in use in some 230 countries around the world, most of which are directly dialled; all operating in real time within a federated arrangement of networks [7]. These requirements place onerous requirements on the deployed platform which include:

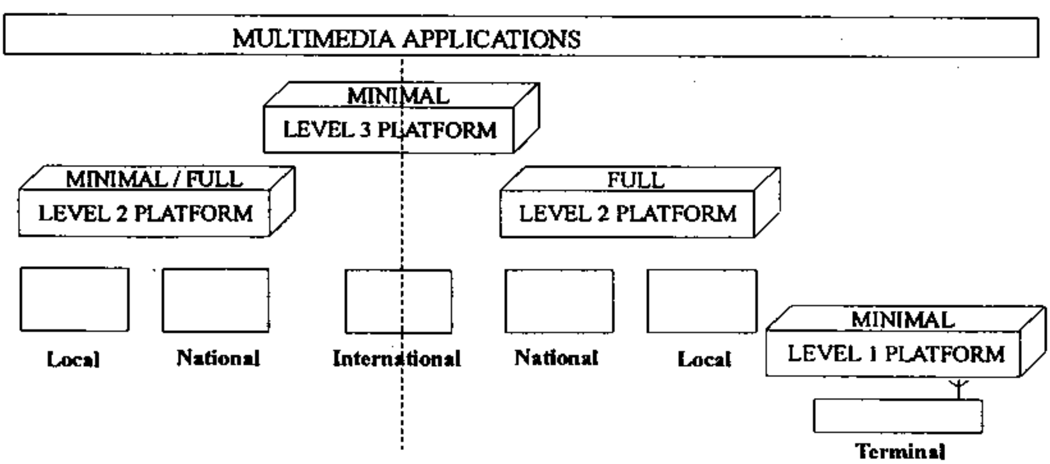

Figure 7: HIERARCHICAL MULTIMEDIA PLATFORM

Platform Distribution Optimisation:
The criterion for an optimal distribution of t-objects needs to be measured in terms of t-object communication. Based upon the use of services, it is necessary to optimise the deployment of the t-objects by their migration to nodes which have the greatest communication requirement. However, the intrinsic value of a software rich t-object may require a restriction of its mobility within a federation boundary.

Real Time Code Generation:
The services envisaged are one off and require the real time generation of service code. This can be achieved by the mechanism that provides for the quick binding of t-objects into a multi-media service. Early binding of t-objects before service invocation is not recommended as it requires advance knowledge of the service required by the user. The Object selection engine is able to facilitate the real time compilation of the service (See section 3 above). It may also be necessary to define a service or comprise a fixed number of t-objects thence to facilitate service testing before usage. This will mean that if certain service components are not required, for example security, then the place of the security component is taken by a null-t-object. Implicit in these mechanisms is a need for all t-objects to have unified interfaces.

Hierarchical Networks:
The hierarchical telecommunications network to which the Platform must interface may provide the model for the DPE. Figure 7 shows how this may be achieved. By the provision of the platform in this fashion the distribution aspects are reduced.

Unfortunately this will reduce the transparency of the platform but will improve on the time to find t-objects necessary to compile the service. One further advantage of this approach is that it allows for the deployment of platforms with differing degrees of functionality. For example, the multi-media platform at the user terminal may initially be minimal and gain functionality by the migration of required t-objects at service invocation and lose this functionality at service completion.

7:CONCLUSIONS

The design of an Engineering Platform for MSE needs to be cognisant of, but treat transparently as far as the user is concerned, the heterogeneous network that will support multimedia services. Multimedia brings the question of how to handle and synchronise complex and differing data types that comprise a single multimedia service. A platform is necessary that will 'integrate' the heterogeneous network by the use of client/server software components with special stream interfaces to cater for synchronisation. It may be necessary to provide a set of syntax and semantics at the interface and the use of late or controlled binding will be required to control the differing information flows. The use of t-objects is one way in which a MSE engineering platform could be provided.

REFERENCES

1: ETSI NA6 Report ETR 062, May 1993 IN Standards
2: ISO/IEC 10746, ITU-T X.901 - X.904
3: Barr W.J. et al. "The TINA Initiative", IEEE Communications Magazine, 1993.
4: RACE EURSAF project. Commission of the European Communities, 1995.
5: Reynolds P.L. & Sanders P.W. "Integrated Services Engineering", IEE 5th Conference of Telecommunications, Brighton, 1995, p297-301.
6: Rumbaug et al. Object-orientated Modelling and Design, Prentic Hall, 1991, Chapter 8 - "Design Methodology Analysis".
7. Rudge A.W. "I'll be seeing you", IEE Review, Vol 39, No 6, 1993.