

**Network Management Interoperability (NIM) Chain  
Guideline #4**

**TMN and TINA coexistence**

This guideline presents a set of recommendations migration architectural issues and discusses in detail the TMN and TINA coexistence options. Specifically the guideline details how to organize the coexistence of TMN and TINA identifying the needed components. Issues of gateways and architectures are addressed.

**Table of Contents**

1. Purpose of Guideline.....	3
2. Target Audience.....	3
3. Requirements to TMN and TINA co-existence .....	3
4. Recommendations .....	4
4.1 TMN-TINA coexistence options.....	4
4.2 Specific Recommendations .....	5
4.2.1 For Operators .....	5
4.2.2 For Platform Providers .....	5
5. Rationale for Guideline.....	5
6. Issues still open .....	6
7. Supporting references, trends and standards.....	6
8. Document History .....	6
1. The ACTS Experience .....	7
2. Input to Architecture .....	7
2.1 ACTS-NMF.....	7
2.1.1 The Prospect CORBA/CMIP Gateway.....	8
2.1.2 The VITAL-REFORM Coexistence and Migration Approach .....	14
2.2 TINA.....	19
2.2.1 Distributed Management Facilities.....	19
2.2.2 Services.....	20
2.2.3 Naming Principles .....	21
2.2.4 Distributed Management Broker Facility .....	21
2.3 XoJIDM.....	21
2.4 EURESCOM .....	22
3. Abbreviations.....	22
4. References.....	23

# **Network Management Interoperability (NIM) Chain Guideline # 4: TMN and TINA coexistence**

Editor: Nicholas Karatzas Algosystems S.A., Greece

Project: AC210 Dolphin

## Contributors and Participating projects

### AC208 REFORM:

Daniel Ranc Alcatel Alsthom Recherche, France

George Pavlou, UCL, U.K.

### AC003 VITAL

George Pavlou, UCL, U.K.

### AC052 PROSPECT:

Lars Bo Sorensen, UHC Ltd, Denmark

Vincent Wade, Trinity College Dublin, Ireland

## **Abstract**

This document provides guidelines for the migration architecture and presents the TMN and TINA coexistence options. It focuses on how to organize concepts and architecture in order to provide a smooth integration between existing TMN-based systems and new TINA conformant components or systems. In this respect NIM GDLN-4 is concerned with:

- TMN architecture
- TINA architecture
- mediation techniques
- OMG CORBA

The guideline objectives therefore are:

- To provide a survey of existing theoretical work on the subject and report on-going experimental results
- To provide the description of a set of migration paths and their particular architectures and technical recommendations

The work presented here is based on the experience gained by the research carried out in ACTS projects, taking into account work from other initiatives (TINA and Eurescom) and relevant standardization organizations.

## **1. Purpose of Guideline**

The main focus of this guideline is the description of architectural issues related to the now emerging introduction of TINA-based network management systems, among the existing TMN OSI environments. Options with respect to the TMN and TINA coexistence are examined. The recommendation for the coexistence options addresses the specific characteristics experiences when developing systems that incorporate both worlds. In a first step, the guideline identifies the audience to which the recommendation is focused; and then explains the rationale behind the recommendation. Issues still open are also discussed. Attached Annexes contain specific work from the relevant ACTS projects and other sources.

## **2. Target Audience**

The guideline is of benefit to TMN product and service developers, to TMN marketing strategy teams, to TMN system design teams, to system providers and telecom operators. Specific topics might also form contributions to standardization bodies, the TINA-C consortium, NMF and OMG.

## **3. Requirements to TMN and TINA co-existence**

The TMN M.3010 architecture of the ITU-T and the TINA architecture have often been thought of as tackling very different requirements of the telecom industry. However, even if this may be true to a certain degree, both paradigms are going to coexist whatsoever. TMN systems are now being deployed, while TINA systems are just emerging. As both paradigms will exist for some period of time the coexistence options should be examined. This report makes use of XoJIDM concepts to achieve the required inter-operation between TMN OSI and TINA management systems. Input from TINA-C includes:

- Basic concepts, methodology.
- Use of DPE specifications and selected DPE services implementation.
- Real-time DPE specifications and prototypes.
- Service Architecture.
- Resource Configuration and Connection Management architecture.

Useful input might come from associated or otherwise related projects, especially in ACTS:

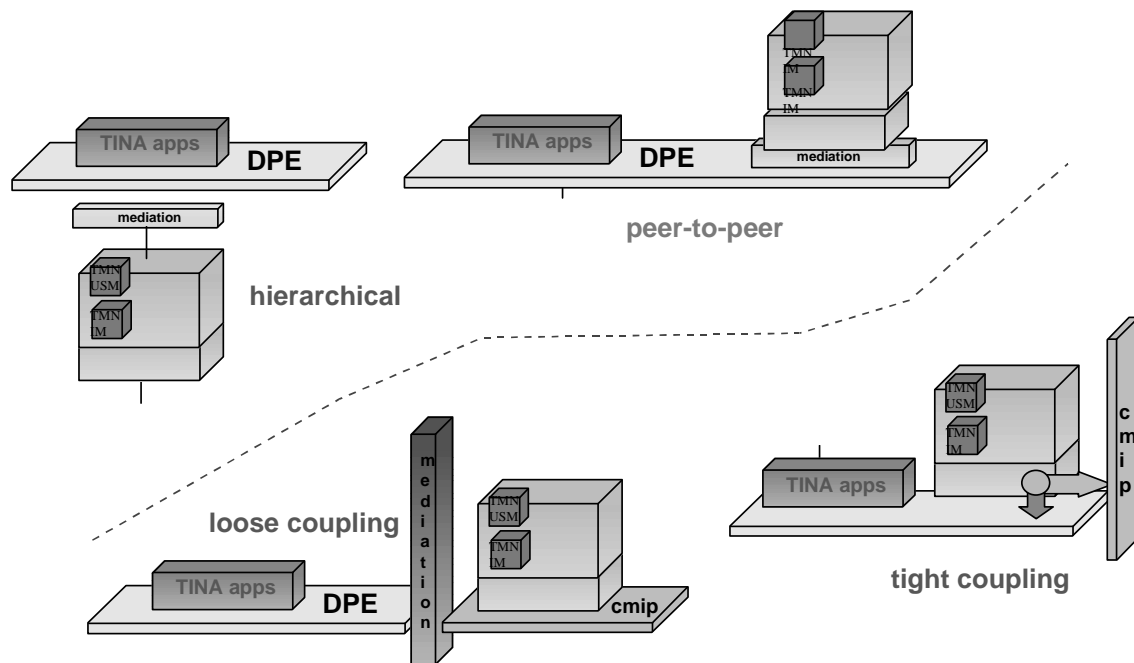
- Connection Management Architecture, specification, implementation (in ReTINA and VITAL).
- Stream interface and implementation for Video Conference (in VITAL).
- REFORM, a project making profit of many architectural features of TINA within its management system.

This document examines some of the above points taking into account a practical view.

## 4. Recommendations

### 4.1 TMN-TINA coexistence options

The possible TMN-TINA coexistence options are depicted in figure 1.



**Figure 1: TMN-TINA coexistence schema's**

The possible coexistence schema's can be clarified in hierarchical or peer-to-peer and loose or tight coupling.

1. Hierarchical. This is the schema when the TMN system acts as a slave to TINA management services.
2. Peer-to-Peer. In this schema some management functions are performed equally by TMN and TINA management applications.
3. Loose coupling. This is the schema of two homogeneous but separate worlds communicating through some unique facility which is the Mediation facility. This facility talks CMIS/CMIP at one end while the other end communicates with the specific DPE that supports the TINA management applications.

Loose coupling can moreover be envisaged from two points of view : a) a gateway point of view where a full protocol translation is undertaken and b) an ad hoc approach where only application-specific, partial translations are computed.

4. Tight Coupling. The TMN applications are made DPE compliant by extending the TMN framework. This can be achieved by including in the external communication the appropriate facilities.

## 4.2 Specific Recommendations

One can conclude that the migration path raises a number of complex issues. From the options presented above two migration paths are currently emerging. Both follow their individual approaches which will converge at a later stage.

### 4.2.1 For Operators

From the operator perspective the loose coupling scheme brings the benefit of inter-operation between existing TMN systems and TINA-based applications while preserving past investments. The existing management software is not modified. An example is for instance an operator adding a TINA-based Service Architecture on top of the existing TMN system.

### 4.2.2 For Platform Providers

The second perspective is favored by the platform providers. It follows the tight coupling path enabling interaction of the TINA and TMN worlds at the management object level.

The tight coupling scheme is implemented by an enrichment of the managed object classes of the given environment towards CORBA communication capabilities. These enriched managed objects are then able to communicate either in CMIS/P or in CORBA (following a CMIS-like protocol defined by JIDM) transparently for the managing application.

The two paradigms are different in nature and the listed coexistence options require a lot of effort to be implemented and validated in the field. The final version of this document will provide some initial validated ideas based on implementation exercises that will take place in some of the participating ACTS projects.

## 5. Rationale for Guideline

TINA applies the principles of ODP and OMG standards with respect to the needs of the telecommunication industry. This resulted in the definition of a *Distributed Processing Environment (DPE)*.

Management activities in a TINA system are modeled similarly to the classical OSI interactions between Manager and Managed objects. Communications between Managers and Managed objects are handled by the TINA DPE. The TINA management framework provides a language for the specification of managed objects: the TINA ODL (Object Description Language). The latter extends OMG's IDL (Interface Description Language) with Quality of Service (QoS) characteristics, streams and informal description of behavior and usage.

TINA's architecture is currently being finalized and its introduction will require transitional experiments that will have to accommodate, OSI management with the new TINA applications. The tight interaction scheme of «legacy» TMN applications will require a high degree of inter-operation between TINA and TMN applications. Since Network Elements produce basic information in GDMO and this information is required by TINA management applications a way to transform it from GDMO to IDL is also required.

Various architectural issues still remain open since semantic gaps between both worlds exist, such as: entity lifecycle, entity naming, place of the protocol etc. A number of coexistence strategies are possible. This strategies are examined in this document.

## 6. Issues still open

The following two issues are very important for Telecom users:

- I. **naming.** This is an important issue that arises from the fact that the TMN has adopted the FDN naming which is very powerful by being:
  - A. universal in topology and time, and
  - B. scaleable.

On the other hand CORBA/DPE naming is based on "reference" which provides a direct handling of objects within the programming languages. This creates two issues.

- C. The DPE naming is bound to the lifecycle of a specific client-server session. This implies that the naming is temporary, since it is valid for a particular session only. This also implies that there is a need for an additional complex service (broker) to overcome this limitation, adding complexity.
  - D. Scalability of this naming service is an issue, because of the "flat" nature of the reference based naming schema. To overcome this second limitation new concepts (for example the recursive naming context schema) have been defined, again at the cost of adding complexity.
- II. **dynamic behavior**
    - A. **event management.**
      1. Corba/DPE events (no scoping/filtering)
      2. TMN events (EFD, scoping/filtering...)
    - B. **synchronous/asynchronous schemes.**

CMIP can handle both synchronous and asynchronous requests (including handling of multiple replies). In Corba/DPE only one-to-one synchronous requests are possible. The OMG is currently defining an asynchronous multithreaded scheme to overcome this limitation.

## 7. Supporting references, trends and standards

The guideline is based on work that is carried out in ACTS projects (see annex 1), as well as NMF, TINA, XoJIDM, and Eurescom (see annex 2).

Detail list of references is also provided in Annex 4.

## 8. Document History

This is the fifth (final) version of the NIM #4 Guideline.

This document was previously published (version 1) at the Rennes NI Chain Workshop, March 1997.

Version 2 was published on May 1997, taking into account:

1. the new guideline structure and format
2. the new results of the NI chain meeting in Paris May 6-7<sup>th</sup>, 1997.

Version 3 was published on August 1997, taking into account:

1. based on input received from REFORM and the Dublin meeting June 20<sup>th</sup>, 1997.
2. updating the previous material based on comments.

Version 4 was published on September 1997, taking into account:

1. Input from projects PROSPECT and VITAL
2. Comments from reviewing partners

Version 5 was published on October 1997, taking into account:

1. Final editorial changes following the endorsement meeting in Brussels.

# ANNEXES

## 1. The ACTS Experience

ACTS Projects are active in defining and experimenting with this migration architecture. One should note that the work in the various projects are in an early stage, but:

- A. some work on developing interface between existing Q3 NEs and DPE implementations is on-going, and
- B. projects envision to use of legacy FCAPS implementations.

Since ACTS projects follow an implementation and trial approach, this raises some fundamental questions on:

- A. how much extend to translate the TMN paradigm to TINA
  - 1. reference vs. FDN,
  - 2. event management,
  - 3. FCAPS.
- B. how to organize coexistence of both paradigms:
  - 1. tight vs. loose coupling.

In this document a consolidated view from the various projects will be presented.

## 2. Input to Architecture

Inputs to architecture comes mainly from the experience gained in ACTS projects (REFORM, RETINA, VITAL etc.). Other input come from:

### 2.1 ACTS-NMF

The following figure summarizes the integration options that are studied by various projects in ACTS and by other initiatives, especially NMF.

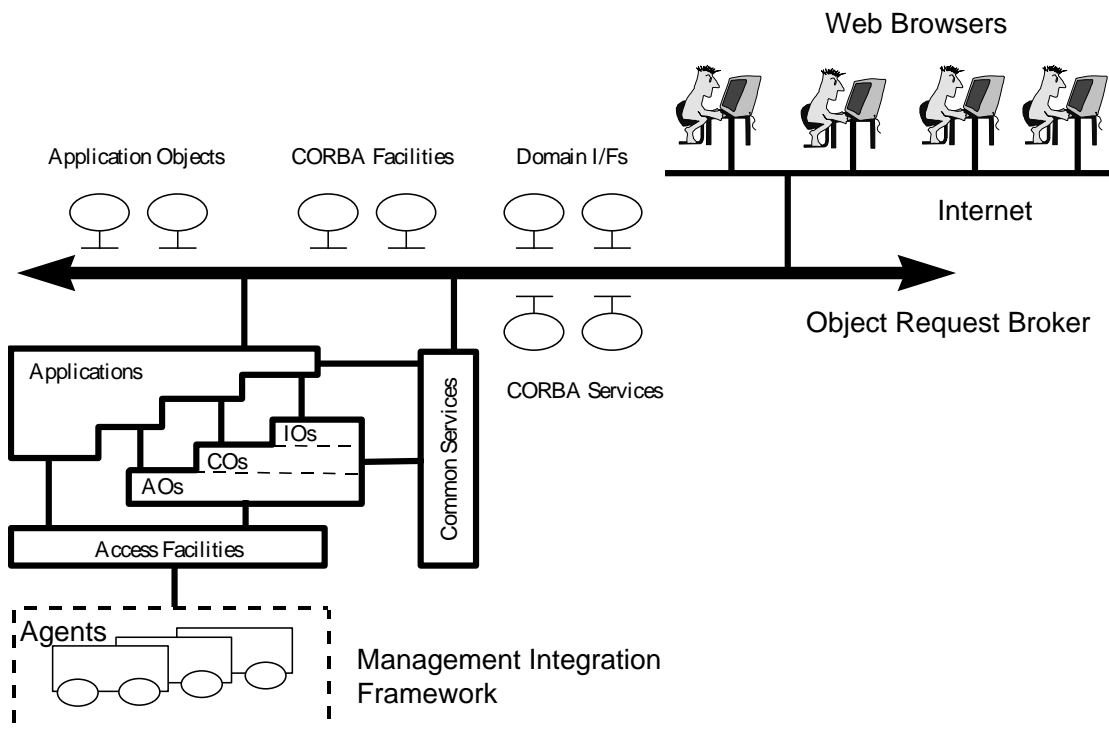


Figure A.1 Technology Integration

The Prospect, VITAL and REFORM projects have experiment using the above paradigms. Section 2.1.1 contains the Prospect developed CORBA/CMIP Gateway and section 2.1.2 the VITAL-REFORM approach for TMN and TINA coexistence and migration.

### 2.1.1 The Prospect CORBA/CMIP Gateway

The evolving liberalisation of the European telecommunication markets results in a growing competition among service and network providers. This forces service and network providers to react more flexibly to changing user needs by creating and deploying new or improved telecommunication services rapidly and in a cost-efficient way. A key to the success of a service or network provider in this future environment is the comprehensive, flexible and integrated management of its telecommunication services. The challenge of deregulation and open network provision establishes the following requirements:

- In order to speed up the development of new, manageable telecommunication services, support for the development of generic management components as well as the re-use of already existing management components is needed.
- The integration of legacy systems as well as the step-wise migration to innovative technology must be supported.

The current view is that CORBA will be one of the main technological foundations on which future distributed Telecommunications management solutions will be built. TMN solutions based upon CMIS/CMIP do however exist today and will exist for many years to come. Prospect has therefore recognized a need to provide interworking between the CORBA and CMIP technology domains.

The requirements, which apply to CORBA/CMIP interworking are examined. The requirements are stated from two different perspectives; a technological perspective and the perspective of the Prospect consortium. The requirement stated from the Prospect perspective constitutes a subset of the requirements as seen from the more general technological perspective. It is only this subset of requirements which are covered by the architectural components defined by Prospect.

The architectural components defined by Prospect will satisfy the CORBA/CMIP interworking needs set by the Prospect trial. Prospect does however believe that the architectural components defined provide a component set useful for the most relevant CORBA/CMIP interworking scenario.

#### 2.1.1.1 Technological perspective

Transparency is of great importance from a technological viewpoint, in particular that no relevant information is lost bridging the technology domains. CORBA entities should be able to communicate with CMIP based entities and vice versa without being aware of the underlying technologies. Issues such as naming, event forwarding, scoping & filtering, multiple replies and managed object creation and deletion must be addressed. The CORBA/CMIP gateway functionality must ensure that TMN solutions built before or after the deployment of CORBA will integrate efficiently into a CORBA/TMN environment. The following constraints must therefore be put on the gateway functionality

- It must be transparent for a CMIP agent whether a CMIS request originated from a true CMIP entity or from a CORBA based manager via the gateway.
- It must be transparent for a CMIP agent whether it is issuing notifications to a true CMIP entity or to a CORBA based manager via gateway functionality.



- It must be transparent for a CMIP based manager whether it is issuing a CMIS request to a true CMIP agent or to a CORBA based agent via a gateway.
- It must be transparent for a CMIP based manager whether it is receiving notifications from a true CMIP agent or from a CORBA based agent via gateway functionality.

The transparency requirements must ensure that existing OSI/TMN solutions will be able to operate in a hybrid CORBA/CMIP environment.

From a CMIP based manager's perspective, these constraints indicate that the full set of CMIS features must be available to the manager even if it is connecting to a CORBA based agent via a gateway. Support of the CMIS features indicates that the gateway functionality must provide solutions for the following subjects:

- The CMIP based manager must be able to connect to CORBA based agents via the gateway using the AE-title of the agent. The CORBA based agent may be distributed on a number of CORBA servers. The gateway must present a unified view of such distributed servers to the CMIP manager.
- CMIP/CMIS containment relationships must be established between the CORBA objects of the possibly distributed CORBA based agent. The relationships between CORBA objects must allow for the support of scoped and filtered CMIS request.
- OSI/TMN names (DN, RDN) must be associated to the CORBA objects reflecting the established relationships. The gateway must be able to resolve the OSI names provided in CMIS requests and invoke operations on the corresponding CORBA objects.
- CMIS Create and Delete requests must generically be supported.
- It must be possible for the CMIP based manager to instantiate and configure EFD's in order to receive events from the CORBA based agent.

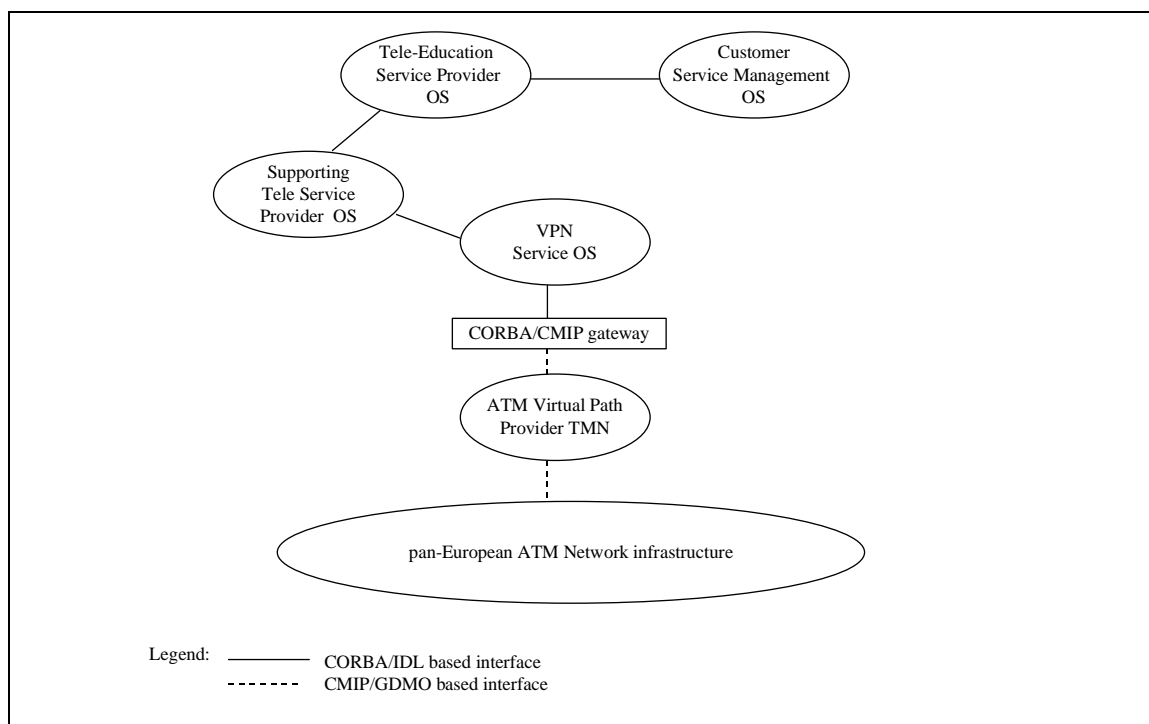
The gateway support for the above listed items should to as large an extent as possible be based on the COSS naming, Event and Life-cycle services.

Initial CORBA based managers will not be able to utilize the full set of CMIS features (e.g. scoped and filtered requests, linked replies) until an extended set of CORBA services are made available. Such services may either be conceived as part of the gateway functionality or as a part of CORBA based manager functionality. These services will have a broader application and are likely to be used also in pure CORBA environments. As seen from the OSI agent however, it is vital that the gateway provides support for the basic CMIS features including CMIS Create/Delete request and CMIS Event reports. In order to do so the gateway must resolve the following issues:

- It must be possible to associate AE-titles with CORBA based managers, in order to configure EFD's or for the EFD's to issue notifications to CORBA based managers. The gateway functionality must be able to receive emitted notifications and to deliver them to the appropriate CORBA manager (this feature should be implemented using the COSS Event Service)
- The gateway must provide generic services, such that a CORBA based manager is able to issue CMIS create and delete requests to the agent.

### 2.1.1.2 PROSPECT perspective

Prospect is implementing a trial network which can demonstrate and validate the management of co-operating and competing services in support of commercial/business end-users. This is illustrated in Figure2-1, which depicts a simplified version of the Prospect management architecture.



**Figure A-2: Prospect management architecture**

The Prospect trial will involve a number of service provider operation systems (OS), which co-operate in order to provide tele-educational services to customers. The tele-educational services provided to the end-users in the trial (i.e., a virtual classroom) require access to other tele-services, which are referred to as supporting tele-services. Typical supporting tele-services are multimedia mail, information retrieval, conferencing, etc. These are available to the tele-education service provider at more than one site. A number of value-added services (i.e., VPN, PCS) and ATM bearer services are required to support the operation of the higher-layer tele-services. The operation systems at the service level offer management services via CORBA based management interfaces.

One of the key components in this architecture is the Virtual Private Network Provider OS. The VPN provider will offer VPN Services to customers, who may be end-users or other tele services providers. Another key component in the Prospect trial architecture is the ATM Virtual Path Provider. The ATM Virtual Path service is realized with a CMIP/CMIS based TMN management system, offering Virtual Path services via a TMN X-interface. With this architecture there is consequently a need for a CORBA/CMIP gateway which allows the CORBA based VPN Service Provider OS to use the ATM Virtual Path service via the X-interface offered by the ATM Virtual Path Provider CMIP/CMIS based Management system. The Prospect trial exemplifies what is believed to be likely future scenarios, where new CORBA based TMN management systems will use management services offered via X-interfaces by TMN based network and network element level management systems.

Within the context of the Prospect trial the overall requirements which must be satisfied by the CORBA/CMIP gateway are summarized as follows:

- The CORBA/CMIP gateway must provide access to OSI managed objects within CMIP based agents from a CORBA based management application. The gateway must provide an IDL mapping to all of the basic CMIS operations, i.e. M-GET, M-SET, M-CREATE, M-DELETE and M-ACTION.
- In order to make use of the power and flexibility of CMIP based agents, the gateway must provide support for scoped and filtered operations, which map to the CMIP scoping and filtering capabilities.
- The gateway must allow CORBA based management applications to receive events emitted by CMIP based agents.
- The gateway must support location transparency and allow CORBA management applications to gain access to OSI managed objects based on their OSI distinguished name.

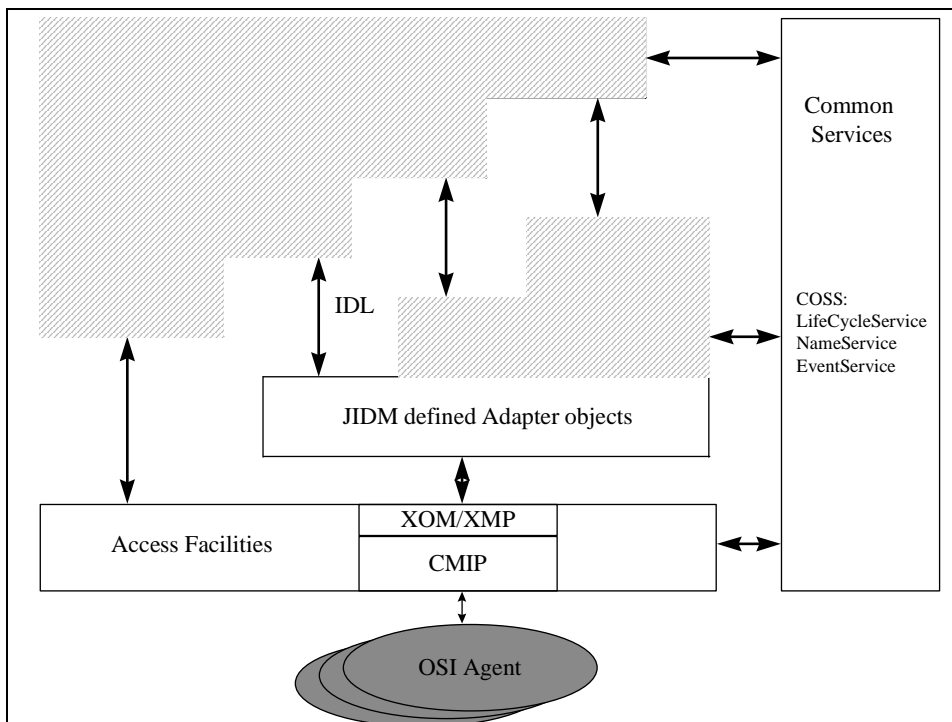
Prospect has chosen to narrow the scope of the concepts for CORBA/CMIP interworking to cover the requirements stated above, i.e. scenarios where CMIP based management application access and control CORBA based agents are not covered. A gateway satisfying the above stated requirements will meet the needs set by the Prospect trial, which itself exemplifies what is believed to be a frequent future TMN scenarios. The narrowing of scope is furthermore justified with the following statements:

- NMF has chosen to adopt CORBA as the technology basis for the Management System Framework. This indicates that future distributed telecommunications management solutions will be based on CORBA.
- TMN standardisation has been focused on the network element and network level. Today a number of standards exist for network element level management, network level standards are in progress. TMN standardisation on the service and business levels has made very little progress. Today's OSI based TMN solutions are to a large degree vertical solutions rather than horizontal. The bulk of today's CMIP based TMN solutions are mainly at Network Element level and to some degree at the Network level. Very few CMIP based solutions exist at the service and business levels and the use of TMN X interfaces to provide distributed horizontal solutions is rare. CORBA is accepted as a good technology for distributed solutions, and the use of CORBA to implement distributed telecommunication management solutions at the service and business level is increasing.
- CMIP is a good protocol in situations where large amounts of information need to be retrieved on a regular basis and where information needs to be retrieved selectively. CMIS/CMIP has proven itself to be a strong technology for the implementation of efficient network element agents. With the current set of services and facilities CORBA is less efficient in these areas. CMIP/CMIS therefore remains a good and viable implementation solution at the network element level, and deployment of CMIP/CMIS at this level seems to be increasing.

Given these statements Prospect believes that there is a strong need for a CORBA/CMIP gateway which will provide CORBA based distributed business and service level TMN management applications access to information maintained within CMIP based TMN agents at the network and in particular the network element level. The need for a gateway with the reverse polarity will be far more limited.

### 2.1.1.3 Architectural Overview

When relating the work of the JIDM group to the NMF Management System Framework it can be seen that the CMIP to CORBA mapping in the specification and interaction translation documents provides the interface definition of MSF Adapter objects for OSI managed objects. The specification translation document provides the basic method to derive the adapter object interfaces from the GDMO description of the information model in the agent(s) accessed through the gateway. The inter-action translation document deals with the dynamic aspects of CORBA/CMIP interworking. It adds new objects and additional method to the basic adapter interfaces, in order to deal with complex CMIP functionality such as scoping, filtering, event handling and linked replies.



**Figure A-3 : JIDM defined gateway architecture**

The JIDM defined adapter objects will use CMIP access facilities to access the underlying OSI agents. JIDM does not specify how the adapter objects should access the real OSI managed objects, but a likely candidate is through XOM/XMP API provided by the OSI access facilities.

The JIDM defined adapter objects depend on the CORBA LifeCycleService, NameService and EventService in order to provide the required service.

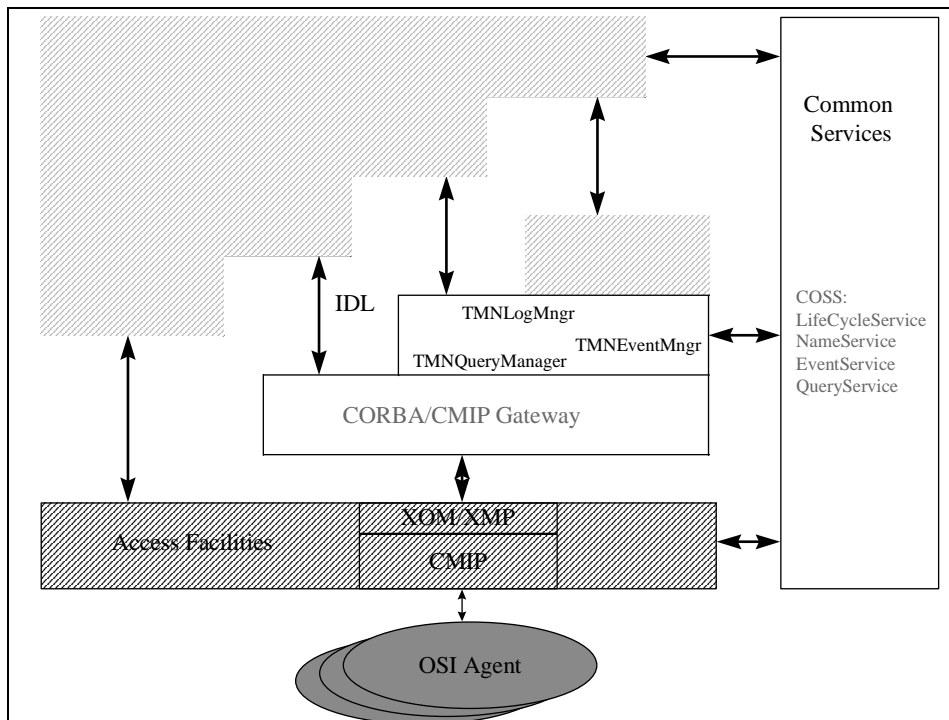
The JIDM defined adapter objects fall into the following categories:

- **GDMO derived adapter objects** each corresponding to a specified GDMO object class. The GDMO derived adapter objects provide the basic access to the OSI managed Object and contain methods to view and manipulate the attributes of the OSI managed object. In addition to the basic operation the GDMO derived adapter objects contain support for advanced CMIS functionality such as scoped/filtered request and the handling of linked replies.
- **Factory adapter objects**, which are not directly derived from the GDMO specification but reflect the CMIS create capabilities expressed through the GDMO Name Bindings,

used for the OSI agents. The factory objects are capable of creating instances of GDMO derived adapter objects. Such a creation maps onto a CMIS create request to an underlying OSI agent.

- **Event Port adapter objects** which provide the functionality of receiving events from OSI agents, translating and forward these events to CORBA based managers.
- **Proxy agent adapter objects** which represent connections to underlying OSI agents. The Proxy agents offer interfaces which allow a manager to obtain references to adapter objects representing managed objects in the OSI agent. The proxy agent interface furthermore provides methods to obtain references to factory objects, which in turn allow for the creation of new OSI managed objects.
- **Finder objects** which allow a user to obtain references to Proxy agent and EventPort objects based on a given set of criteria.

The Prospect CORBA/CMIP gateway architecture builds on the above described concepts but expands the JIDM defined architecture by proposing new MSF composite objects, which in turn make use of an extended set of common services.



**Figure A-4: Proposed Prospect extensions**

Three new composite objects are defined, which in line with the MSF architecture serve to simplify and ease the use of the service provided by the underlying JIDM defined adapter object. The proposed Prospect extension defines a TMNEventMngr interface, a TMNLogMngr interface, and a TMNQueryManager interface.

- TMNEventMngr objects provide a mediation of the event handling capabilities offered via the basic EventPort interfaces and take care of the complex sequence of operations involved with the initialisation of EventPorts and associated EventForwarding Discriminator adapter objects.
- The TMNLogMngr interface mediates the use of Event Logs in both OSI and CORBA domains OSI agents, and provides an easy-to-use CORBA interface to OSI event logging and log retrieval.

- The TMNQueryManager interface provides access to JIDM defined OSI adapter objects, and allows a service user to obtain references to a set of OSI adapter objects with given properties independent of their location. A query to the TMNQueryManager may return references to OSI adapter objects for managed objects located in different OSI agents. The TMNQueryManager is a specialisation of the OMG query service.

The proposed architectural extensions together hide the aspects and details specific to the management of OSI based agents, thereby removing the need for CORBA based management applications to have OSI TMN technology specific knowledge. The TMN Event manager and TMN Log manager composite objects furthermore represent generally useful telecommunication management component interfaces, which could be applied also in pure CORBA applications.

The gateway architecture does however also take into account applications where CORBA based managers possessing OSI management specific knowledge wish to use a lower level mapping to the powerful set of CMIS features in order to provide efficient management of the underlying OSI agents.

### 2.1.2 The VITAL-REFORM Coexistence and Migration Approach

The VITAL and REFORM projects validate and augment TINA specifications through the design and implementation of a fully TINA-compliant prototype. An important aspect of this validation has to do with the TINA Network Resource Architecture (NRA), which addresses network management aspects. The latter is relatively less well developed in comparison to the Service Architecture (SA), which addresses service control and management aspects. Validation of the NRA involves comparisons with TMN methodologies and architectural issues and has led to an approach for the re-use of TMN methodologies and specifications over a CORBA-based DPE. The latter is a means of both protecting relevant TMN investment but also in retaining the relevant advantages of TMN technology for network management, as explained below.

The key aspect of this approach is the introduction of “managed object clusters” administered by “management brokers” as architectural elements of a TINA system. These are similar in nature to TMN Operations Systems but operate in a native fashion over a TINA DPE in the sense there is no CMIP protocol and TMN Q3 interface. Managed objects have individual IDL interfaces which are derived from the relevant TMN GDMO specifications by using the XoJIDM guidelines. These can be accessed either directly, or indirectly through their Management Broker (MB). The latter offers functionality which is superset of that of an OSI/TMN agent, including scoping, filtering, multiple replies, asynchronous events based on slightly modified Event Forwarding Discriminators (EFDs) [X.734], event logging [X.735] and the rest of the OSI System Management Functions (SMFs) [X.73x] e.g. Metric Monitoring [X.739], Summarisation [X.738], etc.

The VITAL-REFORM approach is that of a “tight coupling” as presented in the possible coexistence strategies (see Figure 1). In this case, the advantages of distributed object technologies, i.e. easy programmability, multiple language bindings, portability and fine-grain distribution, are combined with those of the OSI/TMN approach, i.e. optimised access to management information, fine-grain event control and dissemination, scaleable global naming and a host of generic functionality through the OSI SMFs. A key aspect of this approach is the computational interface of the Management Broker which is CMIS-like in nature [Pavlou97b]. This allows TMN-based operation over the TINA DPE. A prototype based on this approach has been implemented in VITAL using parts of the OSIMIS TMN platform [Pavlou95] over a commercial CORBA implementation. The prototype has been used to realise the Resource Configuration Management (RCM) subsystem of the VITAL TINA system [Pavlou97a].

The rest of this section has the following structure. Issues regarding the use of “vanilla” CORBA for network management are presented to illustrate the relevant limitations and scalability problems. The Management Broker approach is then presented in some detail and the section closes with a summary and conclusions.

### **2.1.2.1 Limitations in Using Plain CORBA for Network Management**

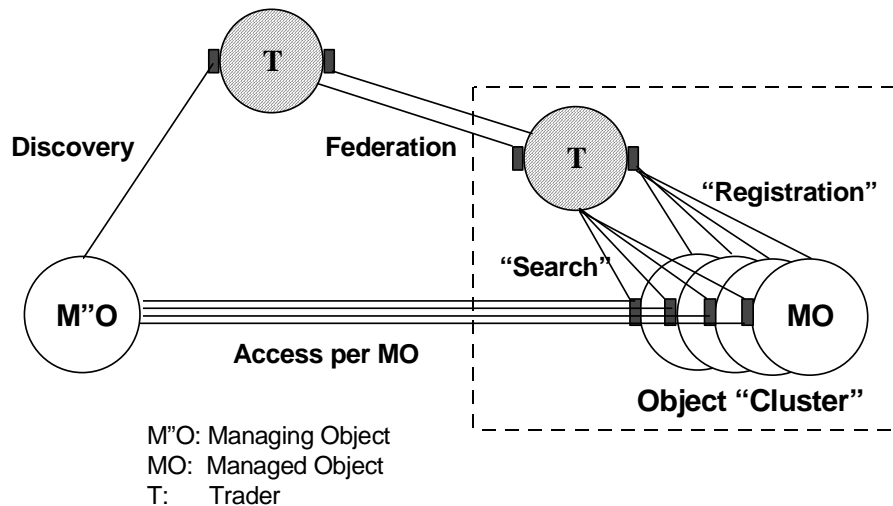
We examine here how CORBA could be used for network management, contrasting its approach to the protocol-based OSI System Management (OSI-SM) approach. But let’s summarise first the operational paradigm of the latter. Managed elements or management applications that assume an agent role provide management interfaces. A management interface consists of the formal specification of management information and of an access service/protocol that is mapped onto a well defined protocol stack. While the management information specification provides the MIB *schema*, object discovery and multiple object access facilities allow applications in manager roles to discover dynamically existing object instances. Operations to objects are always addressed through the supporting agent, which provides query facilities in a database-like fashion. In addition, the agent discriminates emitted notifications according to criteria preset by managers. Applications may discover each other through the directory.

If CORBA is used as the underlying access and distribution mechanism, managed objects could be mapped onto CORBA objects, accessed by client objects in managing roles. The key difference is that clusters of managed objects logically bound together, e.g. objects representing various aspects of a managed network element, are not seen collectively through an agent. As such, an important issue is to provide object discovery and selection facilities similar to OSI scoping and filtering. Such facilities are very important in management environments where many instances of the same object type typically exist, with names not known in advance e.g. call objects. Facilities similar to scoping are not currently provided in CORBA but it should be possible to extend name servers to provide similar functionality since they maintain the logical name space. Facilities similar to OSI filtering could be provided by traders but are not as powerful.

The problem with the use of CORBA is that federation is a key aspect in order to achieve scaleable systems. In essence, it will be necessary to have dedicated name servers, traders and event/notification servers for every logical cluster of managed objects, e.g. in every managed element, in order to reduce traffic and increase real-time response. Those “low-level” servers will be unified by “higher-level” servers in a hierarchical fashion but federation issues have not yet been worked out and are not simple. In addition, even with such facilities in place, the generated management traffic will be at least twice that of OSI management. With CORBA, matching object references will be returned to the client object and the operations will be performed on an object-by-object basis. In OSI management, the multiple object access request will be sent in one packet while the results will be returned in linked replies, one for each object accessed. The use of CORBA for network management, using federated name servers and traders to provide facilities similar to OSI/TMN agents, is depicted in Figure A-5.

In the current state of CORBA, generated traffic will be much bigger since managed objects will have to contact servers across the network. We will use an example to demonstrate the amount of traffic generated by both OSI SM and CORBA for network management. Let’s assume a network element such as an ATM switch which contains  $N$  managed objects representing established Virtual Channel Connections (VCCs). Out of those,  $M$  managed objects originate from a particular source address ( $N > M$ ). If an OSI-SM manager wants to find those VCCs that originate from that address, it will have to locate the element agent through the directory and

send a request with scope and filter constraints. The overall number of application level packets, ignoring connection establishment, will be: 2 for contacting the directory and 2+M for retrieving the objects [M+4 in total]. The same amount of traffic with CORBA will be: 2\*N for the VCC objects to contact the trader, 2 for the client to contact the trader and 2\*M for the client to retrieve the results [2\*(N+M+1) in total]. In addition, if the specified filter asserts on a dynamically changing attribute instead of a static property, the trader will have to generate another 2\*N packets searching through the VCCs.



**Figure A-5: Using CORBA for Network Management**

An additional issue is the complexity of the overall resulting framework as CORBA dictates conformance to internal software interfaces which leaves less space for optimized implementations. The feasibility of network elements with 10's of thousands CORBA managed objects needs to be investigated.

### 2.1.2.2 The VITAL Management Broker Approach

b

As explained in the previous section, the use of the ODP / OMG CORBA model for network and service management presents a number of difficulties to overcome due to the dynamic nature of management information and the number of managed objects typically present in managed elements. On the other hand, the OSI-SM model scales much better and has already been used for managing large telecommunications infrastructures e.g. SDH/SONET, ATM, etc. Based on this observation, an ideal framework would combine the expressive power of OSI-SM and the programmability, portability and distribution aspects of ODP / OMG CORBA. In order to specify such a framework it is important to be able to map management information specifications in GDMO to computational interfaces in IDL. The information modeling aspects of the two frameworks are similar and the X/Open Joint Inter-Domain Management task force (XoJIDM) has defined rules for this mapping [JIDMs]; these are summarised next.

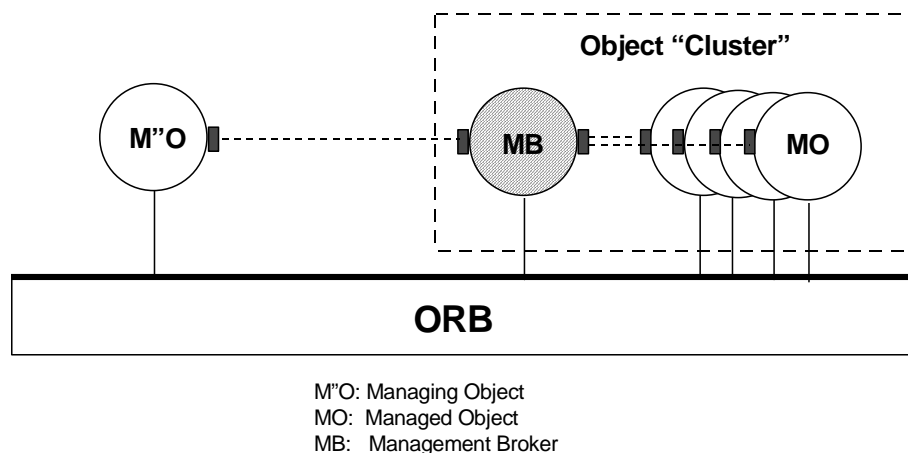
Mapping GDMO specifications to CORBA IDL is not straightforward because GDMO as information specification language and CMIS/P as the access method have a number of aspects for which there exist no IDL and CORBA equivalents. These include the late binding of functionality to managed object instances through the use of conditional packages; the existence of notifications as part of managed object specifications; the fine grain event support; and the use of scoping and filtering as "query language" facilities that may result in multiple replies. It should also be noted that GDMO attributes cannot be mapped directly onto IDL attributes since user exceptions with specific error information may be raised as a result of access to them. In IDL it is not possible to associate user exceptions with attribute access.



Despite these differences, it is still possible to use workarounds in order to achieve a generic mapping. GDMO attributes may be mapped onto access methods specific to the attribute in hand, according to its property information e.g. `administrativeState_get`, `administrativeState_set` etc. GDMO actions may be naturally mapped onto IDL methods. Notifications may be mapped onto interfaces in the opposite direction, corresponding to the push and pull models. Finally, conditional packages can be made “mandatory” by being added to the resulting IDL interface. Their presence though becomes an implementation issue: the standard CORBA *not\_implemented* exception should be raised whenever a method of a non-implemented package is invoked. Translated IDL interfaces follow exactly the same inheritance lattice as the original GDMO classes, while the *Top* class inherits from a *ManagedObject* base interface which in turn inherits from CORBA’s *Object*, as do all IDL interfaces.

The suggested mapping goes a long way towards reconciling the differences of the two object models but some semantics are inevitably lost in the translation. Most notably, in GDMO conditional packages may be included in an object instance at creation time. This facility allows for the late binding of functionality to that instance and it may also be used to configure its “mode” of operation. This cannot be achieved through the suggested translation. Furthermore, some conditional packages for the same class may be mutually exclusive; this again cannot be modeled in IDL. If ISO and ITU-T are to adopt the proposed translation guidelines by XoJIDM, they should also instruct ITU-T GDMO information modeling working groups to avoid the use of conditional packages in a non IDL-compatible fashion.

A more important difference concerning the translation has to do with the access methods. The operational model of CORBA is that of a single distributed object, accessed in a location transparent fashion. In OSI Management, managed objects can be accessed collectively through the CMIS/P scoping and filtering facilities. These may be used for discovery services e.g. “which calls are currently established through that element” and they minimize the management traffic incurred on the managed network. In addition, the same operation may be performed on many managed objects and this is not only an engineering-level optimization but allows as well a higher level of abstraction to be provided to managing functions. Discovery facilities may be provided through naming servers and traders in CORBA but the efficiency of such mechanisms, with potentially thousands of transient managed objects in network elements, needs to be evaluated. In addition, the CMIS/P operational paradigm with potentially multiple operations expressed through a single request is lost, unless similar facilities are provided over CORBA.



**Figure A-6: The VITAL Management Broker Approach**

This is exactly the VITAL approach i.e. to provide OSI/TMN-like facilities over CORBA. In this, the operational framework of OSI management is retained over CORBA through

Management Brokers (MBs). A logically bound cluster of managed objects similar to an OSI/TMN agent application is administered by a management broker; the latter provides multiple object access facilities similar to CMIS. Of course, managed objects may be also accessed directly in the standard CORBA fashion. Event management is provided by event discriminators and logs through filtering, in order to overcome the relevant CORBA limitations. Finally, the rest of the OSI Systems Management Functions [X.73x] are maintained as generic CORBA objects that may be instantiated within a cluster. This approach essentially maintains the OSI operational model over CORBA but replaces the access (i.e. CMIS/P) and distribution (OSI directory) mechanisms. As such, it retains the OSI management expressive power, event model and generic management facilities while it benefits from the distribution, portability and easy programmability of CORBA. The approach is depicted in Figure A-6.

A Management Broker act as object factory, name server, trader and notification server using the relevant OSI-SM methods and techniques i.e. CMIS-like object creation, naming through distinguished names, scoping / filtering and event reporting / logging through event discriminator and log objects. As an example, the following is a fragment of the IDL specification of one of the Management Broker interfaces. The example demonstrates two operations: the first, `objectSelection`, allows a client to identify information objects according to certain scope and filter parameters; the second, `multipleObjectGet`, allows a client to read selected attributes of a group of objects in a single operation.

```
// hierarchical naming in X.700 style
typedef struct RelativeName_t {
    AttributeId_t attrId;
    string        attrValue;
};
typedef sequence<RelativeName_t> DistinguishedName_t;
typedef DistinguishedName_t ObjectInstance_t;

// object selection through scoping and filtering
enum ScopeChoice {
    baseObjectChoice,
    firstLevelOnlyChoice,
    wholeSubtreeChoice,
    individualLevelChoice,
    baseToNthLevelChoice
};
typedef struct Scope_t {
    ScopeChoice choice;
    unsigned long level;
};

// Filter_t is a translation of the X.711 CMISFilter in IDL
typedef struct ObjectSelection_t {
    Scope_t scope;
    Filter_t filter;
};

interface MultipleOpManagementBroker : ManagementBroker {

    void objectSelection (
        in ObjectInstance_t baseObjectInstance,
        in ObjectSelection_t objectSelection,
        out ObjectInstanceList_t objectInstanceList
    ) raises (OBJECT_SELECTION_ERRORS);

    void multipleObjectGet (
        in ObjectInstance_t baseObjectInstance,
        in ObjectSelection_t objectSelection,
        in AttributeIdList_t attributeIdList, // optional (for all
attrs)
        out GetResultList_t resultList
    ) raises (MULTIPLE_OBJECT_OP_ERRORS);
    // ...
}

```

A Management Broker does not only have a CMIS-like interface but also interfaces to be “told” about new objects it needs to administer and to receive notifications from the managed objects it

administers. In addition, every MO inherits from a *ManagedObject* interface through which the MB notifies it that it administers it. A MB and the objects it administers may be physically distributed but they logically form a “cluster” which can be collectively accessed. A managed object may belong to more than one MB domains. In the case of managed network elements, at least one MB needs to be physically located together with the local managed objects so that it provides optimized access and event dissemination facilities with minimal management traffic. In general, it is not necessary to provide separate IDL interfaces for every managed object as this may not be technologically feasible with the current state of CORBA implementations. In this case, the managed objects and the broker may interact through a local mechanism e.g. internal C++/Smalltalk interfaces if they share a common address space (an engineering “capsule” in ODP terms).

A first version of such a management broker has been designed and implemented in VITAL using the OSIMIS TMN platform [Pavlou95] and a commercial implementation of CORBA. We have used this to provide clusters of objects that represent Network Topology Configuration Maps (NTCMs) which model ATM Virtual Path and Virtual Channel layer networks. In those applications, objects are “lightweight” representations of element level managed objects and represent network-wide topological information through containment and other relationships. Those “maps” can be flexibly navigated through scoping and filtering while topology changes may be effected through object creation and deletion. The prototype implementation served to validate and demonstrate the architectural concepts presented.

### **2.1.2.3 Summary**

The above described approach for TMN and TINA coexistence and migration retains the operational model of OSI-SM/TMN over a CORBA-based Distributed Processing Environment. Management Brokers mirror the facilities of OSI-SM but use CORBA as the access and distribution mechanism. The approach is based on the results of the XoJIDM work for mapping GDMO specifications to IDL but it proposes a native CORBA-based Open Distributed Management Architecture (ODMA).

This approach retains the relevant advantages of TMN for network management while it is both compliant and complementary to the JIDM approach. In addition, this is a viable path for gradually migrating existing TMN systems over CORBA-based DPEs. An additional aspect of the presented approach is that it can also be used to provide generic adaptation functions to existing OSI-SM/TMN systems. In fact, VITAL has implemented both native CORBA-based MBs but also generic MB adapters to existing OSI/TMN systems. As such, this approach provides a smooth migration path from current TMN-based management systems to target systems operating over CORBA-based DPEs. This will make possible a single integrated “service engineering” framework that will encompass both network management and service control and management aspects.

## **2.2 TINA**

### **2.2.1 Distributed Management Facilities**

It is descriptive and builds on work done in XoJIDM and the NMF CMIP-API. It deals with CMISE-like services in TINA, including notifications, the inheritance hierarchy and naming principles.

The proposed solutions are specified with IDL interfaces : Base Manager, Managed Object. These services are part of the Common Facilities as defined by OMG, and named “*Distributed Management Facilities*” (DMF).

The material in this section is based on a TINA-C document aiming at providing a framework for Distributed Management in the TINA architecture. In summary, it provides a specification for Managed Object interaction which makes use of a particular component: the Management Broker Facility.

### 2.2.1.1 Overview

The TINA-C document pursues a clear strategy: to propose TMN-like/CMIP-like services within TINA. This led to a definition of an interaction scheme similar to the TMN paradigm. For each TMN-like function, a particular Facility is specified which emulates the function, while naturally staying within the distributed organization. Thus the functionalities related to transparent distribution are taken in charge by the Management Broker; multiple operations are handled by a special Object Selection Service dedicated to this task; the notification flow which is proposed is based on a Notification Facility. It has to be noted that while the document admits that the OMG Naming Service is insufficient, no solution is provided for MOI naming.

Functionally, the Management Facilities specified by the document are situated between applications and DPE services (Common Object Services):

The Distributed Management Facilities are composed of:

- Basic Distributed Management Interaction Facilities, which provide CMIP-like interaction between Managing and Managed Objects on one hand, and Notification functions on the other.
- Systems Management Facilities, like in TMN (10165-xx series), which provide the basics for FCAPS: log, alarm, event reporting etc.

#### DMF and TMN interoperability

Interoperability is achieved by gateways inside the system, between TINA Manager Components and OSI Agents or TINA Managed component and OSI Manager or between TINA management systems and OSI ones.

The approach to build an Operations System proposed by the TINA document is typically in an inheritance type of relation, to TMN. In particular, it states to specify information models in GDMO, and to translate them to IDL.

#### Interaction model

The fundamental structure that is proposed is the Manager/Managed Object interaction. Each Managed Object possesses an IDL interface. The IDL class organization mimics the GDMO layout.

### 2.2.2 Services

Initial requirements for them are found in the TINA-C document.

- Object Selection Services
- Naming Services  
Each MOC is uniquely identified by the interface identifier in the scope of a module  
Every MOI is uniquely identified in a naming context, or globally.  
The definition of that service, as the OMG Naming service may not support OSI Naming, is not part of that document.
- Event Services
- Lifecycle Services
- Relationship Services
- Trading Services

### 2.2.3 Naming Principles

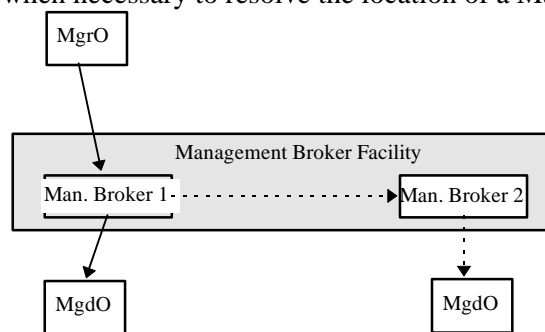
Unfortunately, this important topic is not studied within the document. The only assertion is that the OMG Naming Service should not be used to support an OSI type of naming scheme.

However, the document identifies some requirements on Naming:

- a Naming Service must be available;
- the same Naming Service should support the management architecture as well as the other parts of the TINA system;
- MOCs each have an interface and can be identified uniquely by their Interface Id;
- MOIs can be identified uniquely.

### 2.2.4 Distributed Management Broker Facility

The MBF takes in charge the management operations forming the CMIP-like interaction between managing and managed objects. The MBF is distributed in the sense that a system may carry more than one Management Broker server. The MB servers cooperate, by forwarding requests one to another, when necessary to resolve the location of a Managed Object.



**Figure A-7: Distributed Management Broker**

## 2.3 XoJIDM

The JIDM Task Group, from X/Open and NM Forum, aims at defining an interoperating architecture between OSI and CORBA systems. One of the main outcomes of this project will be an algorithm for translating managed object definitions written in GDMO (as used in OSI, etc.) into IDL (as used in OMG specifications), as a means of enabling the simpler interworking of management systems based on OSI and OMG technology. This is also expected to be particularly important in enabling better integration between systems and network management. The specifications will facilitate the manipulation of conventionally defined managed objects by management systems based on emerging OMG technology, and will form an important bridge between the Distributed Systems Management and Object Technology programs. The work is focusing on the definition of a gateway between CORBA and OSI management.

XoJIDM produces two documents :

- 1- The Specification Translation [JIDMs] relates to the static/compile-time environment. It defines a mapping of ASN.1 to IDL types, using exclusively 'typedef'. Complex constant values cannot be represented in IDL, they are defined as operations returning the constant value. The generated IDL types ignore some ASN.1 specifications : defaults values, tagged types, constrained types and subtypes appear only in IDL comments. Those features do not need to appear in the IDL, which is concerned only by the way the data is coded and transferred, not the constraints applied on their use. The same document presents a translation from GDMO to IDL. A managed object class maps to an IDL interface with the same name, plus two additional interfaces which support multiple replies and notifications. This valuable work will be used in the migration architecture, for it carries out a mandatory step for the migration from OSI to CORBA TMN.

- 2- The Interaction Translation [JIDMi] relates to the dynamic / run-time environment. It describes how the CMISE services can be performed by CORBA entities. The first draft has not been released yet but should be available by the middle of 1997.

## **2.4 EURESCOM**

Results from EURESCOM P508 [PIR8.2] are a valuable input. In summary:

1. The architecture considers a variety of migration paths from TMN based systems integrating a TINA based island to DPE based systems making use of TMN islands.
2. The architecture of the migrating components is specified to a large extent considering interfaces to: UNI signaling, NML information model and EML information model. These components integrate in particular various dynamic type of mappings between the above listed elements.

More details will be presented in the second part of this guideline.

## **3. Abbreviations**

ASN.1	Abstract Syntax Notation 1
ATM	Asynchronous Transfer Mode
BML	Business Management Layer
CA	Customer Access
CMIP	Common Management Information Protocol
CMIS	Common Management Information Services
CMISE	Common Management Information Service Element
CORBA	Common Object Request Broker Architecture
CoS	Class of Service
COSS	Common Object Services Specification
DPE	Distributed Processing Environment
EFD	Event Forwarding Discriminator
EML	Element Management Layer
ETSI	European Telecommunications Standards Institute
FCAPS	Fault, Configuration, Alarm, Performance and Security
FDN	Full Distinguished Name
GDMO	Guidelines for the Definition of Managed Objects
GMS	Generic Managed System
IDL	Interface Definition Language
ISO	International Standards Organisation
ITU	International Telecommunications Union
KTN	Kernel Transport Network
MB	Management Broker
MBF	Management Broker Functionality
MF	Management Function
MIB	Management Information Base
MOC	Managed Object Class
MOI	Managed Object Instance
MSP	Management Service Provider
NE	Network Element
NEF	Network Element Function
NML	Network Management Layer
NMS	Network Management System
NNI	Network-Network Interface
NRIM	Network Resource Information Model
OAM	Operations And Maintenance

OAMC	Operation Administration Maintenance Center
ODMA	Open Distributed Management Architecture
ODP	Object Distributed Processing
OMG	Object Management Group
ORB	Object Request Broker
OSF	Operations Systems Function
OSI	Open Systems Interconnection
OSI-SM	OSI System Management
PNNI	Private Network - Node Interface
PNO	Public Network Operator
PRM	Protocol Reference Model
QoS	Quality of Service
RDN	Relative Distinguished Name
RIP	Routing Information Protocol
RP	Reference Point
SDH	Synchronous Digital Hierarchy
SMF	Systems Management Functions
SNMP	Simple Network Management Protocol
TINA	Telecom Information Networking Architecture
TINAC	Telecommunication Information Networking Architecture Consortium
TMN	Telecommunications Management Network
UNI	User Network Interface
XoJIDM	X/Open Joint Inter Domain Management task group

#### 4. References

- [CORBA] *Common Object Request Broker : Architecture and Specification r2.0*, OMG, 1995
- [ETSI NA5] ETSI, DTS/NA52212.1, "Configuration management for the X-type interface between Operation Systems of a VP-VC Cross Connected Network", June 1996
- [I.311] ITU-T Recommendation I.311 (1993) "*B-ISDN general network aspects*"
- [I.321] ITU-T Recommendation I.321 (1991) "*B-ISDN protocol reference model and its application*"
- [I.610] ITU-T Recommendation I.610 (1995) "*B-ISDN operation and maintenance principles and functions*"
- [I150] ITU-T Recommendation I.150 "*B-ISDN asynchronous transfer mode functional characteristics*"
- [I321] ITU-T Recommendation I.321 "*B-ISDN protocol reference model and its application*"
- [JIDMi] *Inter-Domain Management: Interaction Translation*, X/Open Preliminary Specification planned for 1996.
- [JIDMs] *Inter-Domain Management: Specification Translation*, X/Open CAE Specification status planned for Q1 1996.
- [LaBarre93a] L.LaBarre (Editor), Forum 026 - Translation of Internet MIBs to ISO/CCITT GDMO MIBs, Issue 1.0, October 1993.
- [LaBarre93b] L.LaBarre (Editor), Forum 029 - Translation of Internet MIB-II (RFC1213) to ISO/CCITT GDMO MIB, Issue 1.0, October 1993.
- [M.20] ITU-T Recommendation M.20 (1992) "*Maintenance philosophy for telecommunications networks*"
- [M.2100] ITU-T Recommendation M.2100 (1992) "*Performance limits for bringing into service and maintenance of international digital paths sections and transmission systems*"
- [M3010] *Principles For A Telecommunications Management Network*, ITU-T Recommendation M.3010, 1993

- [M3020] ITU-T Recommendation M.3020 "TMN interface specification methodology"
- [M3100] ITU-T Recommendation M.3100 "Generic Network Information Model"
- [M3400] ITU-T Recommendation M.3400 "TMN Management Functions"
- [M4 req] ATM Forum Network Management Group (1994) "*M4 Interface Requirements and Logical MIB: ATM Network Element View*"
- [M4 spec] ATM Forum. Network Management Group (1995) "*CMIP specification for the M4 interface*"
- [Mazumdar] *Mapping of Common Management Information Services to OMG Common Object Services Specifications*. Subrata Mazumdar, AT&T Bell Lab CSRL, 1996
- [McCarthy95] K.McCarthy, G.Pavlou, S.Bhatti, J.Neuman De Souza, "Exploiting the Power of OSI Management for the Control of SNMP-capable Resources Using Generic Application Level Gateways," ISINM'95.
- [ODMA] *Open Distributed Management Architecture*, ISO/IEC JTC1/SC21 N9947
- [ODP] *X.903: Basic Reference Model of Open Distributed Processing - Part 3: Prescriptive Model*, ISO/IEC JTC1/SC21, 1995
- [OMG95] Object Management Group, The Common Object Request Broker: Architecture and Specification, Release 2.0, July 1995.
- [Pavlou95] G.Pavlou, G.Knight, K.McCarthy, S.Bhatti, *The OSIMIS Platform: Making OSI Management Simple*, Integrated Network Management IV, ed. A.Sethi, Y.Raynaud, F.Faure-Vincent, pp. 480-493, Chapman & Hall, 1995
- [Pavlou97a] G.Pavlou, D.Griffin, *Realising TMN-like Management Services in TINA*, to appear in the Journal of Network and System Management, Special Issue on TINA, 4<sup>th</sup> Quarter 1997, Plenum Publishing
- [Pavlou97b] G.Pavlou, *From Protocol-based to Distributed Object Based Management Architectures*, to appear in the proceedings of the IFIP/IEEE Workshop on Distributed Systems: Operations and Management, Sydney, October 1997
- [PIR 8.2] PIR 8.2 - Migration paths towards TINA (refinement), ref. P508.BT.PL.006, Editor: Peter Loosemore. This is EURESCOM confidential information to EURESCOM shareholders.
- [SNMP] J.Case, M.Fedor, M.Schoffstall, J.Davin, A Simple Network Management Protocol (SNMP), Network Working Group, Request For Comments 1157, May 1990.
- [TINA] *Overall Concepts and Principles of TINA*, TINA-C Deliverable, 1995
- [TINAS] *TINA DPE Services Specification*, TINA-C, 1994
- [UNI] ATM Forum, "ATM User Network Interface version 3.1"
- [X.500] ITU-T X.500, Information Processing - Open Systems Interconnection - The Directory: Overview of Concepts, Models and Service, 1988.
- [X.722] ITU-T X.722, Information Technology - Open Systems Interconnection - Structure of Management Information - Part 4: Guidelines for the Definition of Managed Objects, August 1991.
- [X.730] ITU-T X.730, Information Technology - Open Systems Interconnection - Systems Management: Object Management Function, January 1992.
- [X.733] ITU-T X.733, Information Technology - Open Systems Interconnection - Systems Management: Alarm Reporting Function, February 1992.
- [X.734] ITU-T X.734, Information Technology - Open Systems Interconnection - Systems Management: Event Management Function, February 1992.
- [X.735] ITU-T X.735, Information Technology - Open Systems Interconnection - Systems Management: Log Control Function, September 1992.
- [X.738] ITU-T X.738, Information Technology - Open Systems Interconnection - Systems Management: Metric Objects and Attributes, 1994.
- [X.739] ITU-T X.739, Information Technology - Open Systems Interconnection - Systems Management: Summarisation Function, 1994.
- [X.741] ITU-T X.741, Information Technology - Open Systems Interconnection - Systems Management: Objects and attributes for access control, 1995.



- [X209] *Open systems interconnection - Model and notation - Specification of Basic Encoding Rules for ASN.1*, ITU-T Recommendation X.209, 1993
- [X700] *Management framework for open systems interconnection (OSI) for CCITT applications*, ITU-T Recommendation X.700, 1992
- [X701] *Open systems interconnection - System Management Overview*, ITU-T Recommendation X.701, 1992
- [X710] *Common Management Information Service Definition for CCITT Applications*, ITU-T Recommendation X.710, 1991
- [X711] *Common Management Information Protocol Specification for CCITT Applications*, ITU-T Recommendation X.711, 1991
- [X720] *Open systems interconnection - Structure of Management Information : Management Information Model*, ITU-T Recommendation X.720, 1992
- [X750] *Management Knowledge Management Function*, ITU-T Recommendation X.750, ISO/IEC 10164, 1993
- [X901] *Reference Model of Open Distributed Processing*, ISO/IEC JTC1/SC21/WG7, ITU-T X.901 | ISO/IEC 10746-1, 1995
- [XOPEN] *X/Open, OSI-Abstract-Data Manipulation and Management Protocols Specification*, January 1992.