INTEGRATED COMMUNICATIONS MANAGEMENT OF BROADBAND NETWORKS

Crete University Press, Heraklio, Greece
ISBN 960 524 006 8

Edited by David Griffin

First published in 1996 by
CRETE UNIVERSITY PRESS
Foundation for Research and Technology
P.O. Box 1527, Heraklio, Crete, Greece 711 10
Tel: +30 81 394235, Fax: +30 81 394236
email: pek@iesl.forth.gr

The ICM consortium consists of the following companies:

Alcatel ISR, France
Alpha SAI, Greece
Ascom Monetel, France
Ascom Tech, Switzerland
Centro de Estudos de Telecommunicações, Portugal
Cray Communications Ltd., United Kingdom (Prime contractor)
Danish Electronics, Light & Acoustics, Denmark
De Nouvelles Architectures pour les Communications, France
Foundation for Research and Technology - Hellas, Institute of Computer Science, Greece
GN Nettest AS, Denmark
National Technical University of Athens, Greece
Nokia Corporation, Finland
Queen Mary and Westfield College, United Kingdom
Unipro Ltd., United Kingdom
University College London, United Kingdom
University of Durham, United Kingdom
VTT - Technical Research Centre of Finland

Chapter 4

TMN architecture issues

Editor: George Pavlou
Authors: George Pavlou, David Griffin, Eric Scharf

This chapter discusses issues related to the TMN architecture. The ICM project's work exercised TMN architectural concepts in depth while stretching some of their aspects to the limit. As a result, a clear understanding has been gained on the exact meaning and real-world applicability of those concepts. In addition, extensions and modifications to the TMN architecture have been necessary, most of which have been, implicitly or explicitly, fed-back to the ITU-T Study Group 4. It is these clarifications, modifications and extensions to the architecture that constitute the main focus of this chapter.

An introduction to the TMN architecture is followed by a discussion on the TMN logical layering aspects; the use of the directory to support dynamic discovery and shared management knowledge facilities; a discussion on the nature of adaptation/ mediation and the classification of different types of mediation functions; a discussion on workstation functions and the f reference point; and a presentation of the relationship between functional blocks and constituting functional components with the example decomposition of the operation systems function. The revised TMN architecture is finally presented while the chapter closes with a discussion on various related issues.

4.1 Introduction and background

4.1.1 What is a TMN?

The purpose of a *Telecommunications Management Network* (TMN) [4.1] [4.2] is to support administrations in managing telecommunications networks and services. This means support for planning, provisioning, installing, maintaining and administering these. The TMN is a network which is logically separate from the telecommunications network being managed, but interfaces to it at several distinct points in order to send information to it and receive information from it, with the purpose of managing and controlling its operations. A TMN may use, for its communication requirements, parts of the telecommunications network itself. There may be multiple TMNs within one administration while it is also possible that a single TMN spans across multiple administrations.

4.1.2 TMN functions

TMN functions cover a number of functional areas. OSI Systems Management suggests five functional areas, Fault, Configuration, Accounting, Performance and Security management (FCAPS) [4.3]. Generic aspects of these resulted in the specification of the OSI Systems Management Functions (SMFs) that are part of the X.700 series of recommendations. Early work in RACE-I extended these to the nine areas presented below, which are described in the H400 series of RACE Common Functional Specifications [4.4]:

- design,
- planning,
- installation,
- provisioning,
- maintenance and faults,
- performance,
- security,
- accounting, and,
- customer query and control.

Four of the OSI functional areas are explicitly included; the fifth, Configuration Management, spans across areas such as performance, provisioning, accounting, maintenance and fault management. Test management which is also not explicitly included, spans across maintenance and fault, performance and installation management.

Design Management covers all the activities that are related with the design, engineering and documentation of the management capabilities needed by network elements, customer premises equipment, terminal equipment and service providers equipment in order to allow the provision of TMN management services for IBC.

Planning Management involves the selection, introduction and update of both telecommunications customer services and the management of telecommunication physical resources. Planning also organises, directs and controls the introduction of new customers services in a cost effective manner while recognising the requirements of the customer and operator. Planning is carried out in accordance with government and regulatory conditions and with the recommendations of relevant standards bodies. The principle sub-functions of planning are service forecasting, network modelling, network optimisation, network simulation, interface and protocol planning.

Installation Management ensures that network elements, supporting elements and their components are positioned and physically and electrically interconnected according to agreed engineering practice. New customer requirements may cause the Planning function to produce a schedule of work to be performed by Installation Management, so resulting in the enhancement of an existing network.

Provisioning Management involves sequences of operations for the provisioning, modification and deletion of telecommunication services for a customer by the service provider.

Maintenance and Fault Management is responsible for maintaining the IBC system in its nominal state of operation by carrying out corrective and maintenance. Maintenance and Fault Management responds to alarms by and diagnosing network faults and eliminating them by replacing, repairing or re-configuring the relevant network elements.

Performance Management regulates the IBC system to achieve the levels of performance agreed with the customers. As such it will try to optimise the use of resources on a network-wide basis and provide resources and related facilities to users which include end-users, value added service providers and bearer service providers.

Security Management is responsible for the protection of network resources and the integrity and privacy of information that is exchanged between various TMN functional blocks. Security services include Authentication, Access Control, Integrity, Confidentiality and Non-repudiation.

Accounting Management provides a set of activities for measuring the activation and use of the IBC services. Service and resource records are gathered from the IBC system and are processed to generate customer invoices (billing); this may involve the exchange of information with Accounting Management systems in other TMN domains. Information is also provided for planning tariffs and for assessing how well business objectives have been achieved.

Customer Query and Control allows a customer of telecommunications services to query TMN information and to carry out limited modifications to the network and to the services provided to him. Customer Query and Control is also a service provided by a TMN to other administrative domains.

4.1.3 TMN architecture

4.1.3.1 TMN architectural decomposition

A TMN consists of the following function blocks:

- Operation Systems Functions (OSF),
- Work Station Functions (WSF),
- Mediation Functions (MF),
- Q Adapter Functions (QAF),
- Network Element Functions (NEF).

The relationship between the functional blocks is shown in Figure 4.1. They are interconnected by a number of standardised reference points: q, x, and f. The g and m reference points are not a subject for standardisation as they are outside the TMN. All these reference points are considered later in this section.

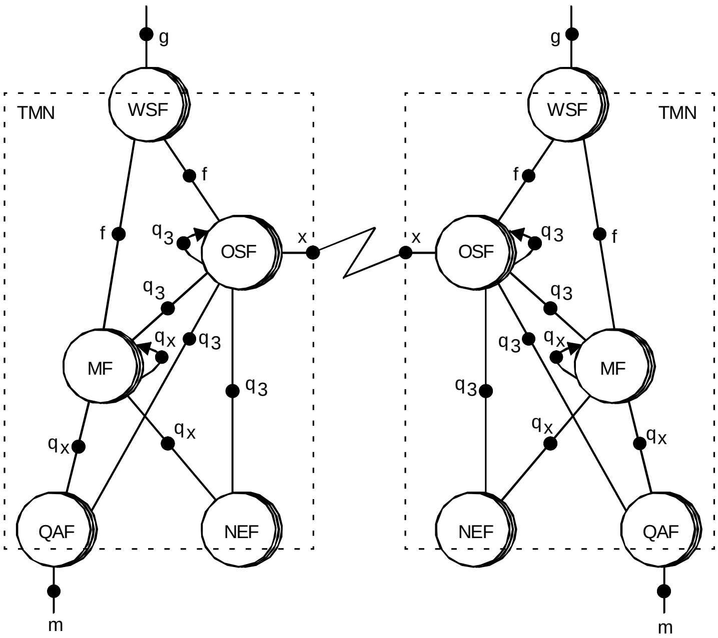

Figure 4.1 TMN function blocks and reference points

A *Network Element Function* (NEF) comprises telecommunications and support equipment (or groups or parts thereof). The NEF communicates with the TMN for the purpose of being monitored and controlled. The NEF includes only those equipment aspects that are subject to management.

The *Q Adaptor Function* (QAF) is used to connect to the TMN those NEFs which do not support standardised TMN reference points. The responsibility of a QAF is to translate between a TMN and a non-TMN (e.g. proprietary) reference point and hence the latter activity is outside the TMN.

The *Mediation Function* (MF) acts on information passing between NEFs and OSFs to enhance the capabilities of “weak” NEFs. Mediation functions may store, adapt, filter, threshold and condense information. Decision making functions as well as conversion and interworking functions may be also part of MFs. Mediation functions may be implemented as hierarchies of cascaded devices.

The *Work Station Function* (WSF) provides the means to interpret TMN information for the management information user. The WSF interfaces to the OSF and the MF which have management intelligence while it is not permitted to interface to an NEF/QAF. The actual interface to the human user is not considered part of the TMN.

The *Operations Systems Function* (OSF) processes information related to telecommunications management to support and/or control the realisation of various telecommunication management functions.

Two additional functional components, concerned with message and data communications, are also defined.

The *Message Communication Function* (MCF) is associated with all functional blocks having a physical interface. It is used for exchanging management information contained in messages with its peers. The MCF provides functionality of the layers 4-7 of the OSI reference model.

The *Data Communications Function* (DCF) provides the means for data communication to transport information related to telecommunications management between function blocks. The prime role of DCF is to provide information transport mechanisms, including routing and relaying functions. It may therefore be supported by the bearer of different types of sub-networks. The DCF provides functionality of the layers 1-3 of the OSI reference model.

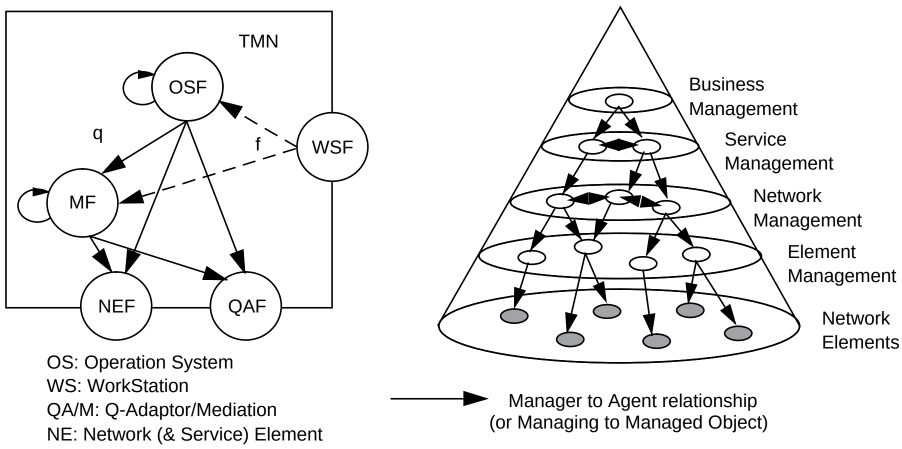

Figure 4.2 TMN functional and example physical architecture

4.1.3.2 Reference points and interfaces

Reference points define conceptual points of information exchange between non-overlapping function blocks. A reference point becomes an interface when the connected function blocks are embodied in separate pieces of equipment. Reference points are denoted by lower case letters (e.g. q_3), interfaces by upper case letters (e.g. Q_3). Interfaces are characterised by two facets: a conceptual facet and a physical facet, with the conceptual facet effectively defined by the reference point. The conceptual facet is characterised by the M-Part (Message Part) while the physical facet is characterised by the P-Part (Protocol Part) [4.5]. In the case of OSI management which is adopted in the TMN, the M-Part defines the structure of the message sent to or received from a managed object, i.e. a Common Management Information Service (CMIS) message [4.6]. The P-Part defines the protocol stack used to transfer the message and this will involve

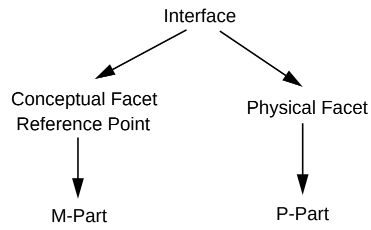

Figure 4.3 Relationship of reference points and interfaces

the selection of the profile to support the Common Management Information Protocol (CMIP) [4.7].

The q_3 *Reference Points* connect OSFs to MFs, QAFs, NEFs and to other OSFs (at whatever level they may be placed in the TMN layered hierarchy). The q_x *Reference Points* connect MFs to QAFs, NEFs and to other MFs. The *f Reference Points* connect WSFs to OSFs and MFs while *g Reference Points* connect the human users to WSFs. The *x Reference Points* connect a TMN to other management type networks including other TMNs; the x reference point is an important open issue which has relation to ownership and domains; security and access rights, and; co-operative problem solving methods.

4.2 Logical layering

One of the key aspects of the TMN as a management framework is that it transforms the old *flat centralised* management paradigm to a *hierarchical distributed* one. We will examine here the aspects and issues behind the TMN hierarchical decomposition.

In the hierarchical model projected by the TMN, management functionality is layered, offering increased abstraction and encapsulation in higher layers. The functionality of each layer builds on the functionality offered by the layer below. Each layer may contain more sub-layers. The functionality of each layer (or sub-layer) is supported by a one or more Operation System Functions (OSFs). These may have peer-to-peer relationships when in the same layer or hierarchical ones when in different layers (or sub-

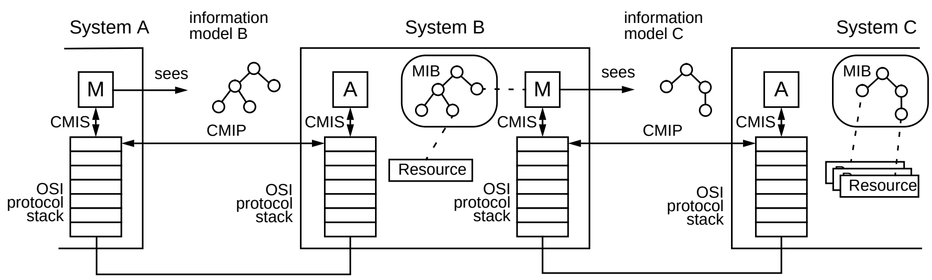

Figure 4.4 Cascaded hierarchical communication (from M.3010)

layers). Each OSF presents an information model to superior OSFs which build on it and present different information models of higher abstraction to their superiors. As such, a layer in this architecture builds on the information model presented by the layer below and presents its own information model to the layer above.

This hierarchical decomposition and the cascaded interactions in manager-agent roles and different information models at each level are depicted in Figure 4.4. The information models at each level represent manageable resources at different levels of abstraction. At the lowest level of that hierarchy, managed network elements contain resources of the finest granularity, representing aspects of the local system, e.g. access points, interfaces, call endpoints etc. At higher levels, objects represent resources of higher abstraction that can be mapped onto lower level resources, e.g. network, services etc. The mapping between information models at adjacent levels is accomplished through Information Conversion Functions (ICFs). Note that such functions are functional components of OSFs and are not shown in Figure 4.4; the latter presents only the manager-agent roles and accessible information models.

TMN layering is not strict in the sense that an OSF may access directly OSFs in layers below the adjacent underlying layer. The key aspect that characterises the TMN hierarchical organisation is that each layer is a fully incremental addition of functionality provided by the layers below. As such, lower layers are essentially unaware of the presence of higher layers; they simply respond to management requests which are always initiated by the higher layers. Manager-agent interactions are only top-down or peer-to-peer in this layered hierarchy. Note that event notifications are always initiated (i.e. requested) by the superior system.

Another way to view the hierarchical nature of OSFs and the manager-agent relationships is depicted in Figure 4.5. This is known as the TMN Logical Layered Architecture (LLA). The LLA uses a recursive approach to decompose a particular

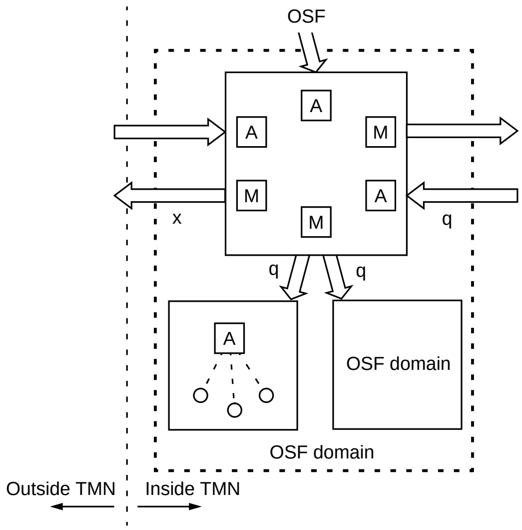

Figure 4.5 TMN logical layered architecture (from M.3010)

management activity into a series of nested functional domains. Each functional domain becomes a management domain under the control of an Operation System Function (OSF). A domain may contain other OSF domains to allow further layering and/or may represent logical or physical resources as managed objects at the lowest level of the hierarchy within that domain. When the OSF domain is at the top of the layered architecture, there is no superior OSF.

All interactions within a domain take place at generic q reference points. Interactions between peer domains, i.e. crossing the OSF domain boundary can take place at a q or x reference point. Reference points may become interfaces depending on the physical realisation. The essence of the LLA is that at each point of interaction with an OSF domain, all the other subordinate OSF domains used to accomplish a management task can be thought as encapsulated (contained) by that domain. The user of a management service or sub-service offered by that OSF does not see its realisation.

The 1991 version of M.3010 [4.1] indicated a potential decomposition of TMN layered functions in element, network, service and business management layers. These were influenced by the early British Telecom Co-operative Network Architecture for Management (CNA-M) [4.8]. This layered decomposition was only informative and was presented in an Appendix of M.3010. In the 1996 version of M.3010 [4.2], this layered decomposition was moved in the main part of the document and is now more pronounced. The potential functions of those layers of management activity are the following.

The *Network Element Management Layer* comprises functions related to a number of network elements. These functions may be concerned with maintenance, configuration and performance monitoring. They will not, however, take any account of network-wide aspects and will have only a limited view of other parts of the network. They will be predominantly technology and vendor dependent.

Network Management Layer is concerned with keeping the network at some operational optimum. Human and automatic decision making processes may be used to optimise the network resources. This layer supports the management of the whole network. Complete visibility of the whole network is provided while a vendor independent view is normally required. At this level, provision, cessation and modification of network capabilities are possible to support services. Statistical data for the whole of the network are kept regarding performance, usage, availability etc.

The *Service Management Layer* contains functions with respect to services, which may be implemented across more than one network. Customer facing (provision, cessation, accounting, QoS, fault reporting) and interfacing with other administrations are aspects of this layer. Additional tasks include interaction between services, service order, complaint handling and invoicing. At this level, a network technology independent view of services is possible.

Business Management is concerned with the implementation of policies and strategies within the organisation which owns and operates the services and possibly the network itself. This is influenced by higher level controls such as legislation or macroeconomic factors. It might include tariffing policies, quality maintenance strategies giving guidance on service operation, and so on. It seems likely that the majority of these functions will not be automated in the near future.

The ICM's view of this layered decomposition is that it constitutes a good initial hierarchical partition of management responsibilities. It should though stay descriptive rather than be a prescriptive concept. In addition, ICM's view is that the logical layered decomposition should be realised by relatively small OSFs, some of which will model generic service components applicable in various contexts, rather than following a monolithic approach with one OSF for each proposed layer.

4.3 Distribution and discovery

The initial version of the TMN architecture as described in [4.1] did not support open distribution and discovery services and the dynamic acquisition of shared management knowledge. The defined reference points (q, x, f) supported only management information exchange, assuming information about the location and capabilities of other applications was acquired through ad-hoc mechanisms. In fact, the first generation of TMN systems in the first half of the '90s used to hard-wire such information in local databases used by each application. This approach makes it difficult to ensure global consistency and cannot support distribution in terms of location and other transparencies.

Research in RACE (ICM, PREPARE) and elsewhere during the same period resulted in efforts to provide open mechanisms for distribution and discovery services. The use of the OSI Directory Service (X.500) [4.9] has been proposed to complement OSI Management (X.700) [4.3] to form together the base technologies for the TMN. The key advantage of the directory is its federated nature which is important for inter-domain management as it allows the seamless exchange of information across domains. In the context of the TMN, the directory is used as a repository of information regarding TMN applications. The required directory objects have been specified in the Shared Management Knowledge Function (X.750) [4.10].

One important aspect of the directory is its hierarchical naming architecture which is very similar to that of OSI management. As such, it has been proposed to unify the two spaces in one global name space, considering management as an extension of the directory name space. The concept is depicted in Figure 4.6 which presents the extended manager-agent model through the use of the directory. Managed objects in an application in agent role such as a TMN OS may be addressed either through local names, e.g. {logId=1, logRecordId=1234}, or through global names starting from the root of the directory tree, e.g. {c=GB, o=UCL, ou=CS, cn=ATM-OS, networkId=ATM, logId=1, logRecordId=1234}. Using global names provides a way to deal with distribution through location and other transparencies.

The directory itself is realised through many Directory Service Agents (DSAs) that administer parts of the global directory tree. All the DSAs are organised in a fashion that they can *chain* or *defer* requests for objects they do not support either as master copies or via caching; they collectively provide a global federated information store. Applications accessing the directory assume the role of a Directory User Agent (DUA) and may access either a local DSA, which may result in chaining or deferring the request, or directly access a remote DSA.

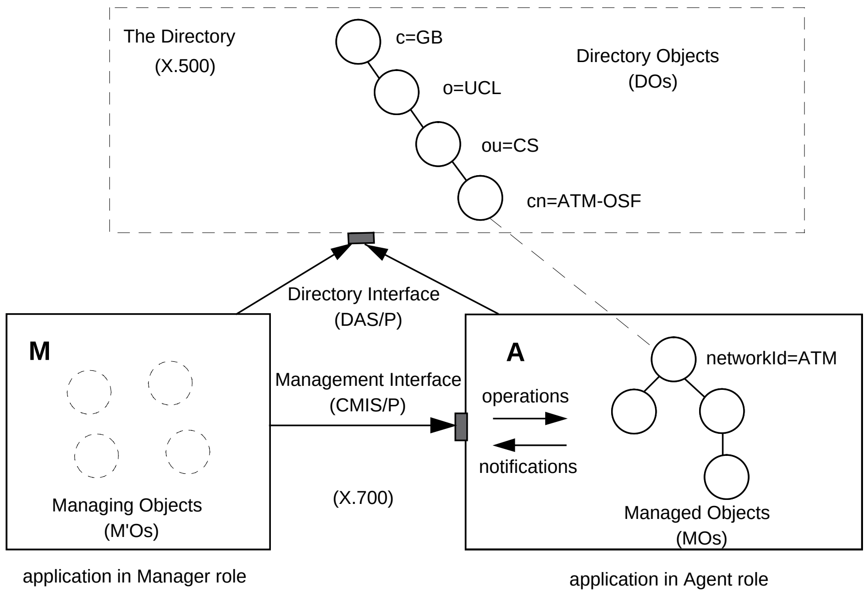

Figure 4.6 The management model and the global management-directory name space

As stated, the TMN model should be enhanced to allow open distribution and discovery services through the directory. Initially in ICM and PREPARE a new function block was introduced, the Directory System Function (DSF). The idea was that this should be accessed through a ***d*** reference point. This immediately poses the problem that additional reference points are required for DSF to DSF interactions: one for interactions within a single TMN and one for directory interactions between TMN domains. As this solution complicates the overall architecture with the addition of new reference points, it was abandoned in ICM. The proposed solution is to enhance the q and x reference points to support directory interactions and is in line with the latest developments in the official TMN architecture [4.2].

The ICM proposal for enhancing the TMN architecture is shown in Figure 4.7. A new Directory System Function (DSF) block is introduced which provides the basic directory capability. This contains the Directory System Support Function (DSSF) that provides its main functionality, a Security Function (SF) and other functional components that are under study (see Table 4.2). Other TMN function blocks that need to access the directory (OSF, MF, QAF, NEF) contain a Directory Access Function (DAF) which allows access and maintenance of TMN-related information. Intra-TMN directory access may occur through the q reference point while the x reference point is used for inter-TMN directory access. These interactions cover both directory access (TMN block DAF to DSF) and directory system (DSF to DSF) aspects as depicted in Figure 4.7. Note also that DSF-like function blocks may also exist totally outside TMNs.

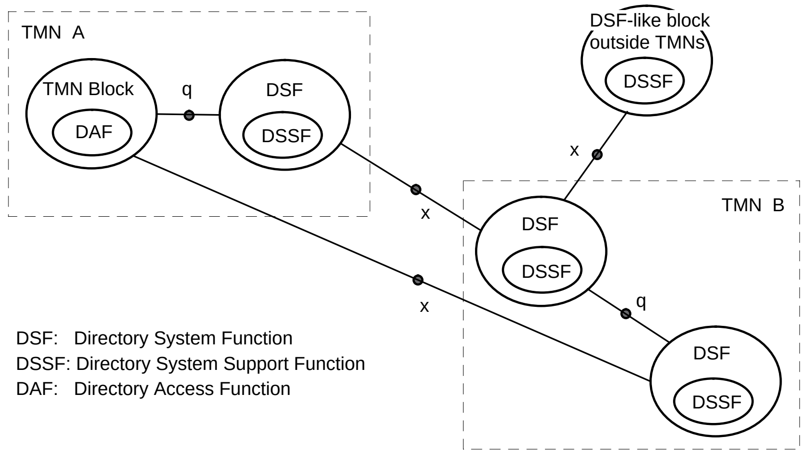

Figure 4.7 TMN and directory integration

4.4 TMN function blocks

4.4.1 Adaptation and mediation

Adaptation and Mediation are similar concepts as they both perform information and protocol conversion and enhancement functions. The key difference is that adaptation converts from reference points which are entirely non-TMN-compliant, i.e. various types of m, while mediation enhances various classes of “weak” q_x reference points to full q_3. Adaptation may yield either q_3 or (more commonly) q_x reference points. In the latter case, mediation is needed to complement adaptation.

4.4.1.1 Aspects of adaptation

The Q-Adaptor Function is used to connect to the TMN network elements which do not support the standard TMN interfaces. Q-Adaptors provide information conversion functions from a proprietary information model and the associated access protocol to a TMN compliant information model and protocol. The information models may be fully object-oriented, simply object-based (e.g. SNMP) or even a character-based serial line command mechanism. Typical older telecom-type interfaces are TL-1 (Transaction Language 1) and TBOS (Telemetry Byte-Oriented Serial protocol). The Q-Adaptor Function should convert a proprietary m model to the semantically equivalent q_3 or q_x one. Powerful querying aspects of the Q_3 access service such as scoping/filtering may need to be emulated through a series of operations across the M interface.

The nature of adaptation is depicted in the upper part of Figure 4.8. Note that this is similar to the nature of mediation class a as explained in the next section. Both adaptor and class a mediation functions are hybrid units with respect to the manager-agent

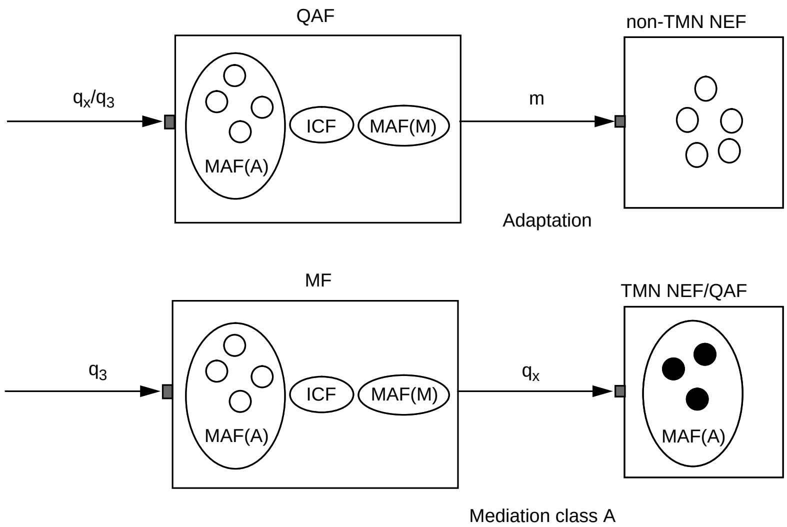

Figure 4.8 Adaptation and mediation class A

model: they access a lower level information model through m or q_x (MAF(M)) and convert it (ICF) to a q_x/q_3 or q_3 reference point respectively (MAF(A)).

Ideally, automatic conversion between information models and associated access mechanisms is desirable as it may be used to automate the Q-Adaptation mechanism. This is only feasible for management frameworks of a similar nature. In principle, if Z is the target framework (i.e. OSI management in the case of TMN) and X a proprietary one, generic adaptation is possible *if and only if*:

$$X \subseteq Z$$

This *pure subset* relationship holds between Internet and OSI management. Research under the auspices of the Network Management Forum to which ICM has greatly contributed has produced industry standards for the generic conversion between the two [4.11]. Their existence makes possible the TMN-based management of SNMP-capable network elements and management systems.

Generic information model conversion together with rules for the relevant access protocol conversion leads to fully automated adaptation. The resulting "raw" model can be augmented and enhanced by the generic capabilities of the OSI Systems Management Functions as for any other TMN interface. The resulting model though is bound to be syntactically different to the equivalent standard TMN model due to the generic nature of the conversion. Semantics will be maintained but it will be in essence a q_x rather than a q_3 reference point. A further mediation step is necessary to achieve full TMN compliance (see next).

4.4.1.2 Aspects and taxonomy of types of mediation

According to the precise TMN definition, Mediation Functions may adapt, store, filter, threshold and condense information. A mediation function acts on information passing between an OSF and NEF or QAF and essentially adds management capabilities to "weak" network elements or groups them together. In fact, a mediation function enhances a q_x reference point and produces a more capable q_x or a fully capable q_3 one. As this procedure may take several iterations, a recursive cascaded structure of mediation functions is possible.

The following types of "weak" q_x reference points and subsequently classes of mediation functions may be taxonomised:

- the information model is non standard (class a),
- the protocol stack is non standard (class b),
- the CMISE and SMASE capabilities are limited (class c).

Another potential aspect of mediation functions is that of grouping various network elements together; this may be combined with any of the above mediation functionality that enhances q_x towards q_3. We will examine closely the aspects of each of these three classes of q_x and mediation.

Class a (see the lower part of Figure 4.8) is similar to adaptation but the non-standard information model is q and not m (i.e. GDMO/ASN.1). This is typical of early TMN-capable elements, implementing early views of relevant developing standards with proprietary changes/additions to fill in gaps and to provide the desired management functionality. Converting those to q_3 is a task that cannot be easily automated. If the two models are semantically equivalent, it should be possible to describe their relationship in a meta-language which, through suitable tools, could be parsed to produce the mediation logic. Further research is necessary to achieve this level of automation. As mentioned in the previous section, generic adaptation between SNMP and OSI Management yields a q_x reference point. An additional level of mediation is necessary to achieve q_3 and as this cannot be easily automated, so the advantages of generic adaptation are partly lost.

Class b refers to data network interworking with respect to non-standard q_x protocol stacks. Note that this is higher layer protocol interworking, any lower layer interworking required due to different data network technologies should be transparently addressed by network layer relays. Examples of non-standard q_x protocol stacks include CMIS/P Over LLC (CMOL) or CMIS/P over TCP/IP (CMOT). Note that CMIS/P over TCP/IP using the RFC1006 method has been recently endorsed as a valid Q_3 lower layer stack profile in Q.811. Network layer relays may be used to provide DCN continuity over different domains. There are 1*N=N types of such mediation devices, translating between full Q_3 and each of the N types of possible non-standard Q_x stacks. It should not be difficult to provide such mappings for protocol translation. This type of mediation device is depicted in Figure 4.9.

Class c mediation functionality is necessary to overcome the inability of q_x to support all CMISE and SMASE capabilities: filtering (CMIS filtering), storage (logging, persistency), thresholding (metric monitoring) and condensing (summarisation) attributes of mediation. Other related aspects such as security/access control, testing, accounting etc. may also be included. Providing this type of functionality involves "mirroring" the subordinate q_x information model and enhancing it with the missing properties so that it becomes q_3. This enhancement may take place through a series of cascaded mediation functions, possibly grouping each a number of similar elements together. If we wanted to depict pictorially this class of mediation function, it would be

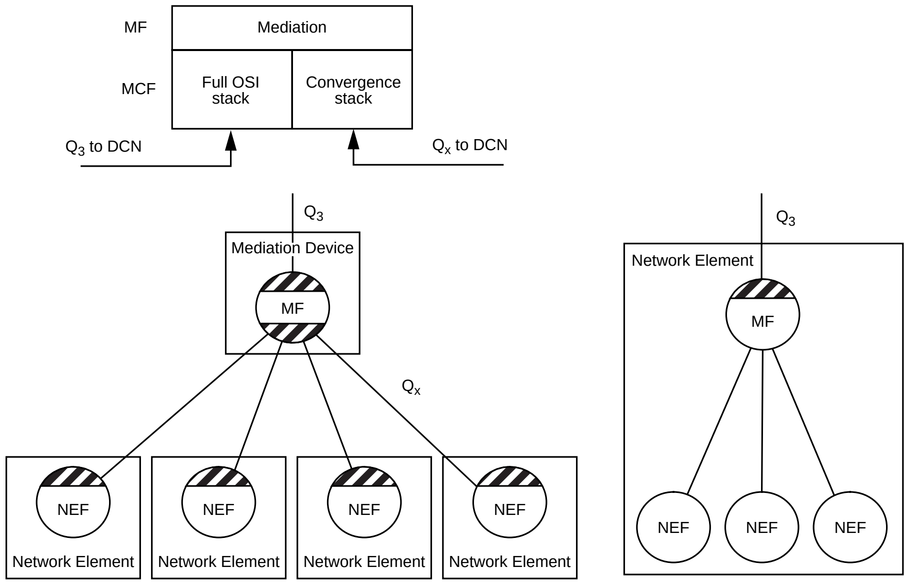

Figure 4.9 Mediation class B and element grouping

similar to class a (Figure 4.8) but the two resource information models would be exactly the same and as such there is no need for an information conversion function. The aspect of grouping elements together is depicted on the second part of Figure 4.9.

Table 4.1 shows the various aspects of M, Q_x and Q_3 interfaces. M does not have any q reference point aspects (CMIS/GDMO) and as such it has none of the three interface aspects as well. Q_x must have at least the q reference point aspects, though it does not have all of the Q_3 interface aspects: conformant protocol stack, CMISE/SMASE capabilities and standard GDMO information model. As such, we can actually characterise 7 types of Q_x: one that has none of those three aspects, three that have one only and three that have two. It should be finally noted that Q_x interfaces are *not* expected to be standardised.

i/f type **i/f aspect**	**M**	**Q_x**	**Q_3**
q reference point	x	m	m
a. standard protocol stack	x	o	m
b. CMISE and SMASE FUs	x	o	m
c. standard GDMO model	x	o	m
	x: not present	o: optional	m: mandatory

Table 4.1 Aspects of M, Q_x and Q_3

Summarising, classes b and c of mediation functions are relatively straightforward to provide. Class a is similar to adaptation but concerns translation between GDMO information models that are semantically similar but syntactically different. Research towards this direction may provide generic solutions to minimise the relevant development and investment and will complement generic adaptation mechanisms.

4.4.2 Workstations

Workstation functions (WSFs) provide the means to support human users of management services in realising management decisions or performing supervisory activities. They essentially provide the means to interpret TMN information to be presented to the human user and vice-versa. A separate reference point and corresponding interface (f and F respectively) realise the logical and physical communication between workstation functions and operations system or mediation functions.

According to the TMN functional architecture, workstation functions are not connected to either network element or q-adaptation functions. The reason for this restriction may be that human users should not be allowed direct access to elements but only through the use of management functions provided by OSFs and MFs so their decisions are implemented in a controlled fashion. Though according to M.3010 [4.1] MFs may support decision functions, the analysis in the previous section suggests that there are no such aspects of mediation but only q_x to q_3 enhancement aspects. As such, ICM's view is that WSFs should not interface directly to MFs but only to OSFs.

While the f reference point and F interface have been an integral part of the TMN architecture from its early stages, there have never been efforts towards their standardisation within ITU-T. The only tangible output has been recommendation M.3300 [4.12] which explains further the workstation concept and presents a number of management capabilities required at the F interface in the various functional areas: performance management (traffic measurement reporting, performance monitoring), fault management (alarm surveillance, testing, trouble ticket administration), configuration management (service order management, resource configuration), accounting and security management.

In the absence of standards for the F interface, some important decisions have been made within ICM:

- f, F are transaction-oriented in a similar fashion to q, Q
- the interactions they support can be mapped onto q, Q; or otherwise, they may be derived from q, Q

From an architectural point of view, this implicitly means that direct workstation access is allowed to any q-capable functional block, including mediation, adaptation and network element functions. Given the previous discussion though, WSFs should only access OSFs: security services (i.e. authentication) may be used to restrict direct access to MFs, QAFs and NEFs.

The decision with respect to the f reference point was reinforced by the observation that messaging capabilities at the f reference point are high-level “directives” that may be decomposed to a series of requests across q reference points. Or otherwise, the information required to support the human user in the context of ICM could be derived from management information present in functional blocks with q capabilities. In fact,

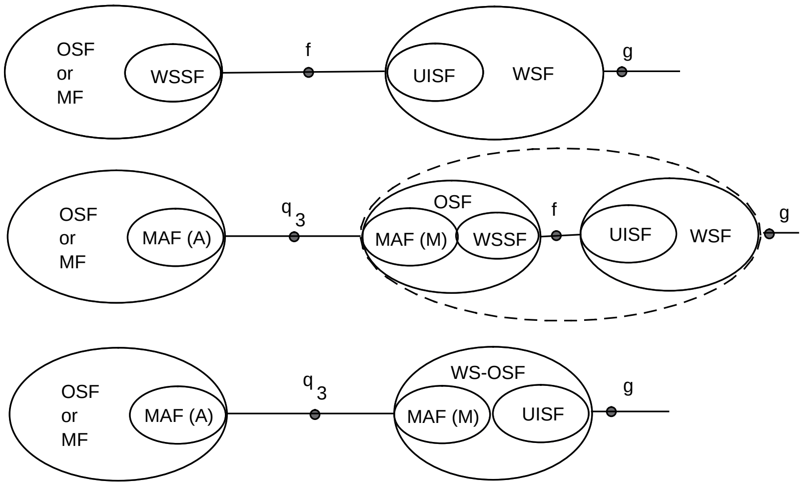

Figure 4.10 The workstation-operations system function block

the q information model can be designed in such a way as to support WSF requirements, in which case the observation about the relationship of q and f can be generalised.

As such, in the ICM TMN architecture there is no f reference point. Figure 4.10 presents the rationale behind that decision. The upper part depicts the notion of a WS as in M.3010. Given the fundamental assumption that f can be derived from q_3 as explained above, the mapping of f to q_3 can be supported by an OSF in managing role. The Workstation Support Function (WSSF) in the latter receives f messages and through a Management Application Function in Manager role (MAF(M)) converts these to q_3 messages (middle part of Figure 4.10). This mapping may not be one-to-one but is always possible if the necessary information is available in q_3 form. Given the fact that we do not expect f to be standardised as it can be "derived" from q_3, we have introduced a new TMN function block called Workstation-Operation System Function (WSF). The latter has a User Interface Support Function (UISF) and a MAF(M) as functional components and may connect to normal OSFs (lower part of Figure 4.10). As a consequence of this architectural decision, the WSSF functional component is no longer necessary.

Finally, it should be possible to run workstations on inexpensive platforms, e.g. personal computers with limited resources. Given the fact that Q_3 is considered generally expensive to implement and operate on such platforms, special lightweight versions of Q with respect to the protocol stack may be used. These are essentially Q_x protocols according to the analysis in the previous section. In those cases, a protocol converter is required to translate them to Q_3. Such converters are essentially "mediation functions" that operate in the opposite direction (they translate Q_3 to Q_x). As the TMN architecture does not cater for such "inverse" mediation functions, we will assume that such translators are part of the WS-OS physical block as an implementation convenience. A string-based CMIP protocol operating directly over a reliable transport service (avoiding the OSI upper layer, and especially the presentation, overhead) is described in [4.13]. This could be used to support such inexpensive workstations.

4.5 TMN functional components

From the early days of the TMN, a number of functional components have been identified as the elementary building blocks of the TMN function blocks. These are not subject to standardisation but help to identify and understand better the internal functionality of the function blocks (NEF, MF, QAF, OSF, WSF). Only the reference points among the latter are subject to standardisation.

Given the previous discussion on distribution, mediation/adaptation and workstation functions, a number of modifications/additions to the functional components and their relationship to function blocks are suggested. These are summarised below:

- A new function block, the Directory System Function (DSF) is introduced; this is only a functional component in the current TMN architecture.
- A new functional component, the Directory System Support Function (DSSF) is introduced to provide the main DSF functionality.
- The Workstation Function is replaced by the Workstation-Operation System Function (WS-OSF).
- As a consequence, the f reference is not visible externally and is essentially replaced by q_3.
- As a second consequence, the Workstation Support Function (WSSF) is no longer necessary.

The revised relationship of function blocks to functional components is shown in Table 4.2. Note that there is no longer a f message communication function. Note also that the QAF_{qx} function block has also a MCF_{q3} messaging capability: this is due to its DAF functionality for DSF access.

Function block	Functional components	Associated message communications functions
OSF	OSF-MAF(A/M), ICF, DAF, SF	MCF_x, MCF_{q3}
WS-OSF	UISF, OSF-MAF(M), DAF, SF	MCF_{q3}
NEF q_3	NEF-MAF(A), SF	MCF_{q3}
NEF q_x	NEF-MAF(A), SF	MCF_{qx}
MF	MF-MAF(A/M), ICF, DAF, SF	MCF_{q3}, MCF_{qx}
QAF q_3	QAF-MAF(A/M), ICF, DAF, SF	MCF_{q3}, MCF_m
QAF q_x	QAF-MAF(A/M), ICF, DAF, SF	MCF_{qx}, MCF_{q3}, MCF_m
DSF	DSSF, SF, other	MCF_x, MCF_{q3}

Table 4.2 Relationship of function blocks to functional components

As the Operation System Function is the most important and complex function block of the TMN, its structure in terms of functional components is shown in Figure 4.11. The Management Application Function in agent role (OSF-MAF(A)) provides support for the information model exported across the q and x reference points. It receives messages requesting operations on managed objects and sends back results while it emits asynchronous event reports according to dynamically pre-defined criteria. Inter-TMN requests across the q reference point originate from other peer or superior OSFs in the logical layered hierarchy or from WS-OSFs. The Security Function (SF) allows only authorised entities to communicate with the OSF (authentication) and controls the level of access to management information (access control).

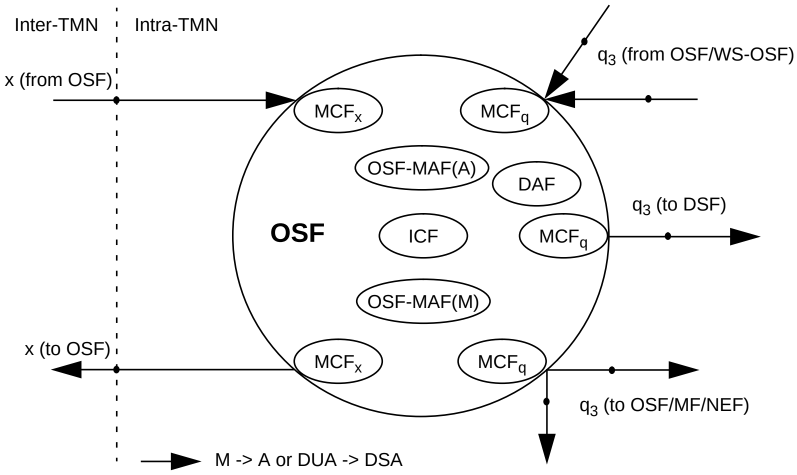

Figure 4.11 Example OSF decomposition to functional components

The Management Application Function in manager role (OSF-MAF(M)) accesses management information made available in other function blocks across the q and x reference points. It performs operations on managed objects and receives asynchronous event reports. It communicates with other OSFs across the x reference point and with other peer or subordinate OSFs and subordinate MFs/NEFs across the q reference point. The OSF-MAF(M) embodies the "managing intelligence" of the OSF. The Information Conversion Function (ICF) bridges the two MAFs by converting information in both directions, e.g. correlating event reports and providing a consolidated report "upwards" or translating operations on managed objects available by this OSF to finer granularity operations on subordinate information models "downwards."

Finally the OSF needs to access and maintain directory related information for shared management knowledge. The Directory Access Function (DAF) accesses the directory on behalf of OSF-MAF(M)s to discover information about the location and capabilities of other function blocks that should be accessed in a managing fashion while it maintains information in the directory on behalf of OSF-MAF(A)s regarding their location and capabilities to be accessed in an agent fashion

4.6 The architecture revisited

In the previous three sections, a number of clarifications, modifications and additions to the TMN architecture have been discussed. Summarising, these addressed the following:

- distribution and discovery through directory services; a Directory Service Function (DSF) block was added to the functional architecture;
- classes of mediation were taxonomised and adaptation/mediation functions were related and explained;
- the Workstation Function (WSF) was replaced by the Workstation-Operation System Function (WS-OSF) and the f reference point was "collapsed" to q_3;
- a modified and simplified relationship of function blocks to functional components was presented.

Given these modifications, the revised TMN functional architecture is presented in Figure 4.12. The most important changes are the addition of a new function block (DSF) and the fact there is no f reference point.

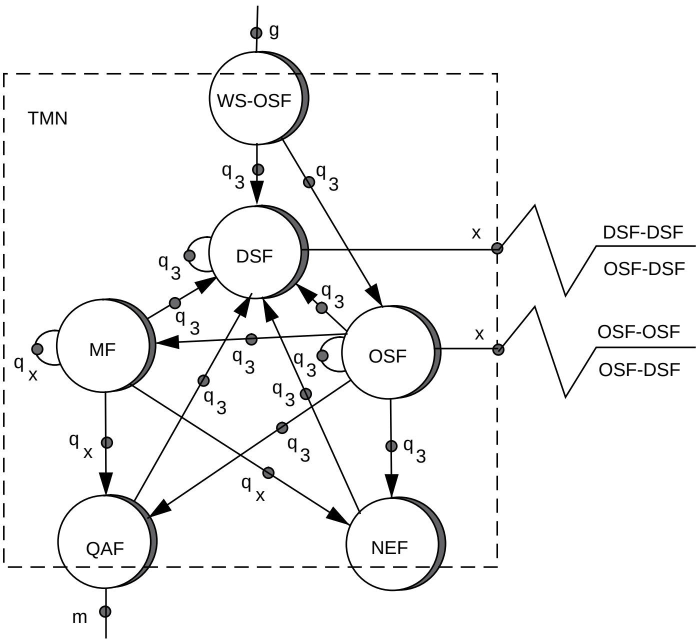

Figure 4.12 Revised TMN functional architecture

The interactions between function blocks across reference points have now direction that denotes a "client-server" relationship. The latter may be manager-agent for management interactions, initiator-responder for file transfer interactions and directory user-directory service for directory interactions.

Mediation, operations system and directory system functions have also a recursive relationship to themselves, shown without direction in Figure 4.12. The relationship between mediation functions is due to cascaded mediation organisation and is always of superior-to-subordinate nature. The relationship between operation system functions may be either superior-to-subordinate or peer-to-peer. Finally, the relationship between directory system functions is always peer-to-peer, supported by Directory System Protocol (DSP) interactions for chaining functions.

The OSF, WS-OSF, MF, QAF to DSF interactions across the q_3 reference point are directory access interactions supported by the Directory Access Protocol (DAP).

The interactions across the x reference point require some explanation. An OSF may interact with other OSFs in peer-to-peer (client-server in both directions), superior-subordinate (client-server) or subordinate-to-superior (server-client) fashion. An OSF may also access the DSF of another TMN directly, as opposed to indirectly through chaining after accessing the local DSF. In this case the relationship is a directory client-server one. The DSF may access a DSF in another TMN through chaining (peer-to-peer relationship) or to be accessed directly by an OSF of another TMN (directory server-client relationship).

4.7 Discussion

In this chapter an introduction to the TMN architecture has been presented followed by a discussion on various issues addressed in the context of the ICM project. As a result, the TMN functional architecture has been reconsidered. As stated previously, no additional reference points were defined while the f reference point has been removed. Interactions across the q and x reference points are now enhanced to include management, directory transactions and file transfer. Let's now qualify exactly how a TMN interface is specified and how the overall capabilities of a TMN physical block should be expressed in terms of both exported and imported interfaces (the latter for blocks other than NE and QA).

A Q_3 or X interface always has a manager-agent or client-server duality: an interface is *exported* by an application in agent or server role or it is *imported* by another application that acts in manager or client role. Interfaces are only specified from the agent or server side but the capabilities of a TMN application such as an operation system are expressed through both exported and imported interfaces; the latter are necessary for it to be able accomplish its management functions. As such, a TMN application with managing aspects other than NE/QA is qualified through the description of both exported and imported interfaces.

As TMN interfaces are transaction-oriented, specifying the capabilities of a Q_3 or X interface has two aspects:

- specify the entities that are made available across the interface in terms of their syntax and semantics, e.g. management and directory objects, files;
- specify the protocols used to convey these entities between systems.

The second aspect should specify the exact protocol stack profile as in the Q.811 and Q.812 standards (lower and upper layer Q profiles). The first aspect should specify the management, directory and file information model.

The management part of a Q_3 or X interface is specified in terms of a GDMO model which constitutes an *ensemble,* i.e. the relevant object instances collectively provide a management service or sub-service. The exact capabilities implemented should be mentioned, e.g. optional aspects included etc. This information should be also made available through the directory as shared management knowledge. It is noted here that semantic aspects in GDMO are specified as plain text. The use of an ensemble by applications in managing roles may be further explained through a description containing a set of messages directed to objects of that ensemble. This description is informal, suggesting potential use of that information model.

The directory aspects of Q_3 and X involve the use of the directory server (DS) by an application to either notify/update it with its location and capabilities or to learn about these with respect to other applications. It is reminded that all TMN applications have OSI Application Process names which map onto a unique directory name, e.g. {c=GB, o=UCL, ou=CS, cn=ATM-RD-OS}. The directory information model for shared management knowledge activities has been specified in X.750, so it suffices to mention that an application supports it.

Finally, the use of file transfer for management has not yet been thoroughly addressed. The accessible filestore has to be named, including the available file names and the syntax and meaning of their content. The latter is necessary for the two communicating entities to be able to interpret them. There is no formal way to describe such aspects, so natural language should be used.

A key aspect in the TMN is that its functional entities are essentially composite computational components or building blocks for management services, presenting capabilities through a managed object cluster made available at the exported interface(s). Internally, within each such functional entity there may exist other computational entities which are not visible externally. Managed objects and other internal computational objects have relationships and should be specified from an information viewpoint using OMT [4.14]. This should be used in addition to GDMO which is used for managed object specification only. It should be added that the modelling and placement of computational logic can be arbitrary, in the sense that managed objects may also have computational properties and not be simply information objects. The computational aspects of all these objects, i.e. internal behaviour, state and response to stimuli can be specified using formal description tools such as SDL [4.15]. The mapping of these objects to engineering objects depends on the relevant platform infrastructure. In ICM, managed and other internal computational objects are mapped onto C++ object instances using the OSIMIS platform (Chapter 10).

One particularly important aspect of the object cluster model used in the TMN is the existence of generic functionality available at every management interface. This functionality is provided by the OSI Systems Management Functions (SMFs) which specify generic support managed objects that may be instantiated to support such functionality. Systems management functions can be taxonomised as follows:

- basic generic capabilities for object, state, relationship and alarm management,
- system capabilities, comprising event reporting and log control,
- generic security support functions, including access control, security alarm reporting and security audit trail,
- other generic support facilities, including metric monitoring, summarisation, accounting, testing, scheduling and many more.

These are all invaluable but the last category in particular can be thought as “enhancing” the information model available at every interface. As an example, metric monitoring, summarisation and logging can enhance raw data such as transmitted and discarded octets across an interface to provide throughput and error rates, QoS alarms and historical trend data over time. This information can be provided by simply instantiating the relevant support managed objects and initiating the relevant functions.

An accusation often directed to the TMN is the lack of computational specifications complementing the GDMO interface definitions. It should be mentioned that according to ITU-T, the TMN is first and foremost a communications concept. As such, the lack of such specifications is deliberate; specifications are concerned with *what* is available at a management interface rather than *how* this functionality will be provided. As explained, additional information specifications in OMT and computational ones in SDL may complement the GDMO interface definitions. Such specifications are deliberately left outside the scope of TMN standardisation by the ITU-T.

A related accusation has to do with the fact that specific TMN OSFs have not been identified in the various TMN layers to support management services. This has some ground and in ICM various types of OSFs supporting different management services have been identified. Identifying such building blocks achieves better re-usability and may lead towards an open market of TMN applications. On the other hand, this will restrict the ways in which a management service will be provided through OSFs.

Finally, another accusation has to do with the inherent asymmetry of the manager-agent model adopted in the TMN. It is true that this model could be limiting if the underlying engineering concepts separate completely its two aspects. In flexible object-oriented support environments such as the OSIMIS platform that was used in ICM, this separation is not strong at all: a managed object may as well act as a managing object in either peer-to-peer or hierarchical fashion. It is the client-server aspect that underlines their relationship, in the same fashion as in other distributed systems models and platforms.

This chapter is concluded by highlighting that according to the ICM view, the TMN *is* a hierarchically-structured distributed processing environment. The unit of distribution is the TMN physical block (e.g. OS), where there is usually more than one such physical block in every TMN layer in peer-to-peer or hierarchical relationships. Finally, the unit of re-usability can be, in ascending order of “size” and contained-in relationships, the managed object, the managed object cluster that models a management service component, the TMN functional/physical block (OSF/OS) and finally the management service.

4.8 References

[4.1] ITU-T M.3010, “Principles for a Telecommunications Management Network,” Working Party IV, Report 28, 1991.

[4.2] ITU-T M.3010, “Principles for a Telecommunications Management Network,” Working Party IV, Working Draft C, 1995, (revised version due 1996).

[4.3] ITU-T X.701, “Information Technology - Open Systems Interconnection - Systems Management Overview,” 1991.

[4.4] RACE CFS H400 series, “TMN Design, Planning, Installation, Provisioning, Maintenance, Performance, Security, Accounting and Customer Query and Control Services,” Issue B, 1991.

[4.5] Guideline ME8, “TMN Implementation Architecture,” Guideline Deliverable 03/DOW/SAR/DS/B/012/b3, 1992.

[4.6] ITU-T X.710, “Information Technology - Open Systems Interconnection - Common Management Information Service Definition,” Version 2, 1991.

[4.7] ITU-T X.711, “Information Technology - Open Systems Interconnection - Common Management Information Protocol Specification,” Version 2, 1991.

[4.8] “Co-operative Network Architecture Technical Overview,” CNA TD 001 Part 1, British Telecommunications plc, December 1988

[4.9] ITU-T X.500, “Information Processing, Open Systems Interconnection - The Directory: Overview of Concepts, Models and Service,” 1988.

[4.10] ITU-T X.750, “Information Technology - Open Systems Interconnection - Systems Management - Management Knowledge Management Function,” 1995.

[4.11] NMF IIMC, ISO/ITU-T and Internet Management Co-existence (IIMC), Forum 026: Translation of Internet MIBs to ISO/ITU-T GDMO MIBs, Forum 028: ISO/ITU-T to Internet Management Proxy, 1993.

[4.12] ITU-T M.3300, “TMN Management Capabilities Presented at the F Interface,” Working Party IV, Report 28, 1991.

[4.13] Pavlou, G., “LCMIP: A Lightweight Protocol for Data and Telecommunication Network Management and Control,” UCL CS Research Note RN/95/61.

[4.14] Rumbaugh, J., et al., “Object-Oriented Modelling and Design,” Prentice-Hall, 1991.

[4.15] ITU-T Z.100, “Specification and Description Language SDL,” 1989.