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 3

The ICM methodology for TMN system design

Authors: David Griffin, George Pavlou

ICM achieves its objectives through the development of a TMN testbed, developed in an incremental fashion over the lifetime of the project. This means that the version of the testbed from a previous phase is the starting point of the next phase of the project's work - in this way the project builds its own specialised tools for development (such as a TMN platform) and the maximum amount of both specification and software reuse is achieved within its lifetime. By following this approach the project is divided into a number of phases whereby each phase produces results which are used as a major input for the next phase.

The approach to developing the testbed in each phase is:

- to select and specify case studies within the area of configuration and performance management,
- to design a TMN system to meet the requirements of each case study,
- to reuse the testbed infrastructure and the available components from previous phases, enhancing them where necessary,
- to develop new components according to the requirements of the case study and integrate the new and enhanced components with existing ones in the testbed,
- to define test suites for validating that the components have been implemented as specified,

- to conduct a number of experiments to demonstrate the effectiveness of the management system in terms of network efficiency improvements, improved Quality of Service and so on,
- to analyse the results of the experimental work and perform a cost-benefit analysis on the developed system.

The above steps describe the overall methodology of the ICM project. The remainder of this chapter deals with a more detailed methodology for designing and developing TMN systems (a description of the validation methodology is given in Chapter 12).

The ICM approach builds and expands on the methodology recommended by the ITU for specifying, designing and developing TMN interfaces. The resulting methodology becomes the ICM project's method of work but we believe that the methodology is of more general utility and, as such, is suitable for adoption by designers of other TMN systems.

3.1 The ITU M.3020 interface specification methodology

The starting point of the ICM methodology is the ITU recommendations on TMN interface specification methodology, management services and management functions. Recommendation M.3020 [3.2] defines the TMN interface specification methodology; the Network Management Forum (NMF) "Ensemble" concept [3.13] is also of relevance. It should be mentioned that M.3020 has evolved over the lifetime of ICM. The latest version, expected to be officially published during 1996, is more aligned with the NMF ensemble concept and no longer uses *service components* and *functional components* as constituents of management services. ICM started with the 1991 version and it is precisely in the area of constituents of management services and hierarchical decomposition where it expands significantly on the ITU methodology. In this chapter and the rest of the book, the 1991 M.3020 document is used as the basis, but references to the 1996 version are also made.

According to M.3020, a management service is defined as:

> *An area of management activity which provides for the support of operations, administration, provisioning and maintenance of the managed network. It addresses the relevant information serving a specific management goal. It is always described from the user perception of the management requirements.*

The management service definitions are not intended to imply any implementation aspects of management functionality. The exact way that management services and management service components are allocated to TMN building blocks (OSs, etc.) is left to the TMN system designers.

As necessary management services are identified, they are registered in Recommendation M.3200 [3.4] and other future M.32xx recommendations. However, the management services themselves are not subject to standardisation by the ITU, they are only used for the specification of management interfaces which are the main focus of standardisation activities by the ITU. Management services are simply registered as encountered in this process in order to avoid duplication of effort and achieve re-usability of relevant functions by interface specifiers. Management service definitions are

also used by TMN designers and are the starting point of the ICM methodology for specifying and designing TMN systems.

The M.3020 methodology considers the TMN as a single entity, providing management services to TMN users, the latter being either human or automated processes (Figure 3.1).

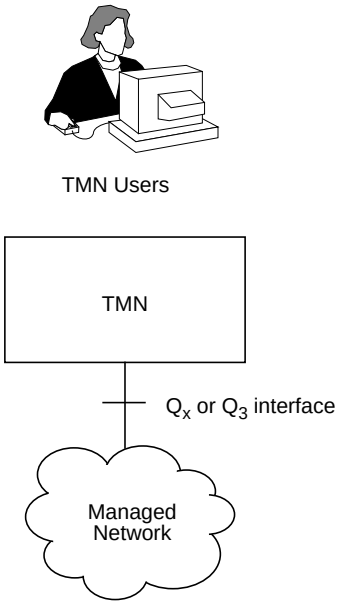

Figure 3.1 Scope of the M.3020 TMN interface specification methodology

The result of following the methodology is a specification of the interface between the managed network and the TMN, where the managed network may be a single network element or a whole network depending on the focus of the pass through the methodology. The specification is intended to cover both the information modelling aspects and the management protocol to be used for conveying the management information (tasks 6 & 7). However, given the fact that protocols for conveying management information have by now been specified (CMIS/P, FTAM), the last two steps are no longer necessary but are presented for completeness. The methodology presented here covers both the initial M.3020 and its latest evolution.

A conceptual *Task 0* provides the background to the methodology, by pointing to already existing generic network information models (e.g. M.3100 [3.3]) and also providing necessary material, particularly in the area of object-oriented modelling and the management framework. The rest of the tasks are:

Task 1: Describe the TMN management services

Identify each area of management activity which is to be supported by the TMN as a list of management services. This task is performed in prose (natural language). A process of user requirements capture is necessary for this task to be completed.

Task 2: Select and assign TMN management functions (1991 M.3020) or Describe the TMN management context (1996 M.3020)

This task is concerned with the decomposition of the management service in its constituent components, which will then be used as the starting point of the next task that addresses management information modelling. According to the 1991 M.3020 methodology, management services consist of *service components* in the first, and of *functional components* in the second level of decomposition. Functional components are further decomposed into *functions*, which may be grouped into *function sets*. It should be noted that a number of management functions have been defined in M.3400 [3.5] and will be defined in future M.34xx recommendations.

The 1996 M.3020 methodology defines a *TMN Management Context* as the collection of *roles* (who manages), *resources* (what is managed) and *functions* (how it is managed). Management services consist ultimately of *functions*, which can be grouped into *function sets* and further into *function set groups*. In this task, roles, resources and functions are associated with a TMN service, while their relationships are identified. This evolution downplays the importance of decomposition through service and functional components, going directly to functions and groupings of functions. In the ICM extension/modification of this methodology, the decomposition into a two level structure of service/functional components, before the expansion into functions and relevant groupings, is considered of special importance as described in Section 3.2.

Task 3: Management information modelling

This task uses the generic information model to identify suitable existing managed object classes. Where suitable classes do not exist, either new ones must be specified or extensions to existing ones through inheritance are required. The management functions identified in the previous task, which are the smallest constituent parts of the identified services in task 1, should be supported by these classes.

Task 4: Consolidation

This task is concerned with ensuring that the management functions can be supported by the identified management information model, i.e. the monitoring part of the function can obtain all the information it requires through monitoring operations (Get/Discover, Event Report) and the control part has the necessary influence over the objects through intrusive operations (Set, Action, Create, Delete). The result of this task may require that task 3 is revisited.

Task 5: Define management information schema

This task defines the management information schema that a managed system will present to a managing application or system, i.e. the object classes and the containment hierarchy that will be visible at the agent part of that system. Additionally, object relationship diagrams are needed to further explain any relationships other than containment. The managed system supporting the management interface will be a Network Element (or Q-Adaptor/Mediation Device) as this interface specification methodology addresses interfaces between the TMN and the managed network only. However, the ICM methodology extends M.3020 to address the hierarchical nature of TMN systems and specify interfaces between other TMN components.

Task 6: Determine communication requirements

This task specifies the communication requirements to support the previously identified management interactions, e.g. simple transactions, file transfer, and the performance requirements of communications in terms of throughput, reliability, etc.

Task 7: Prepare documentation for protocol tasks

This final task analyses the results of the previous tasks in order to document the requirements on the management protocol. The last two tasks have already been accomplished, resulting in the identification of CMIS/P for object-oriented transactions and FTAM for bulk data transfer (efficient log retrieval, software management etc.)

The above methodology is only concerned with the interface between the TMN and the managed network because the ITU does not intend to standardise on the way specific TMN systems are built apart from providing architectural guidelines [3.1].

The NMF "Ensemble" methodology [3.13] is very similar but goes further: *managing* as well as *managed* aspects of interfaces are specified. This means that managing functions are also included, together with scenarios that describe the "ensemble in action." Parts of the NMF Ensemble methodology have been adopted in the 1996 version of M.3020.

In ICM however it was desirable to specify interfaces between TMN applications at the various levels of the TMN layered architecture. It is possible to use the M.3020 methodology as a basis for an extended methodology for specifying hierarchical TMN systems and the interfaces between all TMN components. These extensions are useful for TMN designers and have been used for specifying, designing and implementing all of the ICM TMN systems.

3.2 Extensions to the methodology and relationship to ODP viewpoints

The M.3020 methodology as it stands is targeted for interface specification between network elements and the TMN, where the TMN system is considered to be a black box. However TMNs are hierarchical systems covering the element, network, service and business management layers. Furthermore there is likely to exist more than one OS within each layer, covering different functional areas (e.g. configuration, performance) and there may be yet a further decomposition into OSs to aid performance or allow distribution of the TMN system. All of these factors mean that a TMN will usually consist of more than a single OS, and more than a single OS per TMN layer. In practice a TMN is likely to consist of 10s or more OSs.

In fact, ICM views a TMN as a distributed system where the unit of distribution is the managed object cluster visible across a management interface, together with the associated managing functions that operate on peer or subordinate interfaces (e.g. OSs, MDs, QAs). The practice in ICM was to decompose management services into manageable computational units of well defined functionality modelled by OSs, rather than take a monolithic approach with one OS per TMN layer. The later approach is likely to result in an OS of enormous complexity which might necessitate distribution of func-

tionality *within* the OS itself. As already stated, the exact way in which a TMN system is decomposed into OSs is not addressed at all in M.3020, hence the need for an extension to the methodology to allow TMN system designers to plan and design TMN systems.

The extensions are mainly concerned with task 2 of the methodology. Having as a result from task 1 the description of the management service from an enterprise point of view, including the user perspectives, the management context and the relevant actors and roles, a decomposition is attempted into first level constituents of the service, termed *management service components.* A second level of decomposition of those follows into *management functional components*. The names of these are not important, they could be called service components level 1 and level 2. The essence is that this is a top-down functional decomposition into components of management services that are perceived as computational entities interacting to achieve the overall management goal. In fact, levels could be more than two and sometimes a sub-service offered by such a component may be thought as a worthy new service in its own right. The leaf components of that decomposition are likely candidates for the final mapping of the management service into OSs in TMN terms. The relationships between the derived components are defined and documented at this stage.

The next step considers allocating the components to the TMN hierarchical layers. Those components dealing with a single NE should be placed in the NE management layer. Those concerned with the relationships between NEs should be allocated to the network management layer. Those concerned with aspects of the telecommunications services which are visible to the users of the services, or with the administration of customers should be allocated to the service management layer.

Further analysis of the relationships between the components will result in a high level specification of the information flows between the defined components. Based on the information flows and on a set of non functional requirements (quantity of information flow, delay requirements on component interactions, degree of information sharing, capability for distribution, etc.) the functional components may be allocated to the physical building blocks of the TMN. It is possible that two leaf components of the management service are placed in the same physical block through this analysis. As a result of the whole procedure, a decomposition of the service into constituent computational entities is arrived at and these will become physical TMN blocks i.e. OSs.

From this point, the M.3020 methodology is resumed. Management functions and managed objects are specified for *each* interface in the TMN, i.e. the internal Q_3 interfaces. This results in specifications of all of the management interfaces between the TMN physical building blocks supporting the management service in hand.

One of the key aspects of this type of decomposition is the target to standardise individual management service components, in terms of both the offered (agent role) and accessed (manager role) information models. This will allow components of management services to be sold “off the shelf.” By careful decomposition and identification of generic components that offer useful sub-services, it should be possible to reuse such components as much as possible and facilitate the rapid introduction of new services. It should be added that this is *not* the ITU approach for the TMN.

Given the fact that a TMN *is* a distributed system and Open Distributed Processing (ODP) provides a generic framework for specifying and building distributed systems, it

is valuable to examine the relationship of the ODP *viewpoint* methodology with the TMN methodology that was presented. It becomes immediately apparent that the latter is closely related to ODP concepts and viewpoints [3.6]:

- The specification of the management service and the definition of the relationships between the TMN, the TMN users and the managed system is part of the *enterprise view.*
- The decomposition of the management service into management service components and the assignment of management functions is part of the *computational view.*
- The specification of information flows, the object modelling tasks and the definition of the management schema is part of the *information view.*
- The allocation of the TMN functional components to the physical building blocks, the determination of the protocol requirements and the selection of protocols is part of the *engineering view.*

The only aspect of ODP not covered by this methodology is the *technology view.* This is the case because the methodology is only concerned with design and specification at an implementation independent level. The decisions on implementation technology follow the design work (although it is also the case that technological advances may also influence design decisions in the other viewpoints).

Although ODP does not specify the exact order in which the viewpoints should be considered, it is usual to follow a progression of enterprise, computational, information through to engineering which is the same order prescribed by the methodology. Note that the methodology specifically states that there may need to be a number of passes before the specification is completed. This is consistent with the ODP notion of the viewpoints not being sequential in time.

3.3 The ICM approach to TMN system design and implementation

Based on the extended methodology described above, ICM defined the procedures for specifying, designing and implementing TMN systems. Figure 3.2 documents the steps the project took for each of the phases of TMN testbed development. Future developments of the testbed should also follow this approach.

Initially management case studies are defined. The case studies specify at a high level the management services to be provided and an overall view of the interactions between the TMN, the managed network and the TMN users, including the actors involved and their roles. The ICM project developed 4 main case studies [3.10] which are also partly documented in Chapters 5, 6 and 7 of this book. The case studies do not attempt to address any design or implementation details of the TMN. The resulting case study definitions are a key driving force of the project, leading the remainder of the design, development and validation work. Although not shown in Figure 3.2, the case study definitions are also an important input to the design of the experiments on the final system for functional validation and evaluation.

Based on the case study definition, the next step involves analysing the way the TMN interacts with the underlying managed network and the users of the TMN. This

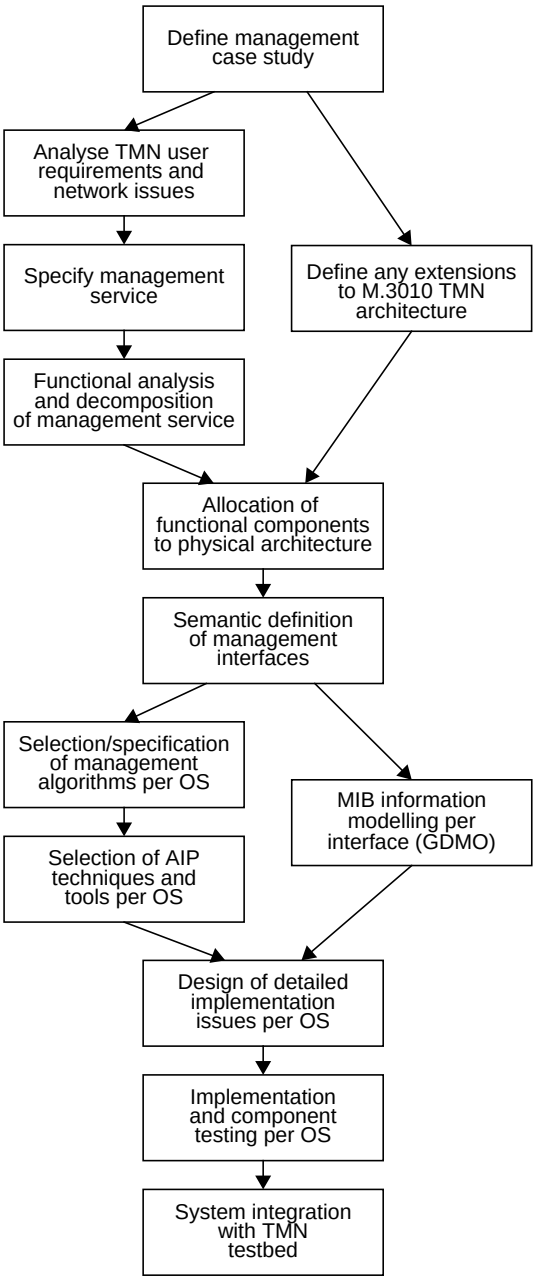

Figure 3.2 The ICM specification, design and implementation steps

involves obtaining a detailed understanding of the managed network and of the aspects to be managed.

The next stage is to define the management service or services involved in providing the functionality described in the case study according to the user and managed network requirements. This corresponds to step 1 of the ITU M.3020 methodology.

The management service is then decomposed into management service components. The relationships between the service components are specified at this stage in order to understand their dependencies on each other and the quantity and complexity of information flows required between them. At this point it may be noticed that a management service component of the management service under examination is similar or equivalent to a management service component derived in previous work (or may have been defined in standards or publicly available work). It may even be discovered that a management service component is equivalent to a different management service in its own right - this is possible because, in general, services are built from collections of services. Right from this point it is possible to identify reusable specifications or even reusable implementations. In the ICM work it was the case that implementations from previous phases could be reused - for example, the whole of the VPCM system (Chapter 5) was reused in the iVPN system (Chapter 6) both at the specification and the implementation levels.

In parallel to the three previous steps, the TMN architecture [3.1] [3.7] [3.8] [3.11] is re-examined in the light of the new case study. This stage is added for completeness as it is usually assumed that the architectural guidelines of M.3010 [3.1] will be adhered to. However, during the ICM project some deviations from M.3010 were made and these are described in Chapter 4.

After the functional decomposition has been completed and the architectural framework has been finalised, a mapping from one to the other is performed. The mapping is done in two stages: first of all the management service components are mapped to TMN function blocks (OSFs, MFs, etc.) and then these are mapped to TMN building blocks (OSs, WS-OSs, etc.). During this process it is likely that additional TMN function blocks are identified which are not strictly part of the management service functionality - for example QAFs to convert proprietary management interfaces on the managed system to standard Q_3 interfaces which may communicate with the management applications located in the TMN. To a large degree the mapping and allocation task is performed informally according to the experience of the designers and a number of guidelines defined by the project. Examples of guidelines highlighting non-functional requirements to be considered in the mapping tasks are:

- Minimise the interactions required between components to minimise the communications overhead in the TMN. (This also has the effect of separating functionality to allow components to be developed by independent teams, although this is not a criterion to be generally applied to the TMN architecture, it is only a pragmatic requirement for ICM).
- Maximise information containment/hiding to allow components to operate as much as possible with internal data.
- Use event reports rather than polling where appropriate.

- Use information models at each layer of the TMN hierarchy according to the information of concern at that layer and provide a mapping between the different views - information encapsulation.
- Avoid replication of information. When information is replicated have a clear hierarchy of responsibility - one master copy.
- Allow for replication of functionality and information for fault tolerance (this partly conflicts with the issue above).
- Allow for fault tolerance - components should continue to operate if other components are unavailable.

Once this step has been completed, the set of TMN building blocks comprising the overall TMN system performing the management service(s) specified by the case study will have been defined. A clear separation of functionality between the components will allow each component to be designed and developed independently in the remainder of the design and development process. An important point here is that the overall functionality of the management service is achieved through the cooperation of the constituent components, and it is important to keep this global view even though the individual components are developed independently by different teams. The ICM project approached this problem by assigning a systems designer who was responsible for the whole system and who worked with each of the component teams to ensure that local decisions made for the design of a component would not interfere in an undesirable way with the whole system.

The next step is to define the interactions between the derived TMN building blocks. This is based on the dependencies and the information flows between the management service components defined earlier. In fact, this step only considers the information flows across the boundaries of the TMN building blocks (where two or more management service components have been allocated to the same OS the information dependencies are an internal issue for the designers of that physical component). At this stage the interactions are defined in more detail, but only at a semantic level - the syntax of the interactions is left until the MIB definition tasks which are carried out in parallel with the design of the management algorithms associated with each component. The set of MIB definitions derived by ICM are not included in this book due to space constraints, but they are documented in full in ICM deliverable 17 [3.12].

In parallel with the information modelling tasks, the management algorithms for each component are selected. At this stage the designers of each component must work closely with the system designer to ensure that the overall system algorithm is met by the cooperation of the individual components. The algorithms may be selected from the literature available on the subject or from existing work performed within the project. In the ICM project it was found that although some work already existed within the areas of the case studies, a substantial amount of theoretical work had to be performed by the project members to either tailor/extend existing algorithms or develop new algorithms from scratch. For example the project was unable to identify existing work specifying algorithms for defining VPC topologies for ATM networks.

The next stage of the design process is to define the techniques and tools to be used for implementing the components. This decision is dependent on the complexity of the management algorithms to be developed and on the maturity of techniques such as

object oriented programming languages and databases, expert systems, neural networks, and so on.

The remaining three tasks are related to implementation rather than design. First of all implementation specific details are designed according to the development techniques and tools selected previously. This is followed by the coding tasks and the associated component testing activities. Finally the whole system is integrated with the existing TMN testbed. The issues related to system integration, validation and experimentation are not dealt with here: ICM deliverable 12 [3.9] describes the ICM approach to this challenging activity, and the system validation and experimentation work is addressed in some detail in Chapter 12.

3.4 References

[3.1] ITU-T Recommendation M.3010, “Principles for a Telecommunications Management Network.”

[3.2] ITU-T Recommendation M.3020, “TMN Interface Specification Methodology.”

[3.3] ITU-T Recommendation M.3100, “Generic Network Information Model.”

[3.4] ITU-T Recommendation M.3200, “TMN Management Services.”

[3.5] ITU-T Recommendation M.3400, “TMN Management Functions.”

[3.6] ITU-T Recommendation X.901, “Open Distributed Processing - Basic Reference Model - Part 1: Overview and Guide to Use.”

[3.7] RACE R2059 ICM Deliverable 2, “Initial TMN Architecture, Functions and Design Approaches,” R2059/CRA/ATG/DS/R/003/b1, Andy Carr, editor, December 1992.

[3.8] RACE R2059 ICM Deliverable 5, “Revised TMN Architecture, Functions and Case Studies,” R2059/ICS/DPG/DS/P/007/b1, David Griffin, editor, September 1993.

[3.9] RACE R2059 ICM Deliverable 12, “Integration and Validation Principles,” R2059/NTUA/EECS/DR/R/P/012/b1, Dionysis Reisis, editor, October 1994.

[3.10] RACE R2059 ICM Deliverable 14, “ICM Case Studies,” R2059/QMW/BM1/DS/P/014/b1, Babul Miah, editor, September 1994.

[3.11] RACE R2059 ICM Deliverable 16, “Updated TMN Architecture and Functions,” R2059/DNAC/DPG/DS/016/b1, Nazim Agoulmine, editor, December 1994.

[3.12] RACE R2059 ICM Deliverable 17, “Specification of Interfaces (Q, X and M),” R2059/DNAC/NAZ/DS/P/017/a1, Nazim Agoulmine, editor, May 1995.

[3.13] NMF Forum 025, “The ‘Ensemble’ Concepts and Format,” August 1992.