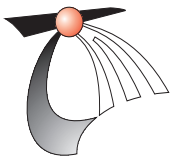
Network Management Architectures

Aiko Pras



CTIT Ph. D-thesis series No. 95-02

P.O. Box 217 - 7500 AE Enschede - The Netherlands
telephone +31-53-893779 / fax +31-53-333815

Centre for
Telematics and
Information
Technology

CIP-DATA KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Pras, Aiko

Network Management Architectures / Aiko Pras
[S.1. : s.n.]. - Ill - (CTIT Ph. D-thesis series, ISSN 1381-3617; no. 95-02)
Thesis University of Twente, Enschede. - With ref.
ISBN 90-365-0728-6
Subject headings: distributed systems; management

NETWORK MANAGEMENT

ARCHITECTURES

PROEFSCHRIFT

ter verkrijging van
de graad van doctor aan de Universiteit Twente,
op gezag van de rector magnificus,
prof.dr. Th.J.A. Popma,
volgens besluit van het College voor Promoties
in het openbaar te verdedigen
op vrijdag 17 februari 1995 te 16:45 uur

door
Aiko Pras

geboren op 30 april 1956
te Zwolle

Dit proefschrift is goedgekeurd door de promotoren

prof. dr. ir. C.A. Vissers

prof. dr. ir. C. Bakker

Abstract

Network management is needed to control and optimize the operation of the network and to respond to changing user requirements. Management includes the initialization, monitoring and modification of the network functions. In order to perform management, special functions are needed. To distinguish these functions from the normal network functions, this thesis introduces the terms 'management functions' and 'primary functions'.
Management functions may be performed explicitly by human operators, but also automatically by dedicated hard- and software modules. In case human operators are responsible for network management, most management functions will be performed from a limited number of remote locations. In case management functions are performed automatically, it is possible to distribute the hard- and software modules that implement these functions over the various systems in the network.

Architectures for network management enable the designers to discuss management functions at a high level of abstraction and guide the design of management protocols and services. In this thesis it is assumed that architectures consist of:

- a set of architectural *concepts,*
- *rules* that tell how to use these concepts,
- *models* that show the application of these rules and concepts to design a specific class of systems.

All current management architectures, notably the ISO, ITU-T (the former CCITT) and the IETF architectures, have been developed after the design of the network functions have been completed. Such approach indicates a specific conceptual view on the role of management functions and invites to apply different architectural concepts for the design of management functions. This thesis proposes an alternative approach, in which no principle distinction is made between the management requirements and the requirements of primary functions. Both sets of requirements can be integrated into one set of requirements and elaborated in a single design process, which uses one architectural model.

The thesis consists of two parts; Part I (Chapter 2 - Chapter 4) analyses the state of the art in network management architectures and Part II (Chapter 5 - Chapter 9) develops an alternative network management architecture.

Chapter 2 analyses the ISO management architecture, which is defined in the 'Management Framework' and the 'Systems Management Overview' standards. As compared to other network management architectures, the ISO architecture received most attention within the research community.

The management architecture of the ITU-T is known as the 'Telecommunications Management Network' (TMN), and is discussed in Chapter 3. The name of this architecture already indicates that this architecture is primarily intended for management of telecommunication (e.g. telephony) networks. TMN in fact consists of multiple smaller architectures:
- a functional architecture
- a physical architecture
- an information architecture, which includes many ideas of ISO management
- a logical layered architecture, which includes a responsibility model.

In 1988 the 'Simple Network Management Protocol' (SNMP) was defined by the IETF to meet the immediate management needs of the Internet. Internet management is analysed in Chapter 4; as opposed to the ISO and ITU-T the IETF did not define a separate architectural standard to describe the concepts behind SNMP. The reason for this is that these concepts resembled the ones that were already described in drafts of the OSI Management Framework and were considered to be obvious.
In 1992 the IETF started the development of a second version of SNMP (SNMPv2). Although the concepts behind SNMPv2 are more difficult to understand and so should be defined in a separate standard, such a definition has not been produced.

The identification of management functions is discussed in Chapter 5. To bring some order in the large number of management functions, special attention is given to the classification of these functions.

Chapter 6 explains how management functions can be designed together with primary functions. It also discusses that it may not always be possible to design all management functions before the start of the operational phase. This is not necessarily a problem, since the management functions that remain can be established during the operational phase by human operators. After the start of the operational phase the designer may decide to add the remaining management functions by developing new generations of network systems.

The alternative management architecture, which integrates primary as well as management functions, is developed in Chapter 7. To demonstrate that both kind of functions can be expressed in the architectural concepts and rules as used by the OSI Reference Model, examples will be given. Several models are developed to explain how management can be performed from one or more remote locations. These models show a number of management protocols as well as special service providers for the exchange of management information. Chapter 9 discusses the management protocols and makes a distinction between two basic types (Variable Oriented and Command Oriented). The service providers to support the exchange of management information are discussed in Chapter 8.

Acknowledgements

This thesis could not be written without the support of others. I am grateful for this support and I would like to thank the following people in particular.

First of all my gratitude goes to my supervisors Kees Bakker and Chris Vissers who gave me the opportunity to do this research; they provided me with many new ideas and detailed comments on the various drafts of this thesis.

Furthermore I would like to thank the members of the TIOS group, especially Marten van Sinderen, for the pleasant and stimulating working environment. I hope this good atmosphere will be retained, despite the gloomy prospects of reorganizations and economy measures.

An enjoyable complement to the more fundamental problems I had to investigate as part of this Ph.D. study, was the applied work I carried out as member of the UT-SNMP group. This work gave me to opportunity to discuss many issues with researchers within the Internet community and provided me with a better understanding of the real problems designers are faced with. I would like to thank Eric van Hengstum and Vincent Berkhout, as well as the many students who participated in the UT-SNMP project.

People often forget the importance of the technical support that is provided by the B&O group. This group keeps our workstations alive and provides us with links to the outside world. I would like to thank the members of this group, in particular Tonnie Tibben.

Thanks also go to DirkJan Speelman, who made the cover design of this thesis.

Finally I want to thank my family and especially my parents for giving me the opportunity to perform my study. Special gratitude goes to my wife Wilma, whose continuous support and understanding I needed to perform this work.

Contents

Part I: Introduction and Analysis of Standardized Management Approaches

1: Introduction

1 Introduction

In the next decade an impressive growth is to be expected in the use of communication networks. To initialize and optimize the operations of these networks, good network management facilities must be developed. The importance of research in this area is confirmed by a number of studies that show the state of current networks. A study in the UK for example showed that LANs go down an average of twenty times a year and subsequently stay out of service for more than four hours [27]. A study in the US showed that every hour of LAN inoperability, 'Fortune 1000' companies loose more than $30,000 [103]. The nine hours breakdown of AT&T's long-distance telephone network in January 1990 even resulted in a $60 million to $75 million loss in AT&T's revenues [30]!

The purpose of this thesis is to improve the understanding of network management and to develop an alternative architecture that avoids the deficiencies of existing management architectures. It is assumed in this thesis that architectures consist of the following elements (Figure 1.1):

- A set of architectural *concepts* or conceptual building blocks. Examples of such concepts are: service provider, service user and Service Access Point (SAP).
- *Rules* that tell how to use these concepts. An example of such rule is that service users must interact with their underlying provider via SAPs.
- *Models* that show how these concepts and rules can be applied to guide the design of a specific class of systems. An example is the OSI Reference Model [43], which was developed to guide the design of computer networks.

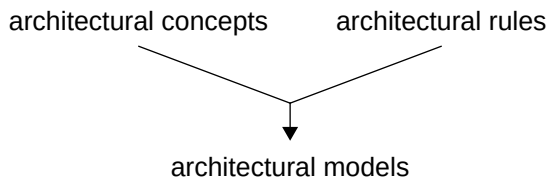


Figure 1.1: Elements of an architecture

1.1 What is management

In literature several definitions of network management exist [18][41][44][80]. Most of these definitions are produced by standardization organizations, which use specific terminology and aim their definitions at specific fields of application. For this reason, these definitions are not suitable for the scope of this thesis. Therefore this section starts with a definition of network management, as considered in this thesis.

> *Definition:* network management is the act of initializing, monitoring and modifying the operation of the primary network functions.

Primary network functions are those functions that directly support the user requirements. They allow for example users to access the network and they take care of the exchange of user data. During the design phase, the primary

functions will be implemented and realized by the designer. How this is done, depends among others upon the skills and experiences of the designer.

Management is needed to bring and keep into operation the network systems that perform the primary functions.

This implies that management should first initialize the various network systems (configuration management). If no errors are made, the network comes into service and the operational phase starts. During this phase, management monitors the various network systems to check if no errors occur. In case of failures, malfunctioning systems will be identified, isolated and repaired (fault management). If systems can not be repaired, they will be replaced by new systems, which must be initialized too. New systems may also be added to allow the connection of new users, to increase performance or to add new functionality. Addition of new systems usually implies reconfiguration. Monitoring the network is also useful to detect changes in the traffic flow. Once such changes are detected, network parameters may be modified to optimize the network's performance (performance management).

To allow management actions to be performed during the operational phase, the designer should define a number of management functions. These functions should be added to the design and implemented and realized together with the primary functions.

An important idea of this thesis is that no *principal* difference exists between the design of primary and the design of management functions. In part II this idea will be elaborated and it will be demonstrated that both kinds of functions can in fact be included in a single architecture.

1.2 Why is management needed

It is interesting to see that the literature usually focuses on '*what* management functions can be identified' and that little has been published with respect to the question '*why* management functions must be performed'. This section discusses this last question and identifies reasons why network management is needed. It reinforces the view that management should not be considered as a set of functions that can immediately be derived from the user requirements.

Cost reduction

Users obviously want the best possible network at the lowest possible price. A way to satisfy this requirement, is to spread the costs of the design over a large number of users. This implies that the design should not be tailored to the specific requirements of a single user group, but be general enough to accommodate the requirements of many potential users (Figure 1.2). The design should thus be a multi purpose design, which means that it should be possible to use mass production techniques.

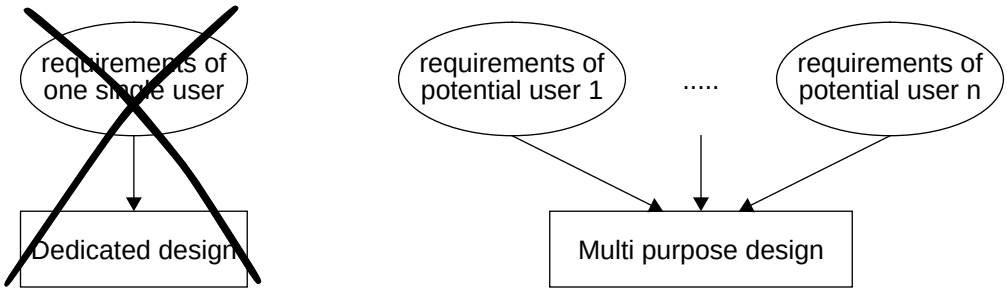


Figure 1.2: Design should not be customized, but general purpose

A way for the designer to deal with the requirements of multiple groups of users, is to abstract from the differences in these requirements and parameterize the design. To allow the network to become operational, the parameters must be initialized to some user specific value. This initialization is the responsibility of management.

> *Example:* Some users want to have a network that spans the entire world, while others want a network that covers a local area. Assume that all users want their networks to be of the packet switched type, and require that every packet will be delivered. To meet this requirement, the designer may decide that after the reception of a packet the receiver should issue an acknowledgement to inform the sender. If the sender does not receive the acknowledgement in time, it will assume that the packet (or the acknowledgement) got lost and the packet will be retransmitted. The time the sender is prepared to wait for the acknowledgement, should be more than the round-trip delay. This delay is much higher in a world-wide network than in a local area network. To produce a multi purpose design, the designer should abstract from this difference and include a management function. This function should arrange that a special *time-out parameter* is set to a high value in case of the world-wide network and a low value in case of the local area network.

Lack of experience

There are rapid developments in the area of networks. In a short period of time both the capabilities of networks as well as their use have increased considerably. As a result the designer will be faced with a number of problems. Because the designer's experience is limited, it is unrealistic to expect that it will always be possible to find good solutions for each and every problem during the design phase. For some problems it may therefore be a good idea to postpone the search for solutions until the operational phase has started; solving such problems will than be the responsibility of management. The advantage of this approach is that it may be expected that during the operational phase additional experience will be obtained, which helps to solve these problems.

Example: Congestion control is a problem that has not yet been solved in a general way. This is due to the fact that there are many different causes for congestion; each cause requiring its own measures. In this example three possible causes will be discussed, including the measures that must be taken to solve each of them (Figure 1.3).

① In a TV-show the viewers are invited to call the studio. This may result in an overload of the telephone network, in which case measures must be taken by management. A strategy to follow could be to restrict the number of call attempts to the studio. This could be implemented by telling *all* switches to accept only a small number of call attempts which have the studio as destination. This measure reduces the amount of prospectless signalling information and allows call attempts to other destinations to proceed.

② Assume a traffic jam develops after an accident has occurred on the highway. In such case it is likely that many car drivers decide to use their mobile telephones to call their homes. The processing of all simultaneous call attempts may overload the network's signalling system; without special measures the switch where the mobile calls enter the network may try to process all call attempts and, as a result, none of them may succeed. This is undesirable: it would be better to tell the switch to accept only a limited number of call attempts. As opposed to the previous case, call attempts will be refused irrespective of their destinations and a single or only a small number of switches will be affected.

③ There is carnival in Rio. Many people stay in the city and the telephone network is heavily loaded. To prevent the Rio area from getting overloaded, calls between other cities which are usually routed through Rio, will now be rerouted around Rio.

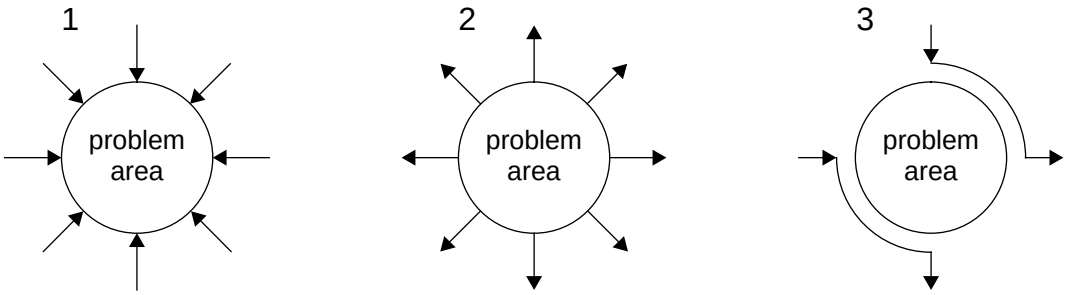


Figure 1.3: Three examples of congestion

These examples showed that it may not always be possible to anticipate all problems during the design phase. Some of the problems should therefore be solved during the operational phase by management. For this purpose the designer should include some general support functions that allow a manager to monitor what is going on in the network, to set alarms, to modify information in remote systems etc.

Fault handling

During a network's operational phase failures can occur suddenly. Failures are situations in which network components (or systems) do not behave in the way that has been specified. As a result of failures, networks may no longer provide the required service and it may even come to a complete breakdown.
The occurrence of failures can be due to ageing and decay of network components (hardware), as well as to human errors (e.g. a dragline that accidentally breaks a cable). The probability that failures occur, depends on the:

- quality of the network components: For a given price, components from certain manufacturers will have lower failure probabilities than components from other manufacturers. Still no manufacturer will be able to built network components that will never fail. No manufacturer will therefore be able to completely satisfy all user requirements.
- way of working: In many cases human errors are the result of unfamiliarity with local circumstances or not following the rules. Although many network failures may be caused by human errors, investigating the origins of these errors is outside the scope of this thesis.

Since it is not possible to prevent all failures and since failures can have severe consequences, the operation of a network should be controlled during the operational phase by management. Such controlling involves the *prediction* of potential failures, the *detection* of existing failures, the *reduction* of the effects of failures and off course their *repair*.
To predict and detect failures, managers should be able to:

- monitor the current behaviour of the network components.
- compare the current behaviour with previous and / or expected behaviour.
- signal exceptional behaviour.

To reduce the effects of failures and to allow reparation, management must have the means to change the state of the network. This may be accomplished by changing network parameters, such as the entries of a forwarding table.

Flexibility

Network designs are commonly described as top-down processes. Characteristic for such processes is the important role of user requirements; the design usually starts with the definition of the user requirements and many design decisions follow from these requirements. The outcome of the design process (the network) is thus primarily determined by the user requirements (Figure 1.4).

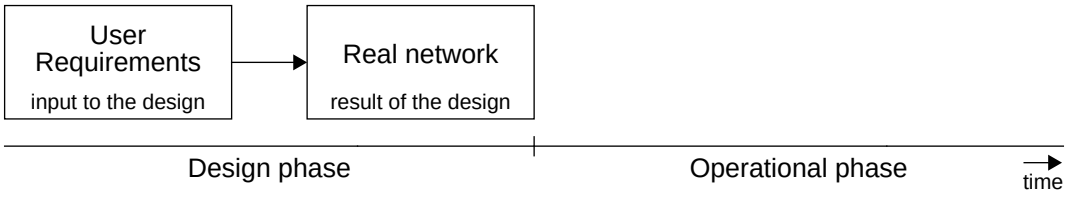


Figure 1.4: Simplified top-down design process

The danger of looking at the design process in this way, is that one may neglect the dynamic nature of the user requirements and consider these requirements as static entities. In reality user requirements change in time and should therefore not be considered as static entities (Figure 1.5).

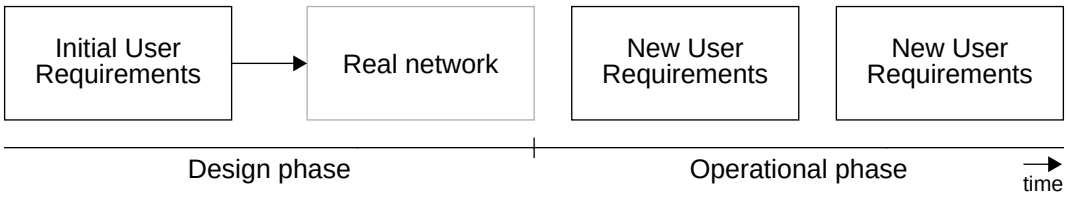


Figure 1.5: Changing user requirements

> *Example:* The user requirements may initially describe that there are only a limited number of users who want to be connected to the network. After the network becomes operational, it may be that others become interested in the network too. As a result, the initial requirements will be changed to accommodate the connection of more users. After some time, it may also be that the users require from the network new kind of services (to support for instance multi-media). Again the user requirements will be changed.

Instead of ordering a new network each time the user requirements change, it is better to built some flexibility into the network. Because of this flexibility the network manager will be able to react during the operational phase upon changes in the user requirements. The designer should anticipate this need and add a number of management functions to the design. Management issues should already be considered during the design phase!

1.3 How is management performed

While designing management functions, the designer will be confronted with a number of design questions. Two of these questions are particularly important because they affect the design process to a considerable extent. These questions are:

- Will management functions be performed by human beings, or will they completely be performed by hard- and software modules?
- Should management functionality be distributed over the entire network, or should it be concentrated as far as possible?

Both questions will be discussed in the following two subsections.

1.3.1 Explicit and implicit management

To denote the case in which human beings are responsible for the initiation of management operations, the term *'explicit management'* will be used. With this form of management, the decision to initiate management functions will explicitly be taken by (human) *operators* during the operational phase.

It should be noted that even this form of management requires the inclusion of a number of management functions during the design. The purpose of these functions is to support the operator while performing his task (see example below).

The opposite of explicit management will be called *'implicit management'*. With this form of management, all management functions will be performed by hard- and software modules; operator intervention is therefore not needed.

> *Example:* On page 3 of this thesis the use of a time-out parameter in a retransmission mechanism was explained. During the design phase, the designer should decide whether this parameter will be set by a human being (explicit management), or by some hard- or software module (implicit management).
> In case the parameter will be set by a human being, the designer should include for example user interface functions that allow the human operator to access this parameter. Such user interface functions may require the introduction of a keyboard and a display.
> In case of implicit management, the designer should include some kind of function that automatically determines the parameter's value. This function may for example measure the average transfer delay and use the result to calculate the best value for the time-out parameter.

An advantage of explicit management is that it is not necessary to elaborate all management functions during the design phase. This is particularly true for the functions that determine at which moment a particular management operation should be initiated and which values should be selected to achieve a specific goal (such functions may be considered as the management 'intelligence' or the management 'decision process').
As a result of this, the design process will be less complicated and requires less time as would be the case with implicit management. Explicit management is particularly useful for solving the unexpected problems that show up during the operational phase and require the invention of novel solutions; explicit management is thus well suited when it comes to fault management.

Since explicit management will be performed by human beings, response time may be poor if compared to implicit management. Other disadvantages of explicit management are its limited capacity and the potential high number of errors.

If we compare the costs associated with both forms of management, we can conclude that in case of explicit management the management functions that are elaborated *during the design phase* will be less complex and therefore *less expensive*. On the other hand, explicit management requires human intervention during the operational phase, thus explicit management will be *more expensive during the operational phase*.

It should be noted that the distinction between both types of management is primarily a matter of realization; in principle it is possible to perform the same kind of functions with both types of management. With the advent of Artificial Intelligence (AI) and expert systems, the distinction between both types diminishes. Real world examples usually show a combination of both forms: some management problems are solved via implicit, while others require the use of explicit management.

1.3.2 Centralized and distributed management

In this thesis the term 'centralized management' is used to denote the case in which management decisions will be taken from a limited number of central locations. The management functionality that takes these decisions is called the *manager*; it represents what can be considered as the management intelligence and is sometimes referred to as the management 'application'.

To manage the operation of the primary functions, *agents* should be added to the systems that perform primary functions. Such agents represent the management support functionality through which manager(s) initialize, monitor and modify the behaviour of the primary functions. As compared to managers, agents are usually simple.

With centralized management, a large number of managed systems can be controlled by a single managing system (Figure 1.6). To allow managers to communicate with their agents, a management information protocol is necessary. Examples of such protocols are CMIP and SNMP (both will be discussed in subsequent chapters).

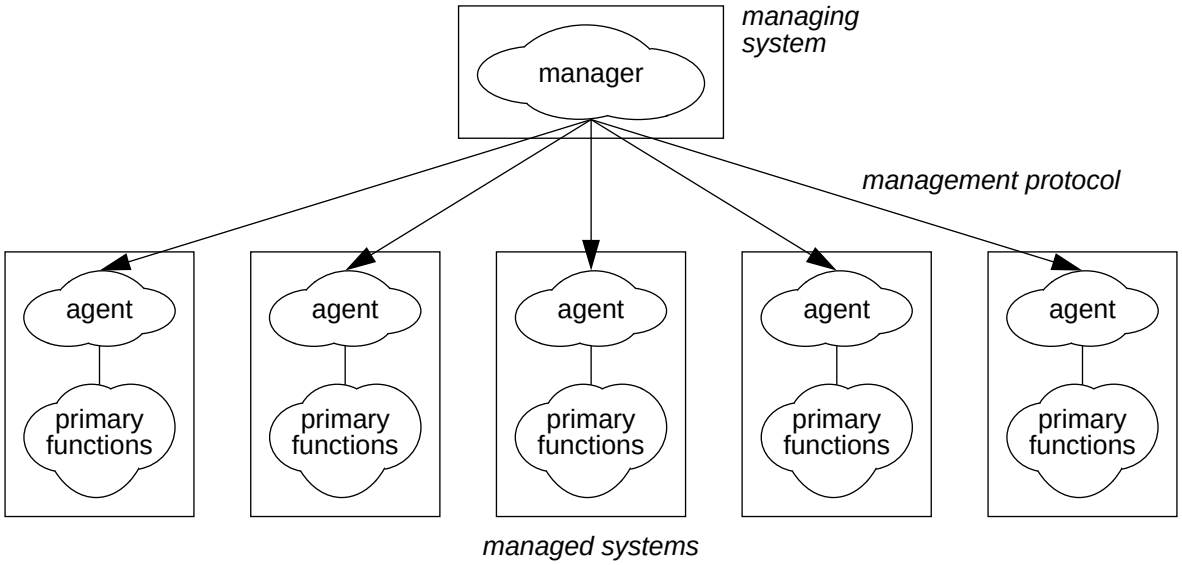


Figure 1.6: Centralized management

Example: forwarding tables are used by the primary functions within the managed systems to determine the path that packets should take to reach their destination. The contents of these tables may be determined by a central manager, who calculates new values periodically or after a change in network topology. For large networks such calculations may be expensive in terms of CPU time and computer memory. After creation of new tables, the manager down loads these tables to the various managed systems.

The term 'distributed management' will be used in this thesis as the opposite of centralized management. With distributed management there are no central systems from which management decisions are taken. Instead, functions that take such decisions will be added to the systems that already perform the primary functions. Such addition will usually be performed on a proportional scale, which means that all systems that perform the same kind of primary functions get equivalent management functions.

Example: with the arrival of powerful and cheap computer components, it has become possible for normal network systems to calculate their own forwarding tables. As a consequence there is no longer a need to bother central managers; management of forwarding tables can be performed in a distributed fashion.
To perform this task, management information must be exchanged between the various network systems. A number of standardization organizations have already defined special routing protocols for this purpose; Figure 1.7 shows some of these protocols.

Number	Title
ISO 9542	ES-IS routing exchange protocol for use in conjunction with ISO 8473
ISO 10589	IS-IS intra-domain routing exchange protocol for use in conjunction with ISO 8473
ISO 10747	IS-IS inter-domain routing exchange protocol for use in conjunction with ISO 8473
ISO 10030	ES-IS routing exchange protocol for use in conjunction with ISO 8878
RFC 1058	Routing Information Protocol (RIP)
RFC 1267	Border Gateway Protocol (BGP)
RFC 1583	Open Shortest Path First (OSPF)

Figure 1.7: Some important routing protocols

Characteristic for distributed management is that each system takes its own management decisions. Because of the potential large number of systems, it will virtually be impossible to let human beings take these decisions. Distributed management must thus be realized in an implicit way.

A disadvantage of distributed management is that it will be difficult to change after the operational phase has started the functionality that makes the man-

agement decisions. This is because such changes require the modification of a large number of network systems, which will be expensive. In case the designer has little experience with a certain management solution, it may therefore be better to use the centralized management approach and concentrate the management functionality that makes the decisions within a single system. The motivation to use centralized management may thus be the same as the motivation to introduce Intelligent Networks (IN).

As opposed to distributed management, centralized management can be realized in an implicit as well as an explicit way. A disadvantage of centralized management is that the entire network may get out of control after failure of a single manager. Compared to distributed management, centralized management may also be less efficient: it is likely that more management information needs to be exchanged and the central managers may become performance bottlenecks.

With some management problems, for instance in case integrity or fairness come into play, it may be better to rely upon centralized management. The determination of system priorities and token holding times, for example, can be better performed by an independent system and not by the systems to which the decisions apply.

1.3.3 Concluding remarks

In this section the following two design questions were discussed:
- should management be performed in an implicit, or an explicit way?
- should management functionality be distributed on a proportional scale over all network systems, or should most management functionality be concentrated within one or more central systems?

Both questions are important, but not specific for the design of management: in principle it is also possible to realize primary functions in an explicit way or to concentrate major parts of the primary functionality within a small number of central systems. The fact that functions are performed in an explicit way or the fact that functions are concentrated within a few number of systems, does not necessarily imply that these functions should be considered as management functions.

> *Example:* An example of a primary function is switching. In the early days of telephony, switching was explicitly performed by human beings. Nowadays switching is implicitly realized by hard and software components.

> *Example:* Controlling the access to a shared medium (e.g. in case of a LAN) may be considered as a primary function. An old form of access control is polling, which is based upon a single master serving many slaves. The slaves are polled by the master to determine if they have data ready for transmission. The slaves are only allowed to start

their actual data transfer after access is granted by the master. Current medium access control mechanisms have abandoned this centralized approach and use distributed approaches instead.

A second remark to be made is that during the design and the operational phase different views of network management may exist. In this thesis the emphasis will be on the design process, and the definition of management as given on page 1 will be applicable. After the operational phase has been entered, it may be difficult however for network users and operators to distinguish between the primary functions and those management functions that are performed in an implicit and distributed way. For this reason several people restrict their view of management to only those functions that are performed from a central location in an explicit way.

1.4 Open questions and contribution of this thesis

During the last decade several organizations recognized the need for network management and developed architectures to guide the design of network management services and protocols. Although these architectures proved their applicability in many cases, they still suffer from a number of problems. In this section some of these problems are identified; the emphasis will be on those problems that will be tackled in Part II of this thesis.

A first problem with current management architectures, is that they are not always properly defined. Some architectures are not even documented, which means that only an intuitive understanding of such architectures can be obtained. Other architectures have been documented, but the definition of their management concepts lacks precision. Finally there are architectures in which the concepts are well defined, but the application of these concepts in the associated management models is done in an inconsistent way.
As a consequence, progression of management standardization is sometimes slow and implementors may not always obtain a sufficient understanding of these standards.

In some management architectures the implicit assumption seems to be, that functions that are being managed can also be used for the transfer of management information. With such architectures, problems may occur in case the functions that are managed cease correct operation. In such cases it is conceivable that the exchange of management information might also be interrupted. As a result, management may no longer be able to reach the functions that should be controlled and it becomes impossible to restore proper operation.

A large number of managed objects have already been defined by the various groups that are active in the area of management standardization. Unfortunately, not all management architectures have paid attention to the classifica-

tion of these objects. In case the intervention by management is required, the manager has to select the appropriate managed object from an unstructured list containing thousands of managed objects. The problem of this is that the manager may not have sufficient experience to determine which object should be selected.

Finally it is remarkable that all standardization organizations consider network management as something special that can be tackled as a separate design process after all primary functions have been developed. Although this approach has certain advantages (e.g. separation of concerns), it also has disadvantages. One major disadvantage is that it will be difficult to comprehend the relationship between primary and management functions; a clear view of *which* primary functions require *which* kind of management functions may not easily be obtained.
If we consider the management products that have been developed thus far, it is apparent that most of these products can be seen as general purpose building blocks that can not immediately be used to solve particular management problems. To obtain practical solutions for real management problems, these general purpose building blocks should be enhanced with 'intelligent' functions that tell how to apply these building blocks in solving actual management problems. To develop such functions, the designer must understand the relationship between primary and management functions. Until now, little progress has been made in the understanding of this relationship and the development of 'intelligent' functions.
A plausible explanation for this problem, which has also been described in literature [101] and discussed on a number of network management mailing lists on the Internet, is that management is investigated in isolation from the primary functions that are the subject of management.

Strategy to solve these problems

Simple measures that solve all of the above mentioned problems are difficult to find (and probably do not exist). This thesis therefore proposes an alternative approach, in which the designer considers the *complete* set of requirements from the outset in an *integrated* way. The principle idea that is put forward in this thesis is that *no difference exists between the design of primary and the design of management functions* (this idea is elaborated in Chapter 5 and Chapter 6 of this thesis). As an implication it should be possible to model primary as well as management functions as part of the same, integrated architecture (Figure 1.8).

How such an integrated architectural model can be developed, is explained in Chapter 7 of this thesis. The advantage of including primary as well as management functions in one single model, is that also the *relationship* between both kind of functions is modelled. The lack of such relationship is one of the reasons why current management products can not directly be used to meet actual management needs (the last problem of the previous subsection). The

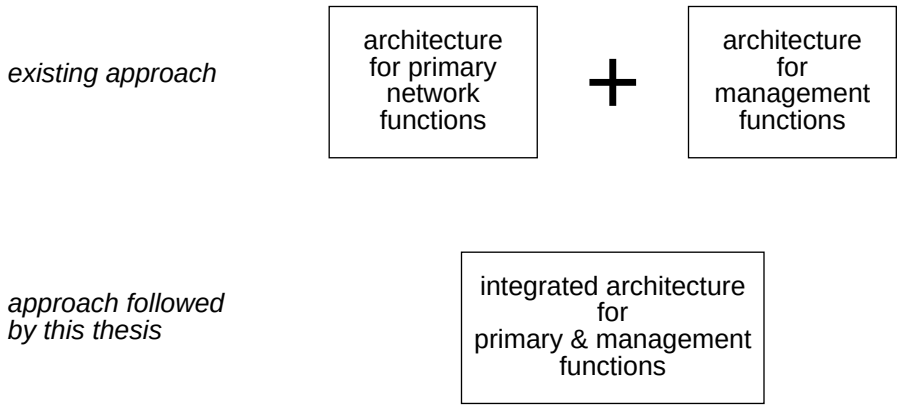


Figure 1.8: Integrated network management architecture

alternative architectural model of Chapter 7 is meant to provide a solution for that problem.

The idea that no difference exists between the design of primary functions and the design of management functions, also implies that it should be possible to model both kind of functions in terms of a single set of architectural *concepts* and *rules*. Instead of developing a management architecture from scratch, one should use the architectural concepts and rules that are already applied for the design of primary functions. To meet the problem that the concepts that are used in current management architectures are not always properly defined (the first problem mentioned in the previous subsection), this thesis proposes to use the architectural concepts and rules of the OSI Reference Model [43]. As compared to others, these rules and concepts have been clearly identified and can be applied in a consistent way [113].

An attempt to use these concepts and rules has been previously made by the members of the OSI management group. As will be explained in Chapter 2 of this thesis, this group was unable to present a consistent architectural model. One of the reasons why this group did not succeed, was the fact that they confused different abstraction levels. This lack of understanding of the various abstraction levels was also one of the reasons why this group suggested to use the managed functions for the exchange of management information (the second problem of the previous subsection).

This thesis demonstrates that it is possible to use in a consistent way the concepts and rules as defined by the OSI Reference Model for modelling management functions. The model that is presented in Chapter 7 can in fact be seen as an extension of the OSI Reference Model [43] or a replacement of the OSI Management Framework [44].

To provide some structure in the large list of managed objects (the third problem of the previous subsection), this thesis proposes to distinguish between three classes of management aspects: service management, protocol manage-

ment and element management. Instead of a monolithic list containing thousands of managed objects, this classification takes care that the manager will be confronted with a limited number of smaller lists.

Contribution of this thesis

The objective of this thesis is to *improve insight and understanding of network management,* and *to present an alternative network management model* (Figure 1.9). This model can be useful to guide the design of network management services and protocols. It should be noted that even though this thesis concludes with a number of general remarks with respect to such management services and protocols, this thesis does not propose any specific new service or protocol. Other issues that will not be addressed are:

- The implementation and realization of individual network systems.
- The definition of new management information models or MIBs.
- The provision of concrete solutions for specific management problems.

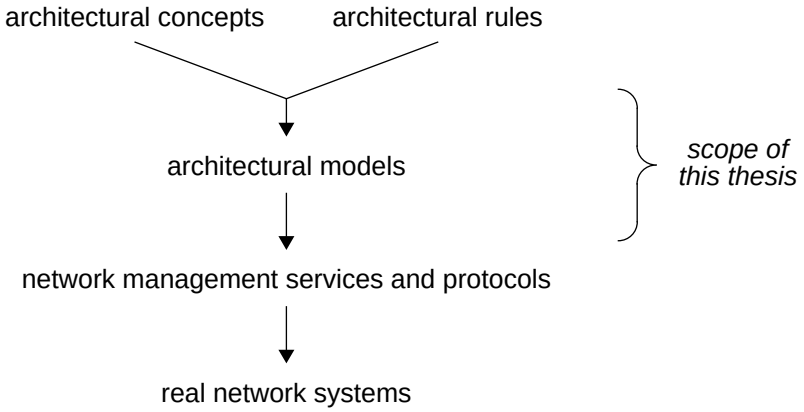


Figure 1.9: Scope of this thesis

1.5 Structure of this thesis

This thesis consists of two parts.

Part 1 discusses the state of the art. It starts to identify the various organizations that have defined network management architectures, and subsequently analyses the three most prominent architectures:

- ISO-OSI's management architecture (Chapter 2).
- The Telecommunications Management Network (TMN), defined by the ITU-T (Chapter 3).
- The Internet network management framework (Chapter 4).

The deficiencies of these architectures are identified and discussed; these deficiencies form the input to part 2 of this thesis (Figure 1.10).

Part 2 discusses an alternative approach to network management. The Chapters 5 and 6 explain how primary as well as management functions might be

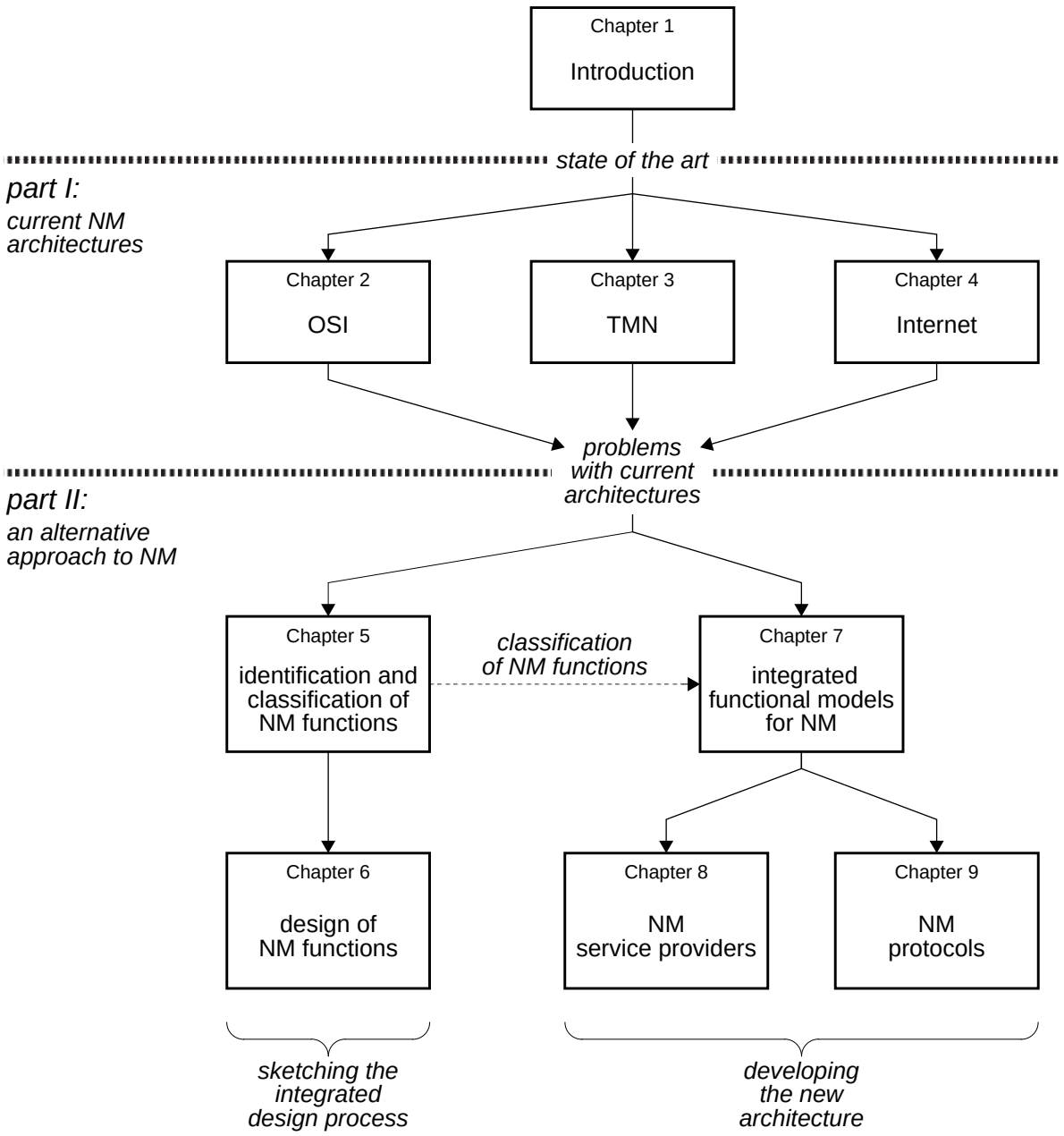


Figure 1.10: Structure of this thesis

tackled as part of a single design process; the Chapters 7 through 9 present the integrated architecture that models the relationship between both kinds of functions.

To identify and classify potential management functions, Chapter 5 discusses the design of primary functions. An important result of this chapter is the proposal to distinguish between three classes of management functions: service management, protocol management and element management.

The purpose of Chapter 6 is to explain when management functions should be developed during the design process. The explanation in this chapter will be based upon the cyclic design model; it will be shown that management functions should preferably be tackled during the later cycles of such design. Since

it may not always be possible to complete the design of all management functions before the start of the operational phase, Chapter 6 proposes to extent the cyclic design model to include the design of future system *generations*.

In Chapter 7 an integrated architecture for primary as well as management functions will be developed. To provide some structure for the various management issues, this chapter uses the classification of management functions as proposed in Chapter 5. This results into three functional models: one for service management, one for protocol management and one for element management. All models show the relationship between primary and management functions and may be useful for the development of management simulators. The models that are defined in Chapter 7 introduce special service providers for the exchange of management information; these service providers will be discussed in Chapter 8.
Chapter 9 further discusses some important characteristics of the management protocols that have been identified in Chapter 7. Two opposite approaches will be presented: a variable oriented approach and a command oriented approach. The object oriented approach, which is used for OSI management, can be considered as a mixture of both approaches and will be discussed too.

1.6 Intended audience

This thesis is intended for:
- Those who want an overview of current network management architectures.
- Those who want an understanding of some of the basic problems with current management architectures.
- Those who are interested in alternative design models for network management.
- Those who want a better understanding of the relationship between primary and network management functions.

In this thesis it is assumed that the reader is familiar with the architectural concepts and rules as defined by the OSI Reference Model.

Part I

Introduction and Analysis of Standardized Management Approaches

There are several organizations who have developed services, protocols and architectures for network management. The three most important organizations are:
- The International Organization for Standardization (ISO).
- The Comité Consultative Internationale de Telegraphique et Telephonique (CCITT); this organization is nowadays called the Telecommunication Standardization Sector (T) of the International Telecommunication Union (ITU).
- The Internet Engineering Task Force (IETF).

Of these three ISO was the first who started, as part of its 'Open Systems Interconnection' (OSI) program, the development of an architecture for network management. The first proposals for such an architecture appeared during the early 1980; nowadays a large number of standards exist for the architecture as well as for network management services and protocols. Of these standards the 'OSI Management Framework', the 'OSI Systems Management Overview' and the 'Common Management Information Protocol' (CMIP) are probably the best known examples. In Chapter 2 of this thesis the OSI management approach will be discussed.

Initially the aim of ISO was to define management standards for datacom networks; development of management standards for telecom networks was left to CCITT. In 1985 CCITT started the development of such management standards; these standards have become known as the 'Telecommunications Management Network' (TMN) recommendations. Originally these recommendations were self standing, but during the 1988-1992 study period they have been rewritten to include the ideas of OSI management. Nowadays OSI management and TMN can be seen as each others complements; Chapter 3 of this thesis discusses TMN.

Looking back at the last decade it may be concluded that the growth of the Internet has played a decisive role in the development of network management protocols. Initially the Internet Architecture Board (IAB) intended to apply the OSI management approach, but at the time the size of the Internet reached a level at which management became indispensable, OSI management groups were still busy with discussing the OSI management framework. Since implementations of OSI management were not expected to appear soon, the IAB requested the IETF (the organization who is responsible for the development of Internet protocols) to define an ad hoc management protocol. This 'Simple Network Management Protocol' (SNMP) was completed within a year and soon many manufacturers started the production of SNMP compliant systems. Although SNMP has several deficiencies, it has become the de facto standard for management of datacom networks. In 1993 an attempt was made to tackle these deficiencies and an improved version of SNMP (SNMPv2) appeared. Chapter 4 of this thesis discusses this Internet management approach.

Next to ISO, CCITT and the IETF also other organizations are worth mentioning for their role in the development of network management. Because of their

comparatively modest role, this thesis will not devote separate chapters to discuss the details of these developments. Instead, a short overview will be given on the next pages.

IEEE

The Institute of Electrical and Electronics Engineers (IEEE) is a professional organization which, amongst others, defines standards for Local and Metropolitan Area Networks (LANs and MANs). These standards are commonly known as the IEEE 802 standards. Some of these standards define how management should be performed in LAN and MAN environments (Figure 1).

Number	Title
IEEE 802.1B	LAN/WAN Management
IEEE 802.1E	System Load Protocol
IEEE 802.1F	Common Definitions and procedures for IEEE 802 Management Information

Figure 1: IEEE Management standards

The IEEE management standards are based upon the ISO CMIP standard. As opposed to ISO, IEEE does not use this protocol at application level (layer 7), but at data link level (layer 2). The name that is used for the IEEE approach, is Common Management Over LLC (CMOL). A problem with this approach is that it is impossible to manage stations located at other sides of routers (routers, by definition, relay via layer 3). IEEE management is thus restricted to single (bridged) LANs or MANs; to manage LANs interconnected by routers, IEEE proposes to use the combination of IEEE and ISO management.

> *Example:* an important advocate of CMOL is IBM. It seems that the restriction that CMOL can not operate over layer 3 routers is acceptable for IBM. This may be because IBM's interconnection strategy is based upon 'source-routing bridges'; usage of layer 3 routers is avoided whenever possible. Since CMOL operates well over source-routing bridges, it is always possible to manage from a central location multiple (IBM) LANs.

Network Management Forum

In 1988 the 'OSI/Network Management Forum' was formed to promote the rapid development, acceptance and implementation of OSI and CCITT management standards [31][32]. The Forum is a non-profit organization whose members are manufacturers, operating companies and research laboratories. After a few years the prefix 'OSI' was removed to indicate that the Forum had widened its scope to reference management standards from other sources.

Examples of such standards are:

- SNMP from the IETF.
- The 'Distributed Management Environment' (DME) [2] from the Open Software Foundation (OSF).
- The 'Management Protocol API' (XMP) and the 'OSI-Abstract Data Manipulation API' (XOM) from X/Open.
- The 'Common Object Request Broker Architecture' (CORBA) from the Open Management Group (OMG).

To organize its work, the NM Forum has defined the OMNI*Point*[1] program. This program comprises "a set of standards, implementation specifications, testing methods plus tools, object libraries that make possible the development of interoperable management systems and applications" [72][74]. The success of the program is somewhat disappointing, presumably because some parts turned out to be more complex than expected (e.g. XOM [25]) and because the delivery schedule could not always be met (e.g. in case of CORBA).

RACE

RACE is a program of the European Community to promote Research and development in Advanced Communications technologies in Europe. The objective of RACE is to introduce community wide Integrated Broadband Communication (IBC) by 1995. To accomplish this goal, the RACE programme includes more than hundred different projects. Many of these projects address management aspects of the IBC. Some projects even invest all of their resources on IBC management (Figure 2).

Number	Name	Description
R1003	GUIDELINE	Advanced Information Processing (AIP) standards for TMN
R1005	NEMESYS	Traffic and Quality of Service (QoS) management for IBCN
R1006	AIM	AIP application to IBCN maintenance
R1009	ADVANCE	Network and customer administration systems for IBCN
R1024	NETMAN	Functional specifications for IBC telecommunications management
R1053	TERRACE	TMN evolution of reference configurations for RACE
R1082	QOSMIC	QoS verification methodology and tools for IBC
R2002	GEMA	General Maintenance Application
R2004	PREPARE	Pre-pilot in advanced resource management
R2021	DESSERT	Decision support system for service management
R2041	PRISM	Pan-European reference configuration for IBC services
R2051	ICM	Integrated communication management

Figure 2: RACE projects on IBC management

1. OMNI*Point* stands for Open Management Interoperability Point

Within RACE, Special Interest Groups (SIGs) and Task Groups (TGs) have been formed to coordinate the results (deliverables) of the different projects. In case multiple projects agree within a TG on some common result, this result can be published as a Common Functional Specification (CFS). Such a specification is often submitted to one of the standardization bodies (usually ETSI).

The RACE programme is dominated by the telecommunications industry and operating companies. It is therefore not surprising to see that research within RACE is based on the work of ETSI and CCITT (TMN in particular). RACE also uses the results of OSI management, because TMN includes pointers to this work. Other standards (e.g. Internet and IEEE) have virtually no impact on RACE.
It is difficult to judge the effect of management CFSs outside RACE. CFSs should not be seen as specifications that can immediately be used by implementers to solve particular management problems. Instead, CFSs can better be considered as collections of ideas that may be useful for standardization organizations such as ETSI and CCITT.

2: OSI Management

2 OSI Management

The purpose of this chapter is to *explain* and *analyse* OSI management. The *explanation* of OSI management may be useful for readers to obtain sufficient background information to understand the remainder of this thesis. The *analysis* will be needed to identify which problems must be solved in Part II of this thesis.

Although the origin of OSI management can be found in ISO, most of the work is performed in collaboration with the ITU-T (the former CCITT). The standards that result from this cooperation are published by both organizations without technical differences. Within the ITU-T, the OSI management recommendations are published as part of the X.700 series.

The first standard that describes OSI management, is the *OSI Reference Model* [43]. This standard identifies OSI management as an important working area and provides initial definitions. Around 1980, a special Working Group (ISO/TC 97/SC 21/WG 4[1]) was formed within ISO to further develop OSI management. The first outcome of this WG was the *OSI Management Framework* [44]. Although the production of this framework took considerable time, it was not generally accepted as an adequate starting point. It was therefore decided to produce an additional standard, which was called the *Systems Management Overview* [53]. Together these standards provide the basis for OSI management (Figure 2.1).

Title	ISO/IEC	ITU-T	Year of publication
OSI Management Framework	7498/4	X.700	1989
OSI Systems management Overview	10040	X.701	1992

Figure 2.1: Basis of OSI Management

This chapter is structured as follows. Section 2.1 and Section 2.2 *explain* OSI management. Reading is recommended for those who do not yet understand this type of management; people who are familiar with it may skip these sections. In Section 2.1 the OSI Management Framework will be discussed; the problem areas that may be solved with OSI management will be identified and the ways in which management information can be exchanged will be discussed. Section 2.2 addresses the OSI Systems Management Overview; it discusses functional, information, communication and organizational aspects of Systems Management.

The *analysis* of OSI management is given in Section 2.3. The purpose of this section is to identify which problems must be resolved in part II of this thesis.

1. Nowadays the group is called ISO-IEC/JTC 1/SC 21/WG 4

2.1 OSI Management Framework

This section discusses the first standard in which OSI management is defined: the OSI Management Framework.
Subsection 2.1.1 describes the problem areas for which OSI management was developed. These areas are the so-called *functional areas* of OSI management. Subsection 2.1.2 discusses the possible ways to exchange management information; these ways are: systems management, layer management and layer operation. Finally Subsection 2.1.3 discusses managed objects, management information and the Management Information Base (MIB).

2.1.1 Functional Areas

The first Working Drafts of the Management Framework already contained sections on management functions. These management functions gradually evolved into what is presently known as the five functional areas of OSI. To denote these areas, the term 'FCAPS' is commonly used (this term is a contraction of the five initial letters of the functional areas).

2.1.1.1 Fault management

Fault management is the set of facilities which enables the detection, isolation and correction of abnormal operation. Possible causes for abnormal operation are: design and implementation errors, overload errors, external disturbances, and lifetime expiration. Fault management includes functions to:
- Maintain and examine error logs.
- Accept and act upon error notifications.
- Trace and identify faults.
- Carry out diagnostic tests.
- Correct faults.

2.1.1.2 Configuration management

Configuration management is the set of facilities which:
- Records the current configuration.
- Records changes in the configuration.
- Identifies network components (give addresses to Service Access Points and titles to network entities).
- Initializes and closes down network systems.
- Changes network parameters (e.g. routing tables).

An important aspect of configuration management, is the assignment of names. To stress this importance, the term *configuration and name management* is sometimes used [110].

2.1.1.3 Accounting management

Accounting management is the set of facilities which enables charges to be established, and costs to be identified for the use of network resources. These resources can be:
- The network service provider, which is responsible for the transfer of user data (e.g. the public network).
- Network applications (e.g. directory services).

Accounting management may:
- Inform users of the costs thus far.
- Inform users of the expected costs in the future.
- Set cost limits (e.g. disable 06 telephone connections).
- Combine costs (to prevent the user from receiving separate bills for each individual connection or, in case of international connections, from each country traversed).

2.1.1.4 Performance management

Performance management is needed to optimize the Quality of Service (QoS). To detect changes in the network's performance, statistical data (e.g. timer and counter values) should be collected and logged on an incidental or periodical basis. The use of such logs is not restricted to performance management; also other management areas take advantage of these logs:
- Performance logs can be used by fault management to detect faults.
- Performance logs can be used by configuration management to decide when changes are needed in the configuration.
- Performance logs can be used by accounting management to adjust bills.

To allow a meaningful comparison of performance logs, also the configuration must be known that existed at the time the logs were made. Configuration information must therefore be logged too.

2.1.1.5 Security management

Security management is the set of facilities which enables the manager to initialize and modify those functions that secure the network from user misbehaviour and unauthorized access. Important parts of security management are key management (for authorization, encryption and authentication), maintenance of firewalls [12][24] and creation of security logs.

2.1.2 Exchange of management information

Three different ways to exchange management information were already identified in the OSI Reference Model: systems management, application management and layer management. Although one would expect that SC 21/WG 4 would use these three approaches as starting point in the development of the

Management Framework, this did not happen. Instead, SC 21/WG 4 decided to remove *application management* and include *layer operation*.

2.1.2.1 Systems management

The initial definition of systems management, as found in the OSI Reference Model, distinguishes between two different properties:
- Systems management is related to the management of OSI resources and their status across all layers of the OSI architecture.
- Protocols for systems management reside in the application layer.

The first property explains *what* is being managed, the second explains *how* management information should be exchanged.

It is interesting to see that the OSI Management Framework focuses on the *information exchange* aspect of systems management (and ignores the aspect of *what* is being managed). Systems management can thus be characterized by the fact that application protocols should be used for the exchange of management information. Application protocols are built upon reliable, connection-oriented underlying services (the term *'royal route'* has sometimes been used to characterize this way of management information exchange [61]).

The decision to use application layer protocols is based upon the assumption that management information should be exchanged in the same way as all other forms of information. According to this view, management should be regarded as just another application on top of the network[1].
To model the exchange of management information, the concept of Systems Management Application Entities (SMAEs) was introduced. SMAEs reside in the application layer and realize the communication aspects of the systems management functions (Figure 2.2).

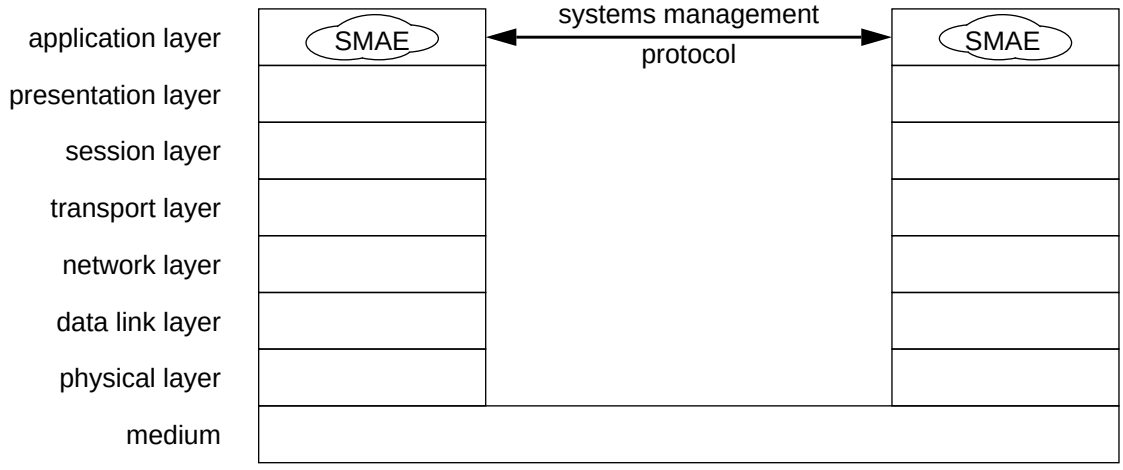


Figure 2.2: Systems management should be seen as an application protocol

1. The OSI Management Framework includes the following text: "it is perceived that the majority of management information exchanges will require context negotiation, the establishment of a management session, a reliable end-to-end transport service etc., in exactly the same way as other application layer exchanges".

The defenders of management exchanges at application level use the following arguments:

- Application layer protocols are the most 'powerful' kind of protocols. One *single* application layer protocol will be capable to transfer many types of management information. Defining *one* powerful management protocol will be much better than defining *many* futile management protocols.
- Services that are provided by lower layers are usually not good enough to satisfy all management needs[1]. To exchange for example large routing tables, the full capabilities of all OSI layers may be required (e.g. error detection, error correction, segmentation, reassembly, context negotiation etc.).
- Management is seen as an application on top of a network. If ISO would model this application not within the application layer, it would undermine its own approach.

The opponents of management exchanges at application level use the following arguments:

- Implementing all seven layers of the Reference Model is expensive. There are many systems that, for their normal operation, do not need to implement all seven layers (e.g. bridges and routers). In these systems it may be a waste of money to implement the remaining layers, just to allow management.
- After a network collapse, an important management responsibility is to restore network services. As a result of the collapse, application layer protocols may no longer function well. In case the exchange of management information relies upon the correct operation of these protocols, management functions may no longer be reachable.
- Application layer protocols involve a lot of processing and are relatively slow.
- Application layer protocols do not have multicast or broadcast facilities.

2.1.2.2 Layer management

While systems management has been defined as the *preferred* way to exchange management information, it is not the only way. The OSI Management Framework allows as alternative for example *layer management*, which has the following properties:

- (N)-layer management supports the monitoring, control and coordination of (N)-layer managed objects.
- (N)-layer management protocols are supported by protocols of the layers (N-1) and below.

The first item relates layer management to *what* is being managed, the second tells us *how* (N)-layer management information should be *exchanged*. Figure 2.3 shows the example of OSI network layer management information, which is exchanged by means of a special purpose layer management protocol located on top of normal communication protocols (a similar figure can be found in the annex to the OSI Management Framework).

1. It is interesting to remember that IEEE's CMOL defines that CMIP (which is a systems management protocol) should be run on top of LLC (page 19).

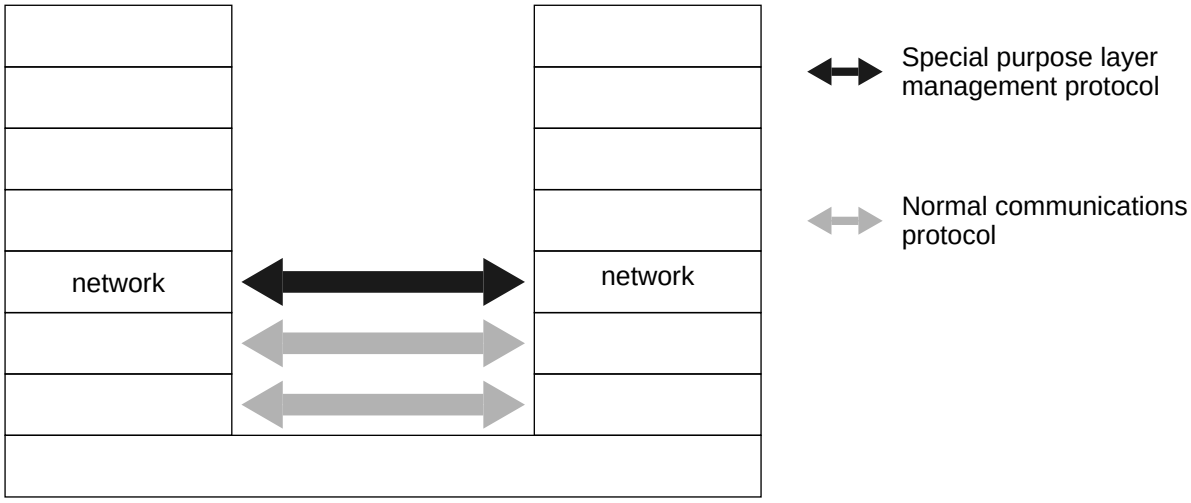


Figure 2.3: Layer management versus normal communication protocols

An important distinction between systems management and layer management, is that systems management uses the *presentation service* for the exchange of management information, whereas (N)-layer management uses the *(N-1)-service*. According to the Management Framework, "usage of layer management is restricted to those cases where usage of systems management is inappropriate".

> *Example:* Layer management is commonly used for the exchange of routing information. In a number of cases, routing information must be broadcasted over an entire routing domain. Since the presentation service has no broadcast capabilities, it may be inefficient to use systems management. Several existing routing strategies therefore rely upon layer management protocols (Figure 1.7).

Other examples of layer management are given in Figure 2.4. The standards which are mentioned in this figure are implemented in many networks that support the OSI ConnectionLess Network Protocol (CLNP) [46]. The figure is included to demonstrate that, contrary to what is sometimes suggested, in real networks layer management exchanges occur frequently.

PDU type	Defined by	When generated
Bridge PDUs	ISO 10038	Generated by all bridges after expiration of Hello timer (default value: 2 seconds)
Configuration PDUs	ISO 9542	Generated by all network entities after expiration of Configuration timer (min. value: several seconds; max. value: several minutes)
Hello PDUs	ISO 10589	Generated by all routers after expiration of Hello timer (default value: 10 seconds)

Figure 2.4: Examples of layer management exchanges

2.1.2.3 Layer operation

The last type of management information exchange is *layer operation*. This form was first defined by the Management Framework and has not been mentioned in the OSI Reference Model. Layer operation is defined as "monitoring and controlling a *single instance of communication*[1]". In case of layer operation, management information is carried as part of a normal layer protocol. Just as with (N)-layer management, (N)-layer operation uses the underlying (N-1)-protocols for the exchange of management information (Figure 2.5).

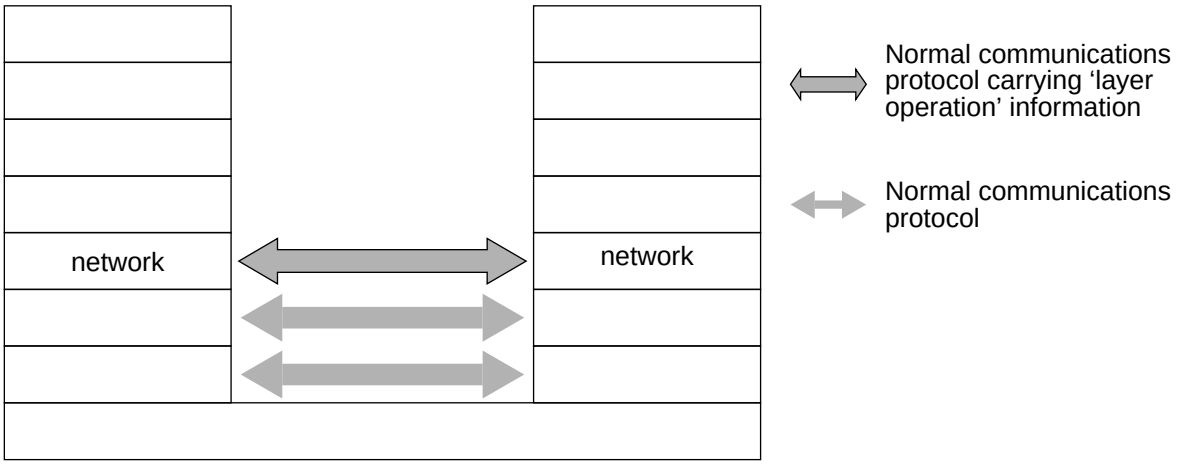


Figure 2.5: Layer operation versus normal communication protocols

2.1.3 Managed objects, management information and the MIB.

To understand the relationship between managed objects, management information and the 'Management Information Base' (MIB), it may be helpful to take a look at the development of the Management Framework standard. Several draft versions of this standard contained the following definitions:

- *Managed object:* "those data processing and data communications resources (whether OSI resources or not) that may be managed through the use of an OSI management protocol".
- *Management information:* "Information *associated* with a managed object that is operated on by the OSI Management protocol to control and monitor that object".

These definitions suggest that a difference exists between managed objects and management information. Although the various drafts of the Management Framework document are sometimes difficult to understand, also the following view emerges:

- managed objects reside in the various layers of the OSI RM,
- management information resides in the *Management Information Base* (MIB).

1. A single instance of communication is a single connection (in case of a connection oriented service) or a single request-response pair (in case of a connectionless service).

The MIB can be seen as a kind of database. The contents of this database is not the set of managed objects themselves, but the information that is *associated* with the managed objects. Layer Managers (LMs) are responsible to maintain the association between MIB information and managed objects (Figure 2.6). In case of problems with Layer Managers, it might occur that the information in the MIB does not accurately reflect the state of the managed objects any more.

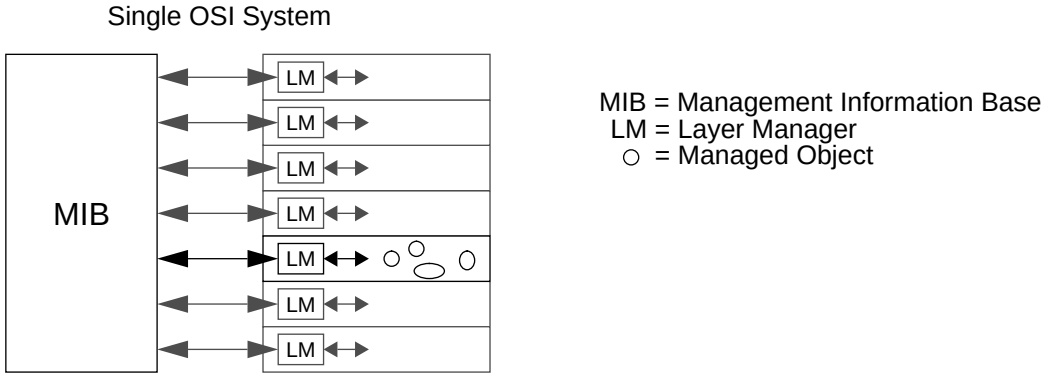


Figure 2.6: Early view of MIB, managed objects and Layer Managers

This view of managed objects, management information and the MIB was still expressed in the DIS version of the management framework (1988). In fact this view is still supported by many people[1]. For unclear reasons (none of the national bodies made an explicit request) the editing meeting that was responsible for resolution of the comments on the DIS ballot:

- Removed the definition of management information.
- Changed the definition of managed objects.
- Changed the description of the MIB.
- Removed an explanatory picture from the Annex.

As a result of these changes, there exists no longer a difference between the management information that can be stored within a MIB, and the managed objects themselves. According to the final version of the Management Framework "the set of managed objects within a system, constitutes that system's MIB". Since this text implies that a MIB is conceptually nothing more than the collection of all managed objects within that system, the MIB concept does not seem to be very useful any more [115].

2.1.4 Impact of the OSI Management Framework

The OSI Management Framework is the first in a series of OSI management standards. It would therefore be reasonable to expect that this standard contains important information and is generally accepted. In this subsection both of these assumptions will be analysed.

1. This view is for instance still being used by the Internet management (SNMP) group.

Information

Although its difficult to determine the *quality* of the information that is in the OSI Management Framework, it is well possible to examine its *quantity*. It is, for instance, interesting to look at the development of the text on *systems management*[1], which is the most important form of OSI management.

Systems management was first defined by the OSI Reference Model, which included about 150 words to explain the idea.

The first working draft of the OSI Management Framework appeared in June 1981. This draft contained about 200 words and several pictures to explain systems management. In subsequent working drafts, new text was being added and existing text was being modified. In the final working draft (number 7, November 1985), about 1600 words were used to explain systems management. Besides, several pictures were included.

Unfortunately, the working drafts were not very consistent and contained several ambiguities. During the various ballot stages[2], SC 21/WG 4 was unable to resolve these ambiguities. As a result, there was no alternative than the removal of controversial parts, including all pictures. In the final text of the OSI Management Framework, the explanation of systems management has been reduced to something less than a single page (200 words).

Although production of the OSI Management Framework took eight years, the final text contains the same number of words on systems management as the first working draft...

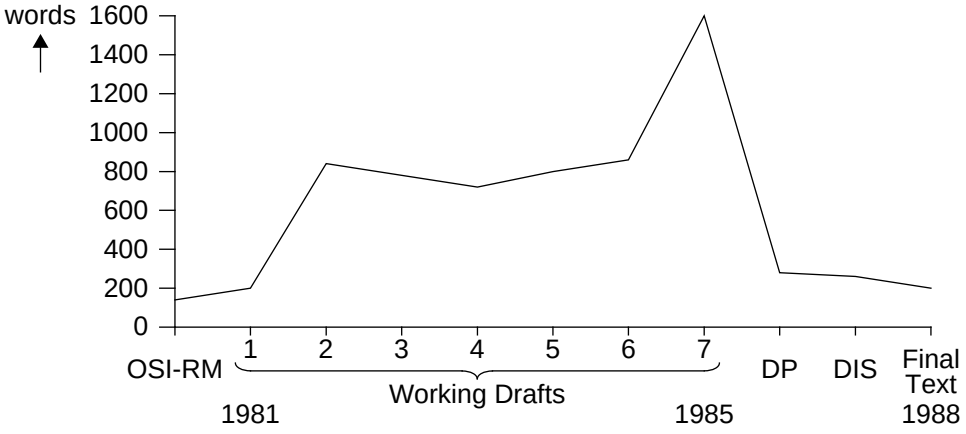


Figure 2.7: Length of Systems Management text (in words)

Level of acceptance

The fact that major pieces of text had to be removed during the various ballot stages should not be a problem, provided that the remaining text was generally accepted. This is barely the case however, as becomes clear from the following examples:

1. In this subsection the standards's informative (non-integral) annexes will not be taken into account.
2. DP: September 1986; DIS: January 1988; acceptance of final text: December 1988; date of publication: November 1989

- The last opportunity for ISO national bodies to judge the Management Framework, was during the ballot on the DIS. National bodies had to vote on an incomplete document however. The section on managed objects contained for example just a single sentence, plus an editors note saying that further detail may be required. The (non-integral) Annex contained multiple 'to be provided' statements.
- The Summary of Voting [62] on the DIS version of the Management Framework showed 13 approvals, 2 abstentions and 2 negative votes. At first sight, this result suggests a reasonable level of acceptance. However, a number of member bodies had severe reservations. Some of these are shown below:
  - "AFNOR is aware that technical architectural material is still missing ..."
  - "This DIS is in no way wrong or misleading. It is, however, according to our opinion, completely insufficient ..." (DIN)
  - "NNI has the strong feeling that the current DIS does not contain those concepts that should be part of the management framework"
  - "The UK disapproves DIS 7498-4, because major revisions are required to remove general inconsistencies ..."
- The editing meeting held to resolve the comments on the DIS version of the Management Framework, was attended by only six people from four countries (previous meetings showed a much better participation). Still it was decided to make major technical changes to the document (see for instance page 30). Despite these changes, the editing meeting did not find it necessary to hold a second DIS ballot.

The sequel

Even the final versions of the Management Framework document could not get substantial technical support. The fact that the document was eventually accepted, should therefore be understood from the following perspectives:

- Most people did not want to spend more time on the document.
- For political reasons it would be unwise not to go to IS (International Standard) status. The alternative would be the much lower TR (Technical Report) status.
- Work had recently started on the OSI Systems Management Overview document. Outstanding issues could be discussed during the progression of this document.

2.2 OSI Systems Management Overview

The definition of the OSI Systems Management Overview (SMO) started around 1987. In June 1991 the final SMO text was ready and the document was submitted for registration as IS. Compared to the OSI Management Framework, the SMO contains much more information and is far better accepted.

The SMO includes a further description of systems management. This description distinguishes between the following aspects:

- information
- organizational
- functional
- communication

The following four subsections discuss each of these aspects. The scope of these discussions is not restricted to the SMO document; each subsection also includes references to derived ISO/ITU-T standards and parts of these derived standards will also be explained.

2.2.1 Information aspects

The information aspects of the systems management model deal with the resources that are being managed. These resources are viewed as 'managed objects'.

The concept of managed objects was introduced as part of the OSI's Management Framework. Initially this introduction was considered to be sufficient; the concept of managed objects was not further elaborated because it was thought obvious and in violation with the OSI principle that stated that only external behaviour of systems may be standardized [112]. As time went on, it appeared that different people interpreted the managed object concept in different ways: the initial assumption that the concept was obvious, turned out to be wrong! After SC 21/WG 4 realized this problem, it decided to refine the description of managed objects as follows:

> "A managed object is the OSI Management view of a resource that is subject to management, such as a layer entity, a connection or an item of physical communications equipment. Thus, a managed object is the abstraction of such a resource that represents its properties as seen by (and for the purpose of) management. An essential part of the definition of a managed object is the relationship between these properties and the operational behaviour of the resource. This relationship is not modelled in a general way."

An interesting part of this description is the last sentence, which states that the relationship between operational behaviour and management properties is not modelled in a general way. Without such a relationship, it is not possible however to express the effect of management operations upon the managed resources. This is clearly undesirable. An important difference between the OSI management approach and the management approach that is presented in part II of this thesis, is that the latter does in fact model such a relationship.

According to OSI's Management Information Model [54], the management view of a managed object is visible at the *managed object boundary*. At this boundary, the management view is described in terms of (Figure 2.8):

- Attributes, which are the properties or characteristics of the object.
- Operations, which are performed upon the object.
- Behaviour, which is exhibited in response to operations.
- Notifications, which are emitted by the object.

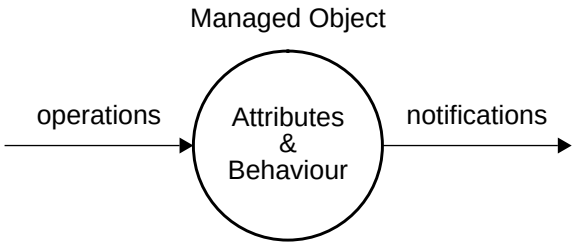


Figure 2.8: A managed object

Next to the managed objects that represent resources, there are also *'management support objects'*. Such objects may be introduced by the designer of management functions during the implementation phase. An example of a management support object is a 'log record', which may be used to store management information.

The managed object concept is refined in a number of additional standards, which are called the Structure of Management Information (SMI) standards (the first six entries of Figure 2.9). The SMI standards do not specify the actual managed objects; managed objects are defined by the working groups responsible for the various layers of the OSI Reference Model (examples of such standards are given in the last four entries of Figure 2.9).

Title	ISO/IEC	ITU-T
Management Information Model	10165-1	X.720
Definition of Management Information	10165-2	X.721
Guidelines for the definition of Managed Objects	10165-4	X.722
Generic Management Information	10165-5	X.723
Guidelines for Conformance Proformas	10165-6	X.724
General Relationship Model	10165-7	X.725
Management Information related to the transport layer	10737	X.284
Management Information related to the network layer	10733	X.283
Management Information related to the data link layer	10742	X.282
Management Information related to the physical layer	13642	X.281

Figure 2.9: Standards for managed objects

2.2.2 Organisational aspects

OSI systems management is organized in a centralized fashion (Subsection 1.3.2). According to this scheme, a single manager may control several agents. The manager performs operations upon (the managed objects within) the agents, agents forward notifications to their managers. Figure 2.10 illustrates this *manager-agent* concept.

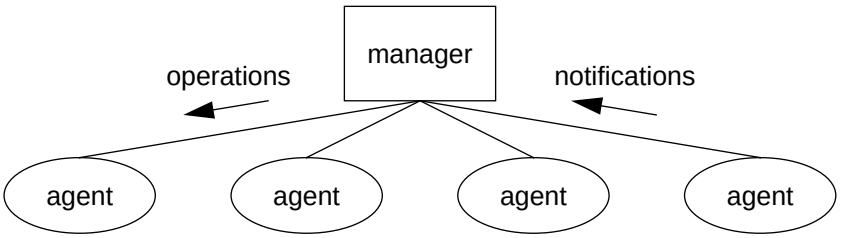


Figure 2.10: Manager-agent concept

The OSI management environment may be partitioned into a number of *management domains*. The partitioning can be based on functional requirements (e.g. security, accounting and fault management), but also on other requirements (e.g. geographical and technological). The idea of management domains is still under development by ISO.

2.2.3 Functional aspects

Soon after the first working drafts of the Management Framework appeared, ISO started to define protocol standards for each of the five functional areas. After some time an interesting observation was made: *most of the functional area protocols used a similar set of elementary management functions*. At the Sydney meeting of SC 21/WG 4 (December 1988), it was therefore decided to stop further progression of the five functional area protocols [66] and concentrate on the definition of *elementary management functions* (Figure 2.11).

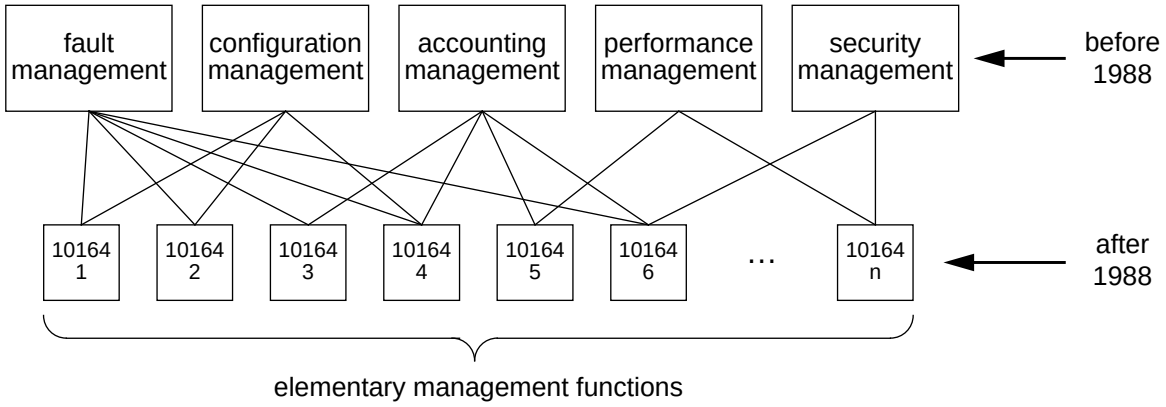


Figure 2.11: Functional areas and elementary management functions

The elementary functions, which are defined at a much lower abstraction level than the original functional areas, are called 'Systems Management Functions' (SMF). Figure 2.12 gives a list of these functions; the functions that have no associated ISO/IEC sequence number are still Working Drafts.

Title	ISO/IEC	ITU-T
Object Management Function	10164-1	X.730
State Management Function	10164-2	X.731
Attributes for representing Relationships	10164-3	X.732
Alarm Reporting Function	10164-4	X.733
Event Report Management Function	10164-5	X.734
Log Control Function	10164-6	X.735
Security Alarm Reporting Function	10164-7	X.736
Security Audit Trail Function	10164-8	X.740
Objects and Attributes for Access Control	10164-9	X.741
Accounting Meter Function	10164-10	X.742
Workload Monitoring Function	10164-11	X.739
Test Management Function	10164-12	X.745
Measurement Summarization Function	10164-13	X.738
Confidence and Diagnostic Test Classes	10164-14	X.737
Scheduling Function	10164-15	X.746
Management Knowledge Management Function	10164-16	X.750
Time Management Function		X.743
Software Management Function		X.744
General Relationship Model		X.747
Response Time Monitoring Function		X.748
Management Domain Management Function		X.749
Changeover Function		X.751
Enhanced Event Control Function		X.752

Figure 2.12: Systems Management Functions

It is outside the scope of this thesis to give an in-depth discussion of all Systems Management Functions. The interested reader is referred to [4][110] and [115].

2.2.4 Communications aspects

OSI has defined the 'Common Management Information Service' (CMIS - [50]) as the preferred service for the exchange of management information (although the use of other exchange services is still allowed, such as services provided by TP and FTAM). CMIS' role is restricted to the transfer of management information; actual control of systems is left to the MIS-users which are located on top of CMIS (Figure 2.13).

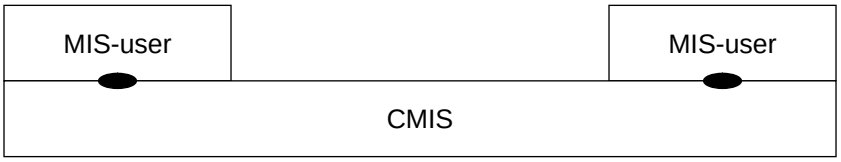


Figure 2.13: MIS-users on top of CMIS

The CMIS service provider may be decomposed, in which case two or more *Systems Management Application Entities* (SMAEs) appear. These entities contain a number of Application Service Elements (ASEs[1]) and use the presentation service provider to transfer their data (Figure 2.14). The interaction between SMAEs is defined by the 'Common Management Information Protocol' (CMIP - [51]).

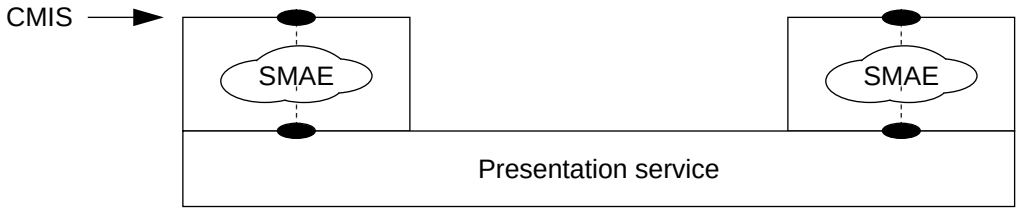


Figure 2.14: Decomposition of CMIS

The CMIS standard defines the following service primitives:

- *M-GET:* to retrieve management information. It can for example be used by a manager to retrieve an agent's network address.
- *M-CANCEL-GET:* to cancel a previously invoked *M-GET*. It is helpful in those cases where the *M-GET* delivers too much information or consumes too many resources. This can happen if, for example, a manager requests an agent to present its entire routing table.
- *M-SET:* to modify the attributes of a managed object. It can for example be used by a manager to change an agent's network address.
- *M-ACTION:* to perform some action on a managed object. It can for example be used by a manager to reboot another network system.
- *M-CREATE:* to create a new instance of a managed object. It can for example be used to add an entry to a routing table.
- *M-DELETE:* to delete an existing managed object instantiation. It is the reverse function of *M-CREATE* and can for example by used to remove an entry from a routing table.

1. See [105] for an explanation of ASEs and application layer structuring.

- *M-EVENT-REPORT:* to report the occurrence of some kind of event. It can for example be invoked by an agent to inform the manager that one of the agent's outgoing links can not be used any more.

The first six primitives define *operations*, the *M-EVENT-REPORT* primitive defines a *notification* (see Figure 2.8). While all primitives can be used in a confirmed way, some (*M-SET*, *M-ACTION* and *M-EVENT-REPORT*) may also be used in an unconfirmed way.

Figure 2.15 lists the ISO / ITU-T standards that define how systems management information should be exchanged; the list does not include the amendments and additions to these standards.

Title	ISO/IEC	ITU-T
Common Management Information Service	9595	X.710
Common Management Information Protocol (CMIP)	9596	X.711

Figure 2.15: Standards for communication aspects

2.3 Analysis

This section discusses some of the main problems of OSI management. It is not the intention to be exhaustive; the focus will be on problems that will somehow be tackled in the alternative management architecture that is presented in Part II of this thesis.

2.3.1 Architectural integrity

An important problem of OSI's management architecture, is that it does not apply the modelling principles of the OSI Reference Model in a proper way. OSI management violates for example the layering principle, which says that users in a particular layer need not know the internal structure of their underlying service provider. According to the layering principle, entities can only interact with entities in adjacent layers via service primitives; it is not possible that entities randomly access components in arbitrary layers by some other means. Still this is exactly what OSI systems management does, as will be explained below.

Consider two systems: one in a manager and one in an agent role (Figure 2.16). The system that operates in the agent role is the one that is being managed; it contains several managed objects to represent the resources that can be managed. The managed objects can be accessed by a SMAE. This SMAE communicates via a systems management protocol (CMIP) with a SMAE that is located in the manager system.

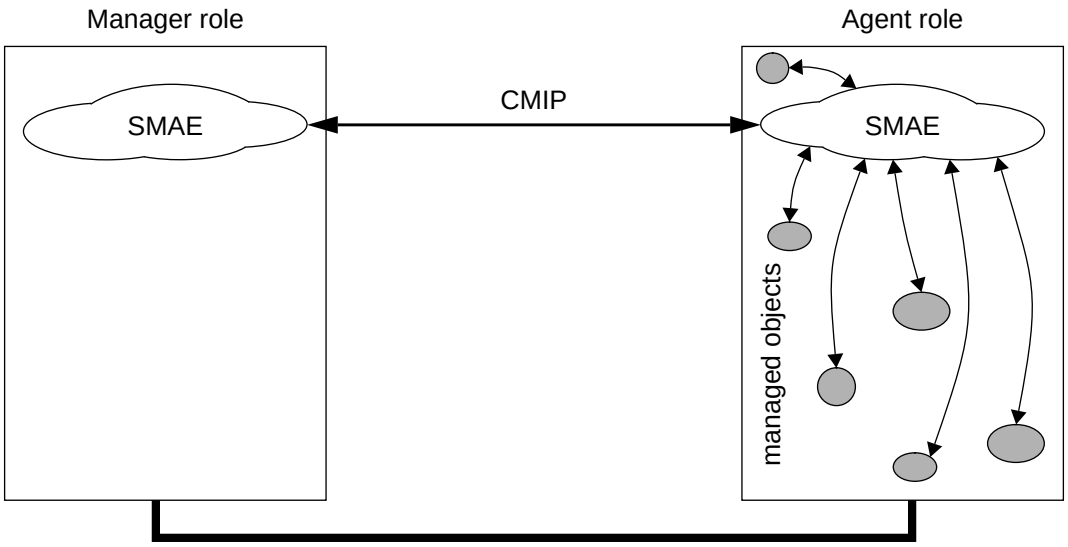


Figure 2.16: OSI systems management

Each layer of the OSI Reference Model may need management. Managed objects can thus be found in all layers of the OSI Reference Model. The SMAE is by definition located in the application layer (Figure 2.17). According to OSI management, the SMAE will be able however to manipulate managed objects, irrespective of the layer in which these objects are located. The implication of this is that the SMAE should have knowledge about the internal structure of the underlying service provider and be able to access components within this provider via some 'magic' interaction mechanism. This is in violation with the modelling principles as defined by the OSI Reference Model.

Figure 2.17: The SMAE knows the structure of the underlying provider

Although several people in the OSI management community are aware of this problem, they have until now not been able to find a solution (in fact the removal of pictures from the management framework drafts can be seen as an attempt to disguise the problem). Part II of this thesis proposes an alternative architecture that resolves this problem.

2.3.2 Problems with fault management

Another weak point of OSI's management approach is that (implicitly) the layer protocols that are being managed, are used for the exchange of management information too. Figure 2.18 will be used to illustrate this relation[1].

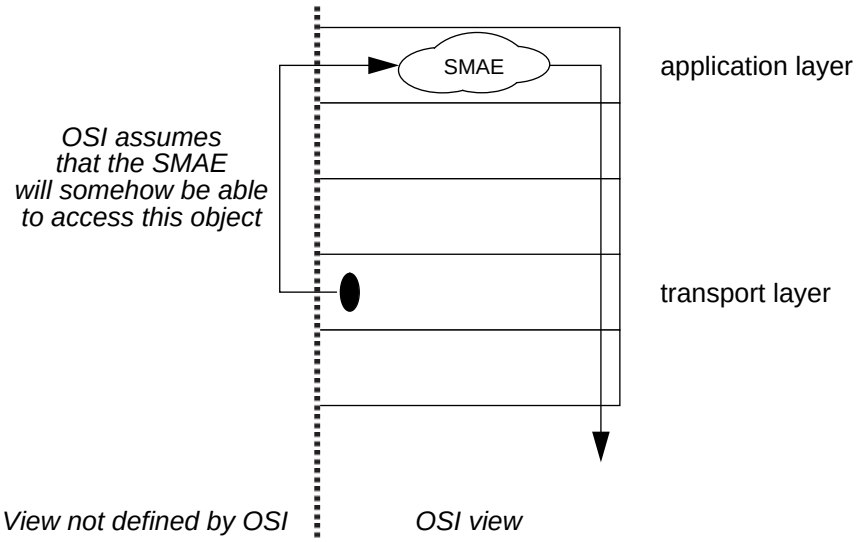


Figure 2.18: Example showing the double role of layer protocols

The figure shows a transport layer managed object, such as a counter that reflects the number of CRC errors. This CRC counter can be read by the SMAE, which resides within the application layer. As explained in the previous subsection, OSI does not describe how the SMAE accesses the transport layer managed object; OSI management assumes however that some form of interaction will be possible. After the SMAE has read the counter, it may decide to send CRC information to other systems. For this purpose, the SMAE presents the information as user data to the underlying presentation service provider. The transport layer protocol is part of this provider, however; the transport layer protocol is thus also used for the exchange of management information.

The protocols that are being managed, will thus be used to exchange management information too. The problem with this dependence is, that fault management may become impossible. Consider for example a system in which the transport entity suddenly breaks. In case all other entities within that system remain operational, the failure may be detected by the SMAE, which may decide to generate an alarm report. This alarm report can not be transmitted however, because of failures within the local transport entity.

2.3.3 Other problems

Besides the two problems that were mentioned in the previous subsections, OSI management is faced with several other problems:

- OSI management explains how individual management operations, such as *GETs* and *SETs*, should be performed. The current management standards do not specify however the sequence in which these operations should be performed to solve specific management problems. Until now, *solutions* for real management problems hardly exist.

1. The merits of Figure 2.18 are, from an architectural point of view, questionable. Variations of it are given however in many publications ([17][33][42][58][59][60][73][109]).

- OSI management is rather complicated. SC 21/WG 4 has introduced several new concepts, which are sometimes difficult to comprehend. Other barriers are the large number of management standards and the size of these standards.
- During the standardization process considerable changes were made in some of the main concepts of OSI management. Examples of such changes are the redefinition of 'managed objects' (see page 30), the removal of 'application management' and the introduction of 'layer operation' (see page 25).
- The standardization of OSI management took too much time. Other approaches, such as SNMP, could therefore emerge.
- Although most manufacturers declared their support for OSI management, only a few offer implementations.
- Management systems that are based on the OSI architecture are presently more expensive than management systems that are based on the Internet management architecture (SNMP).

Due to these problems, it is questionable whether OSI management will reach the dominant market position that has originally been anticipated.

3: TMN Management

3 TMN Management

The term TMN is introduced by the ITU-T (the former CCITT) as an abbreviation for 'Telecommunications Management Network'. The concept of a TMN is defined by Recommendation M.3010 [20].

According to M.3010, "a TMN is conceptually a separate network that interfaces a telecommunications network at several different points". The relationship between a TMN and the telecommunication network that is managed, is shown in Figure 3.1. According to this figure, the interface points between the TMN and the telecommunication network are formed by *Exchanges* and *Transmission systems*. For the purpose of management, these Exchanges and Transmission systems are connected via a *Data Communication Network* to one or more *Operations Systems*. The Operations Systems perform most of the management functions; these functions may be carried out by human operators but also automatically. It is possible that a single management function will be performed by multiple Operations Systems. In this case, the Data Communication Network is used to exchange management information between the Operation Systems. The Data Communication Network is also used to connect *Work Stations*, which allow operators to interpret management information. Work Stations have man-machine interfaces, the definition of such interfaces fall outside the scope of TMN (Work Stations are therefore drawn at the border of the TMN).

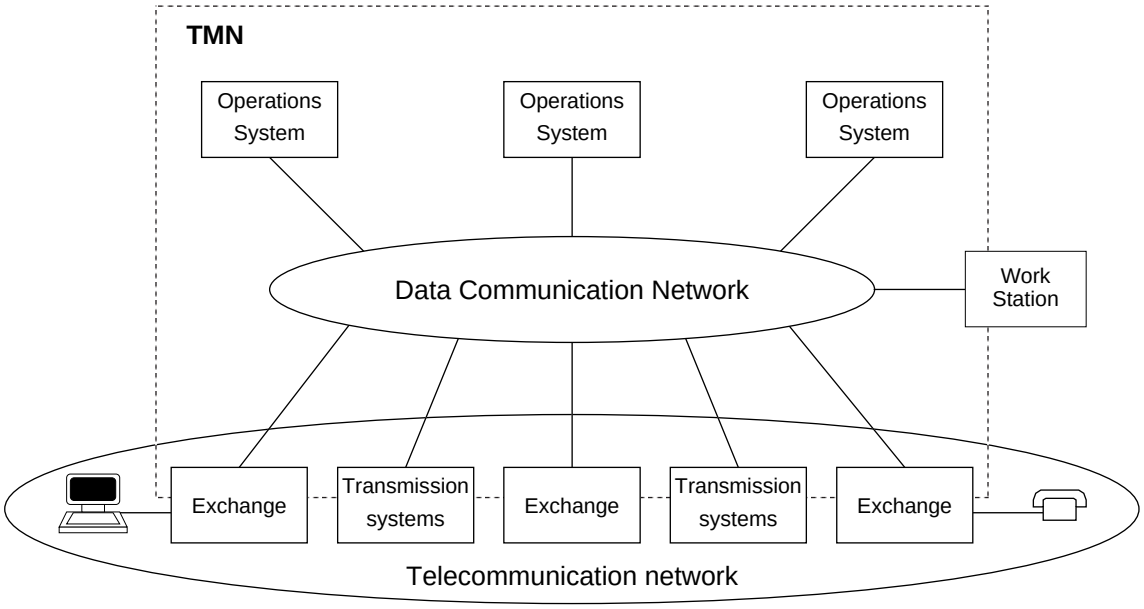


Figure 3.1: General relationship of a TMN to a telecommunication network

In this chapter TMN is introduced and analysed. Section 3.1 identifies which TMN standards exist and how these standards relate to the OSI management standards. The subsequent sections concentrate upon the most important of these standards: M.3010. This standard defines the general TMN management concepts and introduces several management architectures at different levels of abstraction. Section 3.2 discusses TMN's functional architecture, which

describes various management functions. Section 3.3 discusses TMN's physical architecture, which defines how these management functions may be implemented into physical equipment. Section 3.4 discusses one of the best known ideas of TMN: the responsibility model. This model shows how the various management responsibilities can be structured into a convenient arrangement. The analysis is provided in Section 3.5.

3.1 TMN standardization

The TMN standardization was started in 1985 by CCITT Study Group IV [69]. The first TMN recommendation was called M.30 [19] and was published in 1988 as part of the *blue books*. In 1992 a completely revised version appeared and the number of the recommendation was changed into M.3010 [20].

In the 1988-1992 study period, work started on a number of related recommendations (see Figure 3.2). These recommendations define specific aspects of TMN and use M.3010 as the architectural basis.

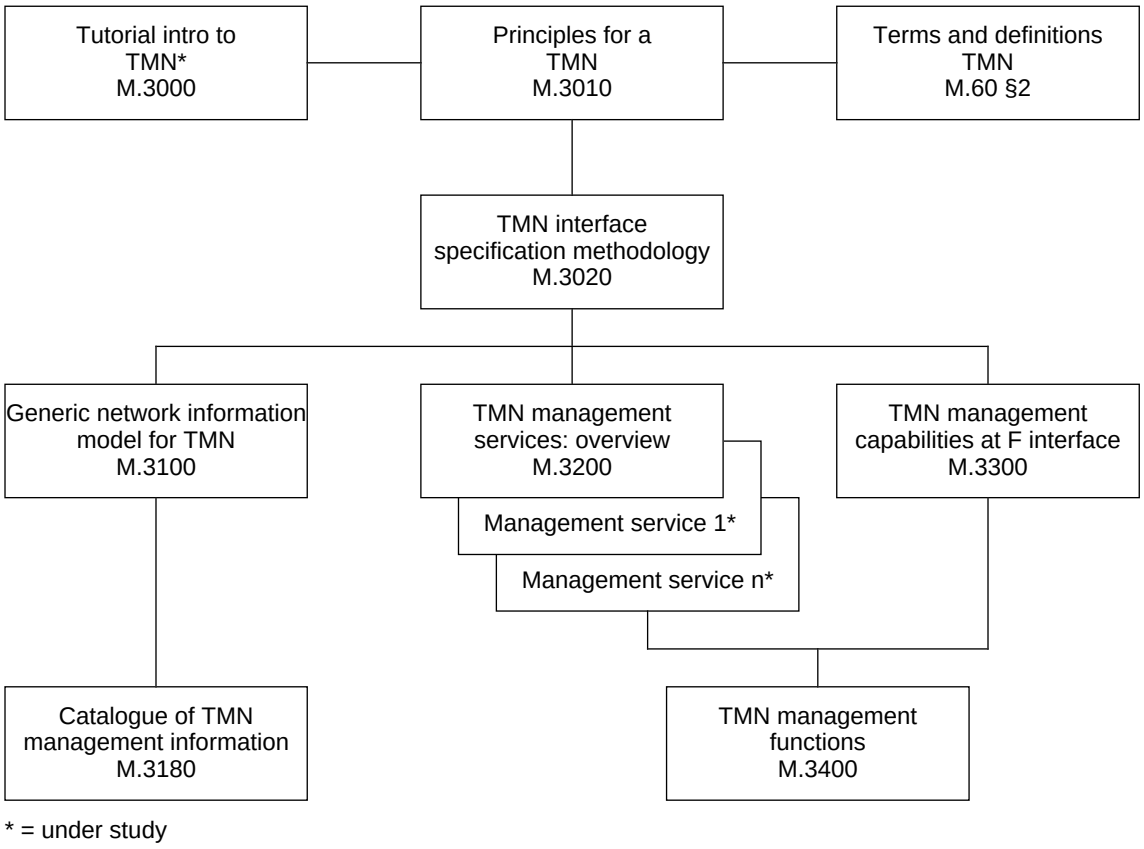


Figure 3.2: TMN related recommendations

As compared to the 1988 (blue book) version of M.30, M.3010 has been completely restructured. In M.3010 several sections have been removed from the main text and new sections have been included. Among the sections that have been removed, are those on 'Planning and Design' (which has become an

appendix) and 'Functions associated with TMN'. Among the sections that have been included, are those on the 'TMN Information Architecture'.

3.1.1 Relation with ISO/IEC

Initially there was little collaboration between the management groups of CCITT and ISO/IEC. As a result, the 1988 version of Recommendation M.30 had no ISO/IEC counterpart and ISO/IEC standards had little impact on TMN. After publication of M.30 the collaboration between CCITT and ISO/IEC was improved, which resulted in the incorporation of many OSI management ideas into TMN.

The most important changes to TMN were:

- The 'manager-agent' concept, as originally developed by ISO/IEC, was adopted. The current TMN text contains for instance a statement saying that "The description of the manager/agent concept … is intended to reflect the definitions given in X.701" (the OSI Systems Management Overview).
- ISO/IEC's 'Object Oriented' approach was copied. The current TMN text says: "… the TMN methodology makes use of the OSI systems management principles and is based on an object oriented paradigm".
- The idea of 'Management Domains' was included. A number of TMN drafts that were developed during the 1988-1992 study period contained notes saying: "CCITT SG VII and ISO have a work item on the definition of Management Domains. Resulting material should be used or referenced when available".

Despite this cooperation between the ITU-T and OSI management groups, fundamental differences in philosophy still exist. Members of the ITU management group, for example, prefer to introduce a *separate* network for the transfer of management information. This preference is clearly illustrated in Figure 3.1, which shows that information to manage the *Telecommunication network* should be transferred over a separate *Data Communication Network.*

As explained in Subsection 2.3.2, members of the OSI management group took a different approach. They preferred to use the *same* components for the network that is managed and the network over which management information is transferred[1].

The idea to introduce a separate network to transfer management information is comparable to the idea to introduce a separate network to exchange signalling information. In this sense TMN resembles to SS No. 7 networks.

1. Unfortunately, the standards do not explicitly discuss this difference. The difference becomes clear, however, after discussion with various experts or the study of existing implementations and literature (e.g. [8]).

3.1.2 Recommendation M.3010

Recommendation M.3010 defines three different architectures:
- A functional architecture.
- A physical architecture.
- An information architecture.

TMN's functional and physical architectures will be discussed in Section 3.2 and Section 3.3. TMN's information architecture describes primarily the concepts that have been adopted from OSI management; since the relevant concepts have already been discussed in Chapter 2, the information architecture will not be discussed in this chapter.

Before presenting TMN's functional and physical architecture, the most important TMN concepts will shortly be explained in terms of OSI concepts. Also the relationship between TMN's functional and physical architecture will be explained.

TMN's functional architecture is defined in terms of *function blocks* and *reference points*. Function blocks contain *functional components* (such as 'Presentation Functions' or MIBs) and may be compared to OSI protocol entities. Reference points are used to interconnect function blocks and may in OSI terminology be compared to underlying (management information) service providers (Figure 3.3).

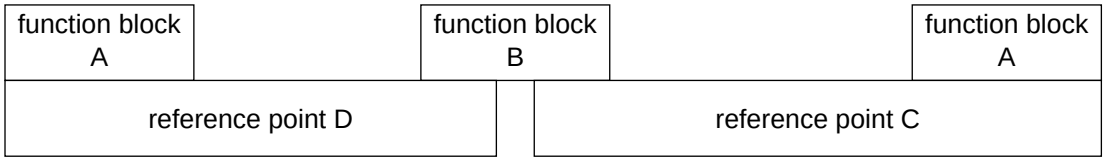


Figure 3.3: Relation between TMN concepts and OSI concepts

TMN's physical architecture is defined at a lower abstraction level. It shows how function blocks can be implemented into *physical equipment* (or *building blocks*[1]) and reference points into *interfaces* (Figure 3.4).

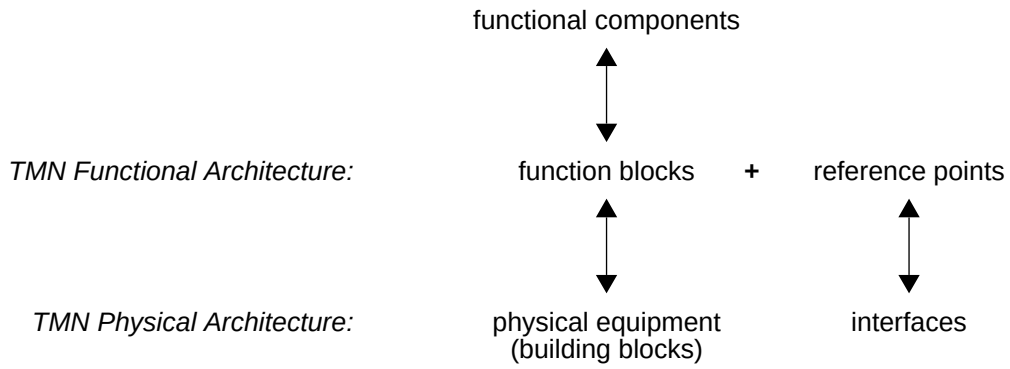


Figure 3.4: Relation between TMN Architectures

1. Recommendation M.3010 use the terms *building block* and *physical equipment* as equivalents. In the remainder of this text the term building block will be used.

3.2 Functional Architecture

Five different types of *function blocks* are defined by TMN's functional architecture. It is not necessary that all of these types are present in each possible TMN configuration. On the other hand, most TMN configurations will support multiple function blocks of the same type.

Figure 3.5 has been copied from the TMN recommendations and shows all five types of function blocks[1]. In this figure, two types (OSF and MF) are completely drawn within the box labelled 'TMN'. This way of drawing indicates that these function blocks are completely specified by the TMN recommendations. The other three types (WSF, NEF and QAF) are drawn at the edge of the box to indicate that only parts of these function blocks are specified by TMN. Subsection 3.2.1 until Subsection 3.2.5 give short descriptions these five function blocks.

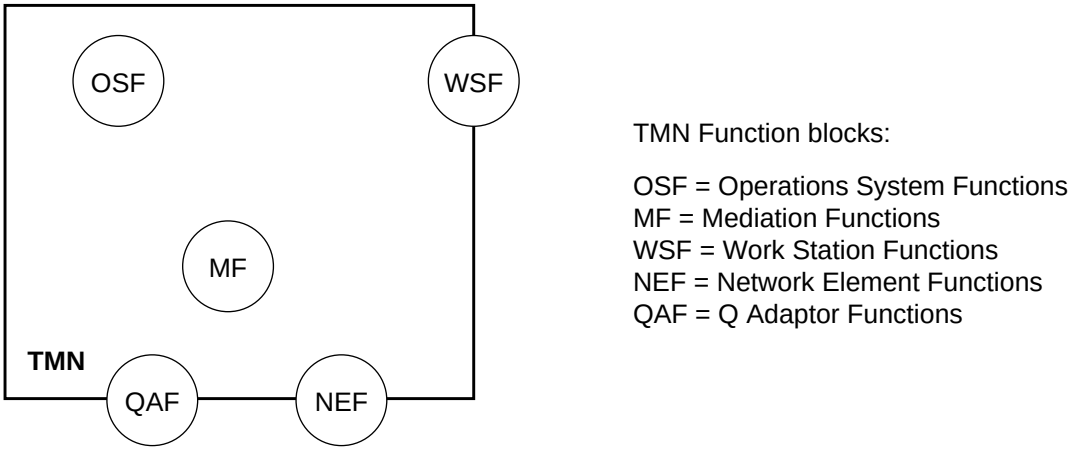


Figure 3.5: TMN Function blocks

The TMN functional architecture introduces the concept of reference point to delineate function blocks. Five different classes of *reference points* are identified. Three of them (q, f and x) are completely described by the TMN recommendations; the other classes (g and m) are located outside the TMN and only partially described.

Figure 3.6 provides an example of reference points and function blocks. The picture shows for instance that the Mediation Function (MF) can be reached via *q* reference points and that the *m* reference point can be used to reach the Q Adaptor Function (QAF) from outside TMN.

3.2.1 Network Element Functions

As explained on page 43, a typical telecommunication network consists of *exchanges* and *transmission systems*. In TMN terminology, exchanges and transmission systems are examples of *network elements* (NEs).

1. To avoid adventitious interpretations, it was decided to copy as far as possible drawings from Recommendation M.3010.

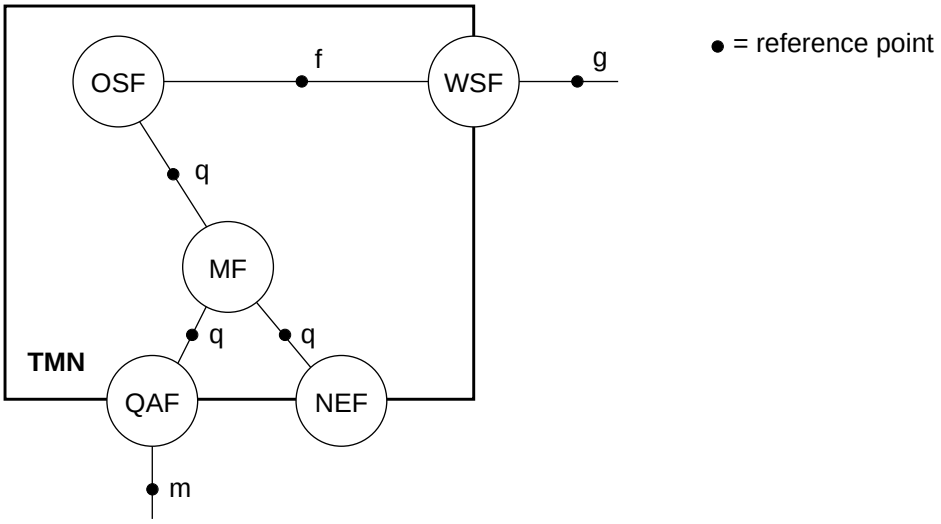


Figure 3.6: Example of reference points between function blocks

The functions that are performed by NEs are 'Network Element Functions' (NEFs). According to TMN, these functions include:

- Primary (or telecommunications) functions. These functions are the subject of management and support the exchange of data between the users of the telecommunication network.
- Management functions, which allow the NEF block to operate in an agent specific role.

As opposed to the second kind, the first kind of functions are not further defined by TMN. This explains why Figure 3.5 locates the NEF at the edge of the TMN.

3.2.2 Operations System Functions

The Operations System Functions (OSF) block initiates management operations and receive notifications. In terms of the manager-agent model, the OSF may be seen as the manager specific functions. An OS communicates with the NEs over a q_3[1] reference point.The service that is provided at such reference point is the *Common Management Information Service* (CMIS [50]).

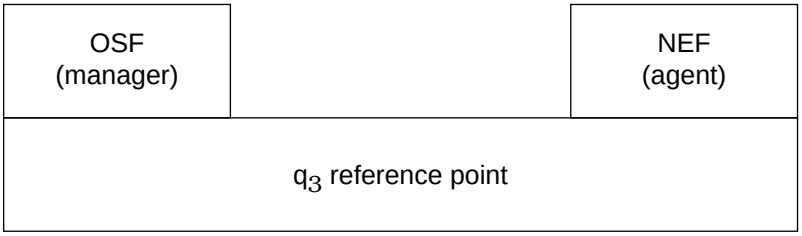


Figure 3.7: Relation between OSF, NEF and q_3

1. The 1988 version of M.30 defined three different q reference points: q_1, q_2 and q_3. After some time it appeared that an acceptable distinction between q_1 and q_2 could not be made. These two reference points were therefore replaced by the generic q_x reference point.

Within a single TMN (operated by a single administration) multiple OSFs may be defined. If necessary, these OSFs can communicate with each other over q_3 reference points. It is also possible that OSFs in different TMNs (operated by different administrations) communicate with each other; in this case communication takes place over a *x* reference points.

3.2.3 Work Station Functions

"The Work Station Function (WSF) block provides the means to interpret TMN information for the management information user. The WSF includes support for interfacing to a human user (at the g reference point). Such aspects of support are not considered to be part of the TMN". Figure 3.5 therefore locates the WSF at the edge, and the g reference point outside the TMN.

3.2.4 Q Adaptor Functions

The *Q Adaptor Function* (QAF) block is used to connect to the TMN those entities which do not support standard TMN reference points. An example is shown in Figure 3.8; in this figure a non-TMN OSF and a non-TMN NEF are connected to the TMN. The responsibility of both QAFs is to translate between *q* reference points (which are TMN reference points) and *m* reference points. Since the m reference point is a non-TMN (e.g. proprietary) reference point, Figure 3.5 showed the QAF at the edge of the TMN.

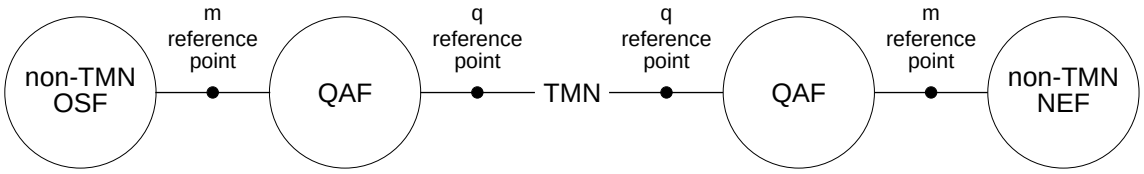


Figure 3.8: Q Adaptor Functions

3.2.5 Mediation Functions

The *Mediation Function* (MF) block is located within the TMN and acts on information passing between NEFs or QAFs, and OSFs. A MF block can be used to connect a single (Figure 3.9), as well as multiple NEFs and QAFs to an OSF. MF blocks can also be cascaded.

Among the types of MFs that can be recognized, are those that:

- Augment OSFs; examples are storage and filtering of management information.
- Augment NEFs; an example is the transformation from the local representation of management information into a standardized form.

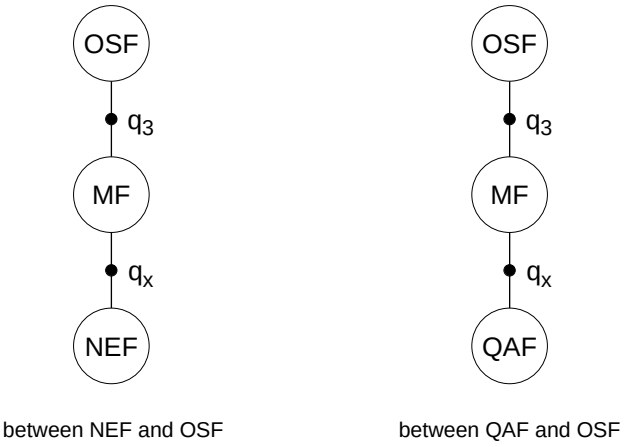


Figure 3.9: MF related to other function blocks

3.2.6 Relationship between function blocks

Now that an initial understanding of all function blocks and reference points exists, it is possible to discern all relationships between these function blocks and reference points. This relationship is given in Figure 3.10.

	NEF	OSF	MF	QAF_{q3}	QAF_{qx}	WSF	Non-TMN
NEF		q_3	q_x				
OSF	q_3	x*, q_3	q_3	q_3		f	
MF	q_x	q_3	q_x		q_x	f	
QAF_{q3}		q_3					m
QAF_{qx}			q_x				m
WSF		f	f				g**
Non-TMN				m	m	g**	

m, g = non TMN reference points

* = x reference point only applies when each OSF is in a different TMN
** = The g reference point lies between the WSF and the human user

Figure 3.10: Relation between function blocks

A function block at the top of a column may exchange management information with a function block at the left of a row over the reference point that is mentioned at the intersection of the column and row. In case an intersection is empty, the associated function blocks can not directly exchange management information between each other.

3.2.7 Further remarks

Besides the function blocks and reference points, the TMN functional architecture introduces some additional concepts. These concepts are:
- TMN's Data Communication Function
- TMN's functional components

According to recommendation M.3010, "TMN's Data Communication Function (DCF) will be used by the function blocks for exchanging information. The DCF provides layers 1 to 3 of the OSI RM".
The definition of the DCF concept has historical reasons: in initial drafts of TMN the DCF was modelled as a function block; it was therefore part of TMN's functional architecture. At present the DCF is no longer modelled as a function block; the text that describes the DCF remained, however.

Each of TMN's function blocks is itself composed of a number of *functional components*. The following functional components are defined:
- Management Application Function.
- Management Information Base.
- Information Conversion Function.
- Human Machine Adaptation.
- Presentation Function.
- Message Communication Function (MCF).

These functional components can be divided into two categories:
- The first five components belong to the first category. These components perform the actual management actions; they do not address problems related to the exchange of management information.
- The last component (MCF) belongs to the second category. This component is associated with all function blocks that require an underlying service for the exchange of their management information. "The MCF is composed of a protocol stack that allows connection of function blocks to DCFs". In many cases the MCF provides the end-to-end functions such as those found in OSI layers 4 to 7.

Recommendation M.3010 contains a picture (Figure 3.11) to illustrate the relation between function blocks, functional components, the MCF and the DCF.

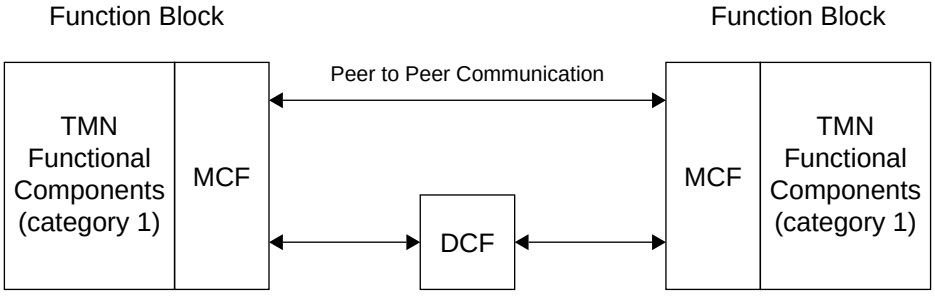


Figure 3.11: Function blocks, components, MCF and DCF

3.3 Physical Architecture

Next to a functional architecture, TMN also defines a physical architecture. The purpose of the latter is to show how function blocks should be mapped upon *building blocks* (physical equipment) and reference points upon *interfaces*. In fact, the physical architecture defines how function blocks and reference points can be implemented.

To avoid confusion between the functional and physical architecture, it is helpful to understand the following conventions. Names of reference points are written in lower case, names of interfaces in upper case (subscripts may be added). Reference points are drawn as small filled circles (bullets), interfaces as open circles. Function blocks are shown as big circles or ellipses, building blocks are drawn as boxes.

Figure 3.12: Drawing conventions

3.3.1 Building blocks

TMN's Physical Architecture defines the following building blocks:
- Network Element (NE).
- Mediation Device (MD).
- Q Adaptor (QA).
- Operations System (OS).
- Work Station (WS).
- Data Communication Network (DCN).

Building blocks always implement the function blocks of the same name (e.g. Network Elements perform Network Element Functions, Mediation Devices perform Mediation Functions etc.).

It is possible to implement multiple function blocks (of the same or of a different type) into a single building block. The Operations System, for example, may be used to implement multiple OSFs, but may also be used to implement an OSF, MF and a WSF. In the case a building block implements multiple function blocks of different types, "the choice on the building block's name is determined by the predominate usage of the block".
Figure 3.13 shows which function blocks may be implemented into which building blocks.

A special kind of building block is the Data Communication Network (DCN). As opposed to the others, this building block does not implement any function block. In fact, the DCN is *used* by other building blocks for the exchange of management information; the DCN's task is to act as a transport network.

	NEF	MF	QAF	OSF	WSF
NE	M	O	O	O	O*
MD		M	O	O	O
QA			M		
OS		O	O	M	O
WS					M

M = Mandatory
O = Optional
O* = may only be present if OSF or MF is also present

Figure 3.13: Relation between function blocks and building blocks

At first sight it seems strange that TMN defines a building block that does not implement any function block. The existence of the DCN can be understood however when we remember that previous TMN drafts (e.g. [21]) modelled the DCF as a function block (see also Subsection 3.2.7). According to these drafts, the DCF had to be implemented by a DCN and, in that case, each building block implemented at least one function block. In 1990 it was decided however to model the DCF no longer as a function block [22]. After this decision was made, the standard was not rewritten in a consistent way and the DCN is therefore still modelled as a building block.

3.3.2 Interfaces

Interfaces may be regarded as the implementations of TMN reference points. Whereas reference points may generally be compared with underlying *services*, interfaces may be compared with the *protocol stacks* that implement these services.

In most cases reference points and interfaces have a one to one mapping. However, no interfaces exist for those reference points that:

- interconnect function blocks that are implemented within a single building block,
- lay outside TMN (g and m, see Figure 3.6). Implementation of these reference points is outside the scope of TMN.

The naming of interfaces is also straightforward: an interface gets the same name (this time written in upper case) as the related reference point. Figure 3.14 shows all possible mappings.

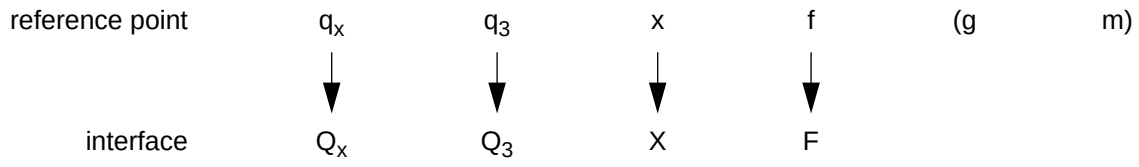


Figure 3.14: Mapping reference points upon interfaces

3.4 Responsibility Model

TMN recognizes that, corresponding to human society, a hierarchy of management responsibilities exist. Such hierarchies can be described in terms of management *layers*. This concept of management layers is discussed in the main text of recommendation M.3010 (Logical Layered Architecture). A specific application of this concept, sometimes called the *responsibility model*[1], is given in appendix II of M.3010. This application is considered to be an important aspect of TMN, this section will therefore discuss this model.

The following layers are defined by the model:

- business management layer.
- service management layer.
- network management layer.
- network element management layer.
- network element layer.

These layers, including their function blocks and reference points, are shown in Figure 3.15.

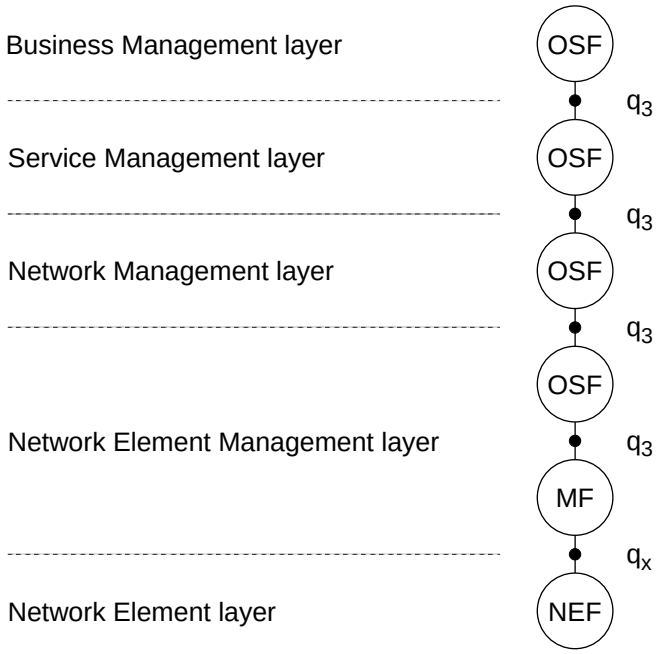


Figure 3.15: TMN Functional hierarchy

The bottom of the management hierarchy is formed by the *network element layer*. This layer contains the Network Element Functions (NEFs). In those cases where the NEFs can only be managed via a q_x reference point, a Mediation Function (MF) is needed at the next higher layer. In terms of a manager-agent relationship, the MF is the manager and the NEF is the agent. The MF will in turn be managed by an Operations Systems Function (OSF) in the same

1. The responsibility model has originally been developed by BT [9] as part of its Open Network Architecture (ONA). BT uses the name *structural architecture* for this model [71].

Network Element Management layer. Examples of functions performed by this management layer are error detection and logging of statistical data.

The Network Element Management layer is responsible for managing NEFs implemented within *single* pieces of equipment. In case the relation between NEFs implemented within *multiple* pieces of equipment becomes important, intervention of an OSF located at the *network management layer* is necessary. Routing can be seen as an example of a management activity located at this layer.

The network management layer, again, is managed by the *service management layer*. Service management is concerned with management of those aspects that may directly be observed by the users of the telecommunication network. An important part of service management is for instance Quality of Service management.

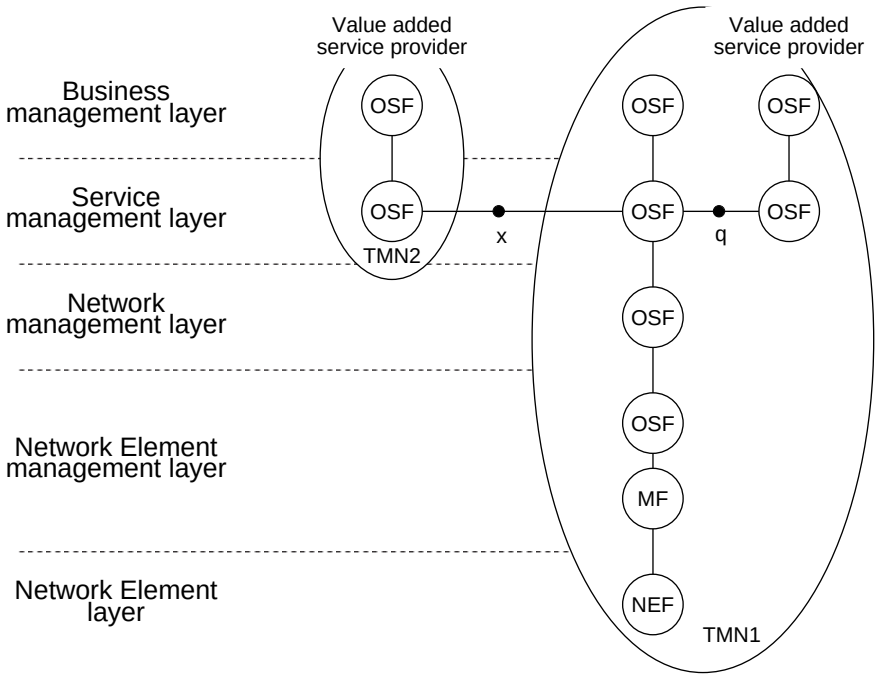


Figure 3.16: Example of Value Added Services

The idea of service Management is particularly useful in the case of Value Added Services (VAS). In such case one OSF may be responsible for management of the VAS and another OSF may be responsible for management of the telecommunications network. Both OSFs must be able to communicate with each other. If these OSFs belong to the same TMN (administration), communication is realized over a *q* reference point. If both OSFs belong to different TMNs, the *x* reference point will be used (Figure 3.16).

The business management layer is responsible for the management of the whole enterprise. This layer has a broad scope; communications management is just a part of it. Business management can be seen as *goal setting*, rather than *goal achieving*.

3.5 Analysis

The current TMN architecture, and in particular the part on TMN's Information Architecture, includes many ideas of OSI systems management (page 45). As a consequence, the analysis that was given in the previous chapter on OSI management is to a large extent also applicable to this chapter. Despite the large number of similarities between TMN and OSI management, there are also some differences; the most interesting will be examined in Subsection 3.5.1. As opposed to OSI, the concepts that have been developed specifically for TMN are not always properly defined. This is a deficiency, since without good definitions multiple interpretation may arise. Some of these interpretations will be discussed in Subsection 3.5.2.

3.5.1 Differences between TMN and OSI

An interesting difference between OSI and TMN management is that OSI has defined a *single* management architecture whereas TMN defined *multiple* architectures at different levels of abstraction.
Subsection 3.1.2 explained that TMN's functional architecture shows the various TMN management functions and TMN's physical architecture shows how these functions can be implemented into physical equipment (Figure 3.17). TMN's physical architecture is thus defined at a lower abstraction level than the functional architecture (at functional level we abstract from equipment issues, at physical level not).

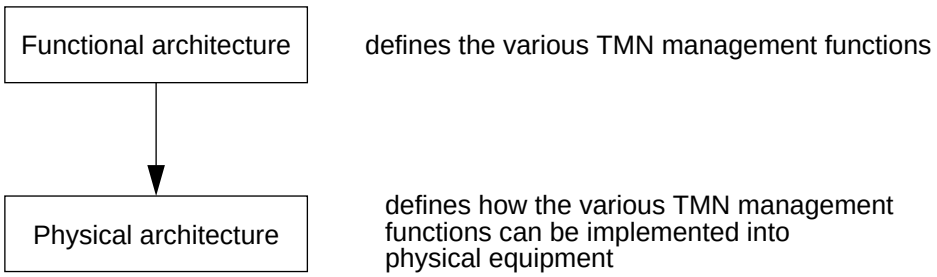


Figure 3.17: TMN has defined multiple, related architectures

In general it may be a good idea to define multiple architectures. This is particularly true in case each architecture elaborates an additional, orthogonal issue. Care should be taken, however, that the relationship between the various architectures remains easy to understand. In the specific example of TMN's functional and physical architecture, this has been the case.

A second difference between TMN and OSI management is, that TMN provides a structure for the multiple levels of management responsibility that exist in real networks; OSI management does not provide such structure. The TMN structure is known as the 'responsibility model' and was discussed in Section 3.4. The advantage of having such structure, is that it becomes easier to understand and distinguish the various management responsibilities.

A final difference between TMN and OSI management is that, as opposed to OSI, TMN suggests a conceptual separation between the network that is managed (the telecommunication network) and the network that transfers the management information (the DCN). This difference was already identified at the end of Subsection 3.1.1.
Such separation prevents the problems with fault management as discussed in the analysis section of OSI management (Subsection 2.3.2). Despite of failures in the managed network, management will always be able to access failing components. TMN has thus better fault management capabilities than OSI.

Unfortunately, a DCN requires the introduction of additional equipment and transmission systems. Besides, failures in the DCN can not be excluded, which implies that it will be necessary to manage the DCN too. The costs of introducing a DCN should therefore not be neglected!

There are also other reasons to introduce a DCN. An important reason may, for instance, be that the managed network does not provide adequate facilities to transfer management information. This is, for example, the case with telephony networks, which provide an isochronous type of service. Such type of service does not correspond to the asynchronous (packet oriented) type of service that is required by most management protocols; a DCN may thus be inevitable to manage such kinds of networks. The better fault management capabilities of the DCN are in such case only a secondary consideration.
As opposed to TMN, OSI is particularly aimed at management of datacommunication networks. The type of service provided by such networks is usually the same as the type of service required for the exchange of management information. With datacommunication networks, and thus in case of OSI, a serious consideration is needed whether the advantages of a DCN outweigh its costs.

3.5.2 Imprecise and ambiguous concepts

As opposed to the OSI management standards, recommendation M.3010 does not include a separate section that clearly *defines* its main architectural concepts (such as function block, reference point, building block and interface). To get an understanding of these concepts, readers have to derive the ideas behind these concepts from the various pieces of text in which these concepts are mentioned. As will be demonstrated in this subsection, different readers may draw different conclusions, depending on the text that has been read.

To proof that different interpretations of TMN's architectural concepts are possible, this subsection explains the *function block* and *reference point* concepts in terms of the relatively well understood concepts of the OSI Reference Model [43]. Readers who are interested in an analysis of the *building block* and *interface* concept are referred to the literature [78].

Function blocks

TMN's functional architecture contains only two pieces of text that explain what function blocks are:

- "Function blocks provide the TMN general function which enable a TMN to perform the TMN management functions".
- "Each function block is itself composed of functional components".

By reading the text on functional components, it becomes clear that most of these components can be compared to OSI application layer functions. It seems reasonable to conclude that function blocks have some relationship to OSI application layer entities.

There is one particular functional component that does not perform application layer functions, but functions of OSI layer 4 until 6. This component is the Message Communication Function, which is "associated with all function blocks". Depending on the meaning of the word *associated*, function blocks possibly also perform layer 4-6 functions.

At the time the Data Communication Function was still modelled as a function block (page 51), function blocks even performed functions that belonged to layers 1-3 of the OSI Reference Model. We may thus conclude that multiple interpretations of the term *function block* are possible.

Reference points

M.3010 gives the following descriptions of TMN reference points:

- "Reference points define service boundaries between two management function blocks. The purpose of reference points is to identify the information passing between function blocks".
- "Reference points are conceptual points of information exchange between non-overlapping management function blocks".

Also in case of reference points, multiple interpretations are possible. Figure 3.18, which is copied from M.3010, shows two of such interpretations.
In case the DCF is explicitly modelled (left part of the figure), each reference point is used to connect the Message Communication Function (MCF) to the Data Communication Function (DCF). Since the MCF provides functions similar to those found in the layers 4-6 of the OSI Reference Model and the DCF provides functions similar to those of layer 1-3, a reference point can be seen as an OSI (network layer) Service Access Point.

In case of an implicit DCF (right part of the figure), a single reference point is sufficient to connect two remote function blocks. In this case, reference points seem to bridge distance and can be seen as OSI (network) service providers.

There is even a third possibility to consider, since reference points may also be used to connect function blocks that are located at the *same* place (these func-

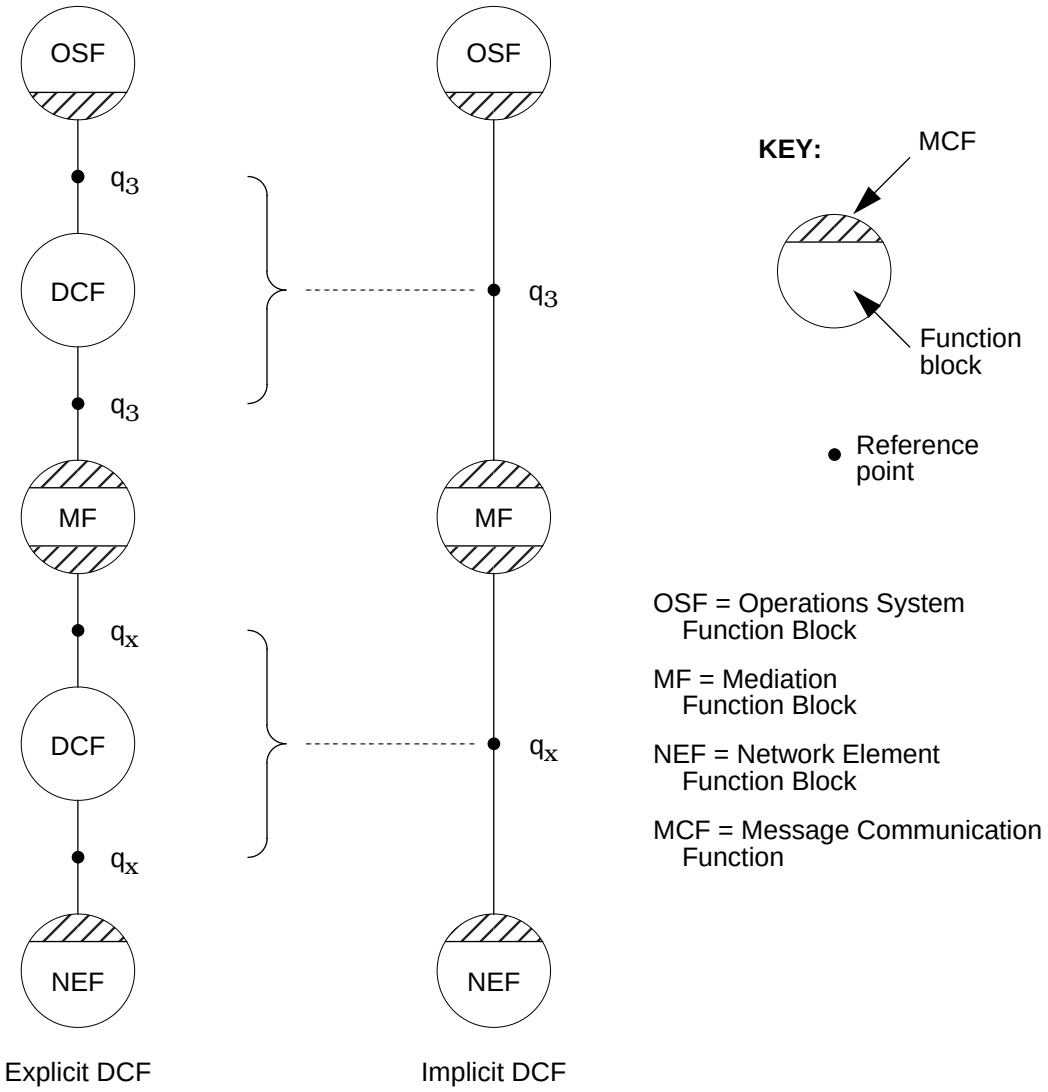


Figure 3.18: Implicit and explicit DCF

tion blocks will later be implemented into a single building block). For this possibility no corresponding OSI concept exists (OSI concepts are not meant to model the internal structure of a system).

4: Internet Management

4 Internet Management

This chapter discusses and analyses the management approach that is standardized by the Internet Engineering Task Force (IETF). This approach is also known as the Simple Network Management Protocol (SNMP) or the TCP/IP management approach.

In the second half of the past decade the Internet grew to a size that management of the Internet could no longer be provided on an ad hoc basis: a structured and standardized approach to Internet management was required.
In 1987 three management proposals therefore appeared. One of these, the *High-level Entity Management System / Protocol* (HEMS / HEMP) was withdrawn soon [86][87], so only two remained: the *Simple Network Management Protocol* (SNMP) and *Common Management Over TCP/IP* (CMOT). At the March 1988 meeting of the Internet board, the decision was made to use SNMP in the short-term and CMOT in the long-term [89].

CMOT [93] was an attempt to use OSI systems management standards (such as CMIP) in the Internet environment [23][70]. CMOT faced the same problems as OSI management: the specifications did not appear in time, there were virtually no implementations and operational experience could not be obtained. As a result, the support for CMOT slowly diminished. In 1992 all work on CMOT was stopped.

SNMP [92] is actually a further development of SGMP (Simple Gateway Monitoring Protocol)[88]. SGMP was aimed at management of Intermediate Systems (gateways)[13]. Because SGMP appeared to be a success, it was decided to extend its scope and include management of End Systems. To reflect this change, the protocol was renamed into SNMP.

An interesting difference between the IETF and ISO is that the IETF takes a more pragmatic and result driven approach than ISO. In the IETF it is for instance unusual to spent much time on architectural discussions; people prefer to use their time on the development of protocols and implementations. This different attitude explains why *no special standards have been defined for the Internet management architecture*; only protocols and MIBs have been standardized. Fortunately many articles and books have been written (even by the editors of the standards) that describe the principles behind Internet management (e.g. [3][14][15][16][101][102]). From these publications the following ideas appear:

- *All* systems connected to the network should be manageable with SNMP.
- The cost of adding network management to existing systems should be minimal.
- It should be relatively easy to extend the management capabilities of existing systems (by extending the Management Information Base).
- Network management must be robust. Even in case of failures, a small set of management capabilities must still be available.

Apparently SNMP was the right solution at the right time. Already a few years after publication of the standard most datacommunication equipment could be managed via SNMP; SNMP had become the de facto standard for management of datacommunication networks. Still SNMP has some deficiencies. In 1992 work was therefore started to develop an improved version of SNMP; this new version was called SNMPv2.

In this chapter both protocol versions will be presented. Section 4.1 discusses the original SNMP protocol, while SNMPv2 will be discussed in Section 4.2. It should be noted that both protocols only define *how* management information should be exchanged; they do not define *which* management information exists. Such information is defined by the various MIB standards; Section 4.3 discusses some of the most important ones. Section 4.4 provides an analysis of Internet management.

4.1 The original SNMP protocol

The ideas behind SNMP are relatively straightforward and easy to understand. In fact, there is little difference between the ideas behind SNMP and the ideas that existed in ISO around 1987, thus before ISO adopted the Object Oriented approach for management. Examples of such common ideas are the manager-agent concept, the idea to use the managed functions also for the exchange of management information, the idea to use *GET* and *SET* PDUs for operations on management information, the idea to use ASN.1 for the definition of management information and the idea of a MIB.

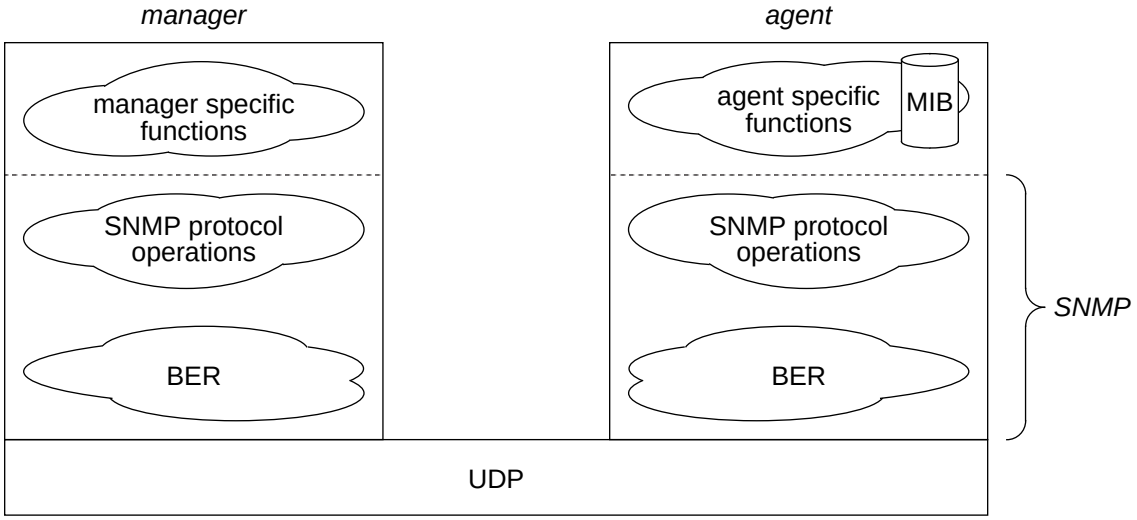


Figure 4.1: Internet management structure

With SNMP, a single manager may control many agents. As shown in Figure 4.1. the SNMP protocol is built upon the User Datagram Protocol (UDP), which is a connectionless transport protocol [83]. Since the Internet management information as well as the formats of SNMP PDUs are defined according to (a subset of) the ASN.1 syntax, encoding functions are needed immediately on top of UDP. These functions operate according to the Basic Encoding Rules

(BER). Five types of SNMP PDUs are defined: *GetRequest*, *GetNextRequest*, *SetRequest*, *Response* and *Trap*. The SNMP protocol standard does not address the functions that are specific for managers or agents; the SNMP standard is thus restricted to the functions below the dotted line (Figure 4.1). This implies that the scope of SNMP is equivalent to the one of CMIP; as opposed to CMIP no standard exists, however, which defines the service that is provided on top of SNMP.

The IETF has not (yet) defined *manager specific* functions; further work in this area is therefore urgently needed [101]. The lack of manager specific functions sharply contrasts to the overwhelming number of agent specific functions (primarily MIBs) that have been defined. Section 4.3 gives an overview of these MIBs.

4.1.1 Transport mappings

The choice to operate SNMP over the connectionless UDP has several implications.

In the first place UDP is unreliable, which means that user data may get lost. The decision to use an unreliable transport service provider, has been taken deliberately. The reason is that even in case of repeated provider failures, it should still be possible to exchange *some* part of the management information. With a reliable (connection-oriented) provider this may not be possible. Connection-oriented providers are designed according to an 'all or nothing' approach: either *all* data will be delivered or *nothing* will be delivered. If data can not be delivered, the connection will be released. Connectionless providers are designed according to an 'best-effort' approach: even in case of failures some of the data may arrive at the destination. Management may therefore still be possible, although in a limited way.

It is interesting to see that the SNMP protocol does not itself perform retransmissions. The responsibility to detect data loss and initiate retransmission is left to the manager, because it is assumed that managers are usually better equipped to determine *whether* and *when* retransmissions are required.

A second implication of using a connectionless transport protocol, is that managers should perform some kind of *polling* to detect whether agents are still operational. With connection-oriented providers (e.g. OSI's presentation service) this would not be necessary, because such providers already include lifetime control functions[1]. Such functions periodically check whether the remote systems (in our case the agents) are still operational. In case they went down, the provider takes the initiative to release the connection and informs the user (in our case the manager).

1. In OSI, this function is part of the connection oriented transport protocol.

A characteristic of UDP is that packets can not exceed a certain size. To ensure that only limited size packets will be generated, the SNMP protocol has defined a number of rules. One of these rules is that, if the response to a certain SNMP request would exceed the maximum packet size, no information will be returned at all. Managers should be aware of this rule and, instead of issuing a single all-embracing request, issue multiple smaller request to get the information piece by piece. Unfortunately, managers will in many cases not be able to predict the amount of information that can be obtained via a single request.

Although SNMP is intended to operate over UDP, there are also RFCs that define how to operate SNMP on top of other protocols (e.g. Ethernet, IPX or even OSI).

4.1.2 Protocol operations

In SNMP, communication from the manager to the agent system is performed in a confirmed way. The SNMP entity at the manager's side takes the initiative by sending one of the following PDUs: *GetRequest*, *GetNextRequest* or *SetRequest*. The *GetRequest* and *GetNextRequest* are used to retrieve management information from the agent, the *SetRequest* is used to store (or change) management information. After reception of one of these PDUs, the SNMP entity at the agent's side responds with a *Response* PDU (Figure 4.2). This PDU carries the requested information or indicates failure of the previous request.

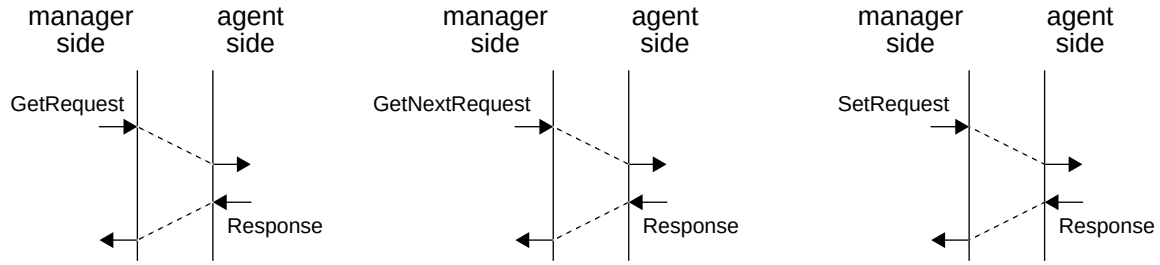


Figure 4.2: Managing system takes the initiative

It is also possible that the SNMP entity at the agent's side takes the initiative. This happens in case the agent detects some extraordinary event, such as a re-initialization or a status change at one of its links. As a reaction, the agent's SNMP entity sends a *Trap* PDU to the managing system (*Traps* may be compared to OSI *Event-Reports*). Reception of the *Trap* is not confirmed (Figure 4.3).

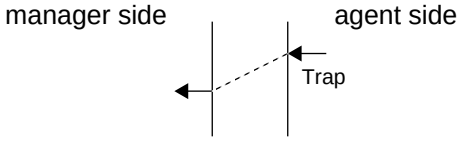


Figure 4.3: Agent system takes the initiative

SNMP does not describe how to relate the various *Get*, *Set* and *Trap* interactions. What to do after reception of a *Trap* is, for example, not defined by SNMP. Instead, determination of this relationship is considered to be a responsibility of the manager specific functions.

4.2 SNMPv2

Since publication of the original SNMP protocol, several proposals have been presented to improve SNMP. In 1992 it was decided to collect these proposals and produce a new standard: SNMPv2. Unfortunately SNMPv2 became far more complex than the original SNMP [79]; whereas the description of the original SNMP protocol required, for example, only 35 pages, the description of SNMPv2 required about 250 pages[1].

The main achievements of SNMPv2 are the improved performance (Subsection 4.2.1), the better security (Subsection 4.2.2) and the possibility to build a hierarchy of managers (Subsection 4.2.3).

4.2.1 Performance

As explained on page 64, the original SNMP protocol includes a rule which states that if the response to a *Get* or *GetNext* request would exceed the maximum size of a packet, no information will be returned at all. Since managers can not determine the precise size of response packets in advance, they usually take a conservative guess and request per PDU just a small amount. To obtain all information, managers may be required to issue a large number of consecutive requests.

To improve performance, SNMPv2 introduced the *GetBulk* PDU. As opposed to the *Get* and *GetNext*, the response to the *GetBulk* always returns as much information as possible. If the requested information exceeds the maximum size of an UDP packet, the information will be truncated and only the part that fits within the packet will be returned.

4.2.2 Security

The original SNMP protocol had, except for a simple mechanism which involved the exchange of passwords (the term 'community string' was used to denote this password), no security features. To solve this deficiency, SNMPv2 introduced a full-fledged security mechanism. This mechanism is based upon the use of 'parties' and 'contexts'; two concepts that can not be found in other management approaches. Although the SNMPv2 standards include definitions of both concepts, these definitions are difficult to understand. This subsection presents a somewhat simplified view of these concepts.

Parties have some resemblance to protocol entities. Usually multiple parties are active in a single SNMPv2 subsystem and these various parties will be configured in different ways. One party may, for instance, be configured such that it is prepared to communicate with every other party in every other system.

1. These numbers do not include the pages describing the Structure of Management Information.

Another party may be configured such that it is only prepared to interact with one particular remote party. In such case, the MD5 authentication mechanism is used to ensure the authentication of the other party. Finally parties may be configured in a way that they are only prepared to interact with particular remote parties and in addition require that all management information is encrypted according to the DES algorithm.

A graphical representation of parties is provided in Figure 4.4. In this figure three parties have been configured in the manager system (Pa1, Pa2 and Pa3) and three parties in the agent system (Pb1, Pb2 and Pb3).

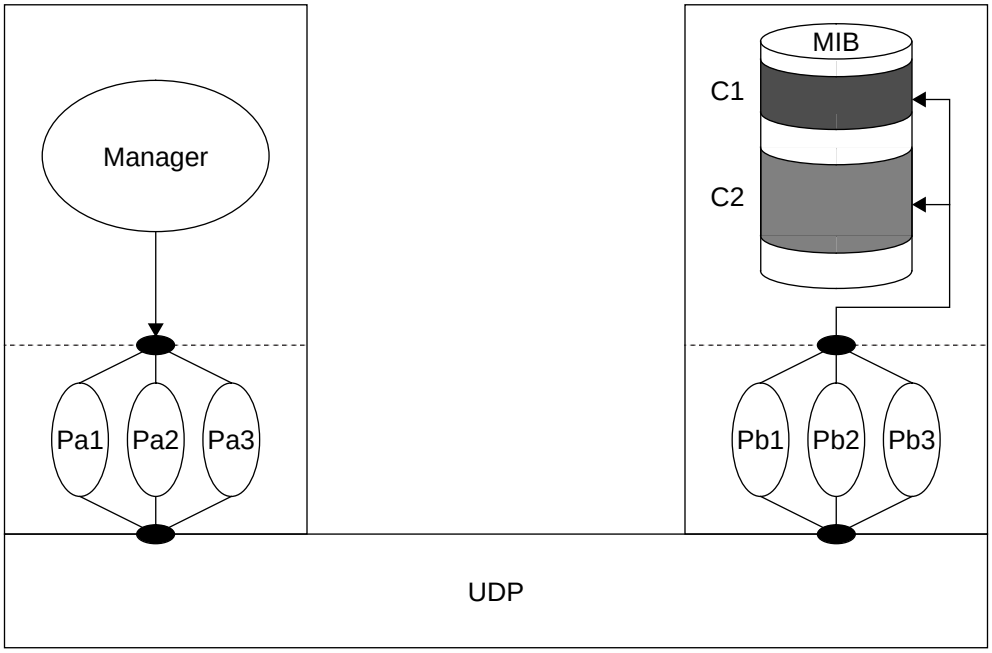


Figure 4.4: Parties and contexts

To control access to the various parts of a MIB, SNMPv2 has introduced the *context* concept. Each context refers a specific part of a MIB. In the example of Figure 4.4, context C1 and context C2 refer to the two dotted areas in the MIB. Contexts may be overlapping and are dynamically configurable, which means that contexts may be created, deleted or modified during the network's operational phase. Different contexts may be configured for different systems.

remote party	local party	context	operation
Pa1	Pb1	C1	get
Pa2	Pb2	C1	get
Pa3	Pb3	C1	get + set
Pa3	Pb3	C2	get

Figure 4.5: Example of an Access Control List (ACL)

To determine which parties are allowed to perform which operations upon which part of the MIB, SNMPv2 has associated with each agent an Access Control List (ACL). Figure 4.5 shows an example of such a list. The first row indicates that party `Pa1`(in the manager system) may perform *Get* operations via party `Pb1` (in the agent system) on that part of the MIB that is identified by context `C1`. The third row shows that `Pa3` may via `Pb3` also perform *Set* operations on this MIB part.

4.2.3 Management hierarchy

Practical experience with the original SNMP protocol showed that in many cases managers are unable to manage more than a few hundred agent systems [3]. The cause for this restriction is in SNMP's polling nature: the manager must periodically poll every system under his control, which takes time. To solve this problem, SNMPv2 introduced the idea of *intermediate level managers*. Polling is now performed by a number of such intermediate level managers under control of the top level manager.

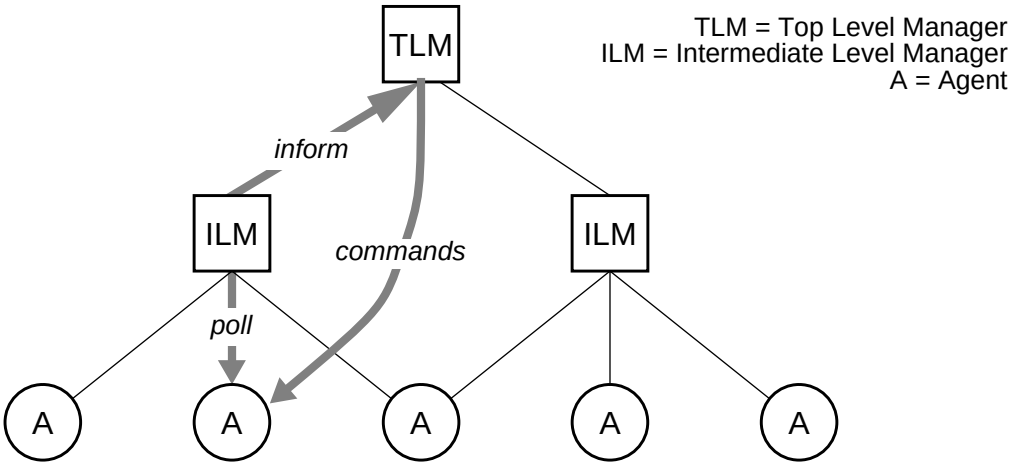


Figure 4.6: Intermediate Level Managers

Figure 4.6 shows an example. Before the intermediate level managers start polling, the top level manager tells the intermediate level managers which variables must be polled in which agents. Besides, the top level manager tells the intermediate level managers of the events he wants to be informed about. After the intermediate level managers are configured, they start polling. In case an intermediate level manager detects in a particular agent an event about which the top level managers wanted to be informed, a special *Inform* PDU is generated. After reception of this PDU, the top level manager directly operates upon the agent that caused the event.

4.3 MIBs

To identify all variables that can be managed, a large number of Management Information Base (MIB) standards have been developed. Next to these standards, one special standard exists defining *how to describe* MIB variables. This

standard is called the Structure of Management Information (SMI). It defines for instance the subset of ASN.1 constructs that can be used to describe management variables [90].

To ensure the unique identification of each management variable, the SMI introduces the concept of a naming tree. The leaves of this tree represent the actual management information. An (imaginary) example of this is shown in Figure 4.7; in this figure the object identifier of the network address is *root.1.1*, the object identifier of the collision counter is *root.2.2* and the identifier of the token holding timer is *root.3.4*.

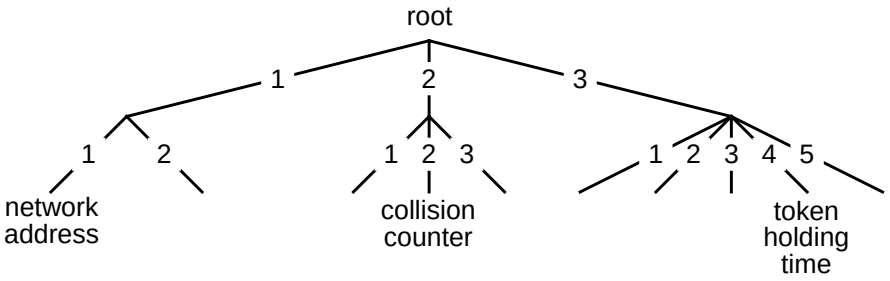


Figure 4.7: Concept of naming tree

The MIB-II[1] [94] is the most important and probably best known MIB; it contains all the variables to control the major Internet protocols (e.g. IP, ICMP, UDP, TCP, EGP and SNMP). The structure of this MIB is simple: all management variables that belong to the same protocol are grouped together (Figure 4.8). Within a protocol group there is hardly any additional structure that helps understanding the various variables within that group.

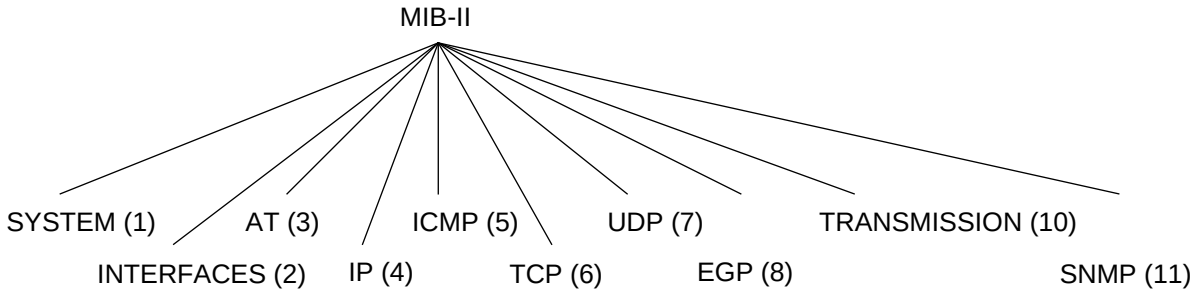


Figure 4.8: The various protocol groups of the MIB-II

Soon after definition of the MIB-II other MIBs appeared; Figure 4.9 shows some of the standardized ones.

Next to the standardized MIBs there are also a large number of enterprise specific MIBs. Together these MIBs define more than twenty-thousand management variables [63]. Unfortunately no clear structure has been developed to explain the relationship between these MIBs; the only indication of a MIBs purpose is its name.

1. The suffix II was added to indicate that this MIB replaces the earlier defined 'Management Information Base for management of TCP/IP based internets' [91].

Title	RFC	Date
MIB-II	1213	March 1991
IEEE 802.5 Token Ring	1231	May 1991
Appletalk	1243	July 1991
OSPF version 2	1253	August 1991
Remote Network Monitoring	1271	November 1991
IP Forwarding Table MIB	1354	July 1992
RIP Version 2	1389	January 1993
DS1 and E1 Interface Types	1406	January 1993
DS3 and E3 Interface Types	1407	January 1993
X.25	1461	May 1993
Point-to-Point Protocol	1471-1474	June 1993
Bridges	1493	July 1993
FDDI	1512	September 1993
Remote Network Monitoring - Token Ring	1513	September 1993
Host Resources	1514	September 1993
IEEE 802.3 Medium Attachment Units	1515	September 1993
IEEE 802.3 Repeater Devices	1516	September 1993
Source Routing Bridges	1525	September 1993
DECnet Phase IV Extensions	1559	December 1993
Network Services Monitoring	1565	January 1994
Mail Monitoring	1566	January 1994
X.500 Directory Monitoring	1567	January 1994
SNA APPN Node	1593	March 1994
SONET/SDH Interface	1595	March 1994
Frame Relay Service	1604	March 1994
Domain Name System	1611-1612	May 1994
Uninterrupted Power Supply	1628	May 1994
Ethernet-like Interface Types	1643	July 1994
Border Gateway Protocol	1657	July 1994
Character Stream Devices	1658	July 1994
RS-232-like Hardware Devices	1659	July 1994
Parallel-printer-like Hardware Devices	1660	July 1994
SNA NAU	1666	August 1994
SMDS - SIP Interface Type	1694	August 1994
ATM	1695	August 1994
Modem	1696	August 1994
Relational Database Management System	1697	August 1994

Figure 4.9: Some existing MIB definitions

4.4 Analysis

Internet management can be compared to OSI management. In fact, Internet management uses many of the concepts that existed in OSI at the time SNMP started (around 1988). As a result, the remarks that were made in the analysis section of OSI management (Section 2.3) are to some extent also applicable to Internet management. As opposed to OSI management, however, Internet management uses only a small part of the managed functions for the exchange of management information. Problems with fault management (see Subsection 2.3.2) are therefore less likely to occur.

4.4.1 The management architecture has not been described

No standards have been produced defining the Internet management architecture. To get an understanding of the architectural concepts behind Internet management, readers have to derive the meaning of the various concepts from the protocol standards. Although this may not be a problem in case of the original version of SNMP, it certainly is a problem with SNMPv2. Experience has shown that without a good understanding of these concepts, it is difficult to implement SNMPv2 [79]. A recommendation for the IETF is therefore to develop such a standard.

The concepts that cause most of the problems, are parties and contexts. Although the interpretation of these concepts given in this thesis (Subsection 4.2.2) may be sufficient to understand most parts of the SNMPv2 standards, certain parts of the standards are based upon some other interpretation. A good example of such alternative interpretation can be found in the standards that define how to use intermediate level managers. According to these standards a context does not only refer to the specific part of a MIB, but also identifies one of the agents that is controlled by the intermediate level manager.

4.4.2 Too many management variables

Now that thousands of management variables have been defined, the lack of a good functional structure to classify these variables has become a problem. Without such structure, managers will be confronted with large lists of management variables. To determine which variables must be watched and which modifications must be made, managers must understand the precise meaning of many variables.

In case management is performed by human beings, it is unlikely that there will be many people with sufficient ready knowledge. As a consequence, it can be expected that managers need a lot of time before they decide what to do. Network management may therefore become a time-consuming and thus expensive activity.

4.4.3 Manager specific functions have not been defined

The Internet management standards explain how individual management operations, such as *GET* and *SET*, should be performed. Currently they do not specify, however, the sequence in which these operations should be performed to solve particular management problems. Such sequences are part of the 'manager specific functions' (see Figure 4.1); until now the IETF has not defined such functions.

> *Example:* Suppose a router breaks down. Actions must be initiated by management to prevent data from getting lost. These actions include the change of routing tables. Since networks consist of thousands of systems, management must decide which tables to change and which not. Of course, management must also specify the exact contents of the modified routing tables.
> Internet management standards do not describe any of these actions. Instead, Internet management provides only a general approach to read and modify individual management variables.

The approach that is taken by Internet to manage networks is comparable to an approach in which debuggers are used to 'manage' computer programs.
Ordinary debuggers allow programmers to watch and modify program variables. A debug program does not help, however, to determine which variables must be watched and which modifications must be made. Such decisions must be made by the programmer; the debugger only helps to access the variables.
Internet management standards define *distributed* 'debuggers'. These 'debuggers' allow managers to watch and modify management variables; they do not tell which variables must be watched and which modifications must be made. Such decisions must be taken by the 'manager specific functions' (e.g. the operator); Internet management standards only tell how to access management variables.

Part II

An Alternative Approach to Network Management

5: Identification & classification of management

References

[1] Aidarous S.E., Proudfoot D.A., Dam X.: "Service Management in Intelligent Networks", IEEE Network Magazine, January 1990

[2] Autrata M., Strutt C.: "DME Framework and Design", in: Network and Distributed Systems Management, Chapter 23, Addison-Wesley Publishing Company, 1994

[3] Ben-Artzi A., Chandna A.,Warrier U., "Network management of TCP/IP Networks: Present and Future", in: IEEE Network Magazine, page 35-43, July 1990

[4] Black U.D.: "Network Management Standards - The OSI, SNMP and CMOL Protocols", McGraw-Hill Series on Computer Communications, 1992

[5] Blaauw G.A., Brooks F.P.: "Computer Architecture", lecture notes (draft), University of Twente, Department of Computer Science, Enschede, The Netherlands, August 1985

[6] Bogaards K., Pires L., Pras A., Schot J.: "The Pangloss method", Proceedings of the Esprit Conference, Elsevier, 1988

[7] Bogaards K.: "A Methodology for the Architectural Design of Open Distributed Systems", Ph.D. thesis, University of Twente, 1990, ISBN 90-9003554-0

[8] Bootman S., Shabana M.: "Generic building blocks for the Telecommunications Management network", Globecom - IEEE Global Telecommunications Conference & Exhibition, 1988, Conference record Volume 1, page 163-167

[9] Boyd R.T., Brodrick K.J.: "Operational Support Systems for the future Local Network", BT Technology Journal, Vol. 7, No. 2, April 1989, page 136-150

[10] Brooks Jr. F.P.: "The mythical man-month", Addison-Wesley publishing company, 1975

[11] Brugnoni S., Bruno G., Manione R., Montariolo E., Paschetta E., Sisto L.: "An Expert System for Real Time Fault Diagnosis of the Italian Telecommunications Network", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 617-628, North-Holland, 1993

[12] Carl-Mitchell S., Quarterman J.S.: "Building Internet Firewalls", Unixworld, Febr. 1992, page 92-102

[13] Case J.D., Davin J.R., Fedor M.S., Schoffstall M.L.: "Introduction to the Simple Gateway Monitoring Protocol", in: IEEE Network, page 43-49, March 1988

[14] Case J.D., Davin J.R., Fedor M.S., Schoffstall M.L.: "Network management and the design of SNMP", in: ConneXions, the Interoperability Report, page 22-26, March 1989

[15] Case J.D., Davin J.R., Fedor M.S., Schoffstall M.L.: "Internet network management using the Simple Network Management protocol", in Proceedings of the 14th IEEE Conference on Local Computer Networks, page 156-159, Oct. 1989

[16] Case J.D., Davin J.R., Fedor M.S., Schoffstall M.L.: "Keeping it simple (network management)", in Unix Review, Vol. 8, No. 3, page 60-66, March 1990

[17] Cassel L.N., Partridge C., Westcott J.: "Network Management Architectures and protocols: Problems and Approaches", IEEE Journal on Selected Areas in Communications, Vol. 7, No. 7, page 1104-1114, September 1989

[18] CCITT: "Recommendation E.410 - Telephone Network and ISDN - Quality of Service, Network Management and Traffic Engineering - International Network Management - General Information", Geneva 1992

[19] CCITT Blue Book: "Recommendation M.30, Principles for a Telecommunications Management Network", Volume IV - Fascicle IV.1, Geneva 1989

[20] CCITT: "Recommendation M.3010, Principles for a Telecommunications Management Network", Geneva 1992

[21] CCITT COM IV-42-E, Question 23/IV: "Draft Recommendation M.30 - Version R1", November 1990

[22] CCITT COM IV-61-E, Question 23/IV: "Draft Recommendation M.30 - Version R4", August 1991

[23] Cerf V.: "Network management comes of age in the Internet", in: ConneXions, the Interoperability Report, page 2, March 1989

[24] Cheswick W.R.: "Firewalls and Internet Security", Addison-Wesley, 1994

[25] CMIP Run!; "New APIs for Management Data", Vol 3, No 2, 2nd Q '94, page 11

[26] Comer D.E., Stevens D.L.: "Internetworking with TCP/IP - Volume 2: Design, Implementation and Internals", Prentice Hall International Editions, 1991

[27] Data Communications: "The Price of Failure", 21 June 1991, page 11

[28] Eertink H., Wolz D: "Symbolic Execution of LOTOS Specifications", in: Proceedings of the 5th International Conference on Formal Description Techniques, FORTE '92, ed. Michel Diaz, R. Groz.

[29] Eldering A.M.: "Realizing Management Transactions", Msc-thesis University of Twente, Enschede, The Netherlands, 1994

[30] Elmer-DeWitt P.: "Ghost in the machine", in: Time, 29 January 1990

[31] Embry J., Manson P., Milham D., "An Open Network Management Architecture: OSI/NM Forum Architecture and Concepts", in: IEEE Network Magazine, page 14-22, July 1990

[32] Embry J., Manson P., Milham D., "Interoperable Network Management: OSI/NM Forum Architecture and Concepts", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 29-44, North-Holland, 1991

[33] Fernandes F.P.L.B., Monteiro E.H.S., "An Application Process Framework for Advanced Communications Management", in: Proceedings of the fifth RACE TMN Conference, 1.3/2 page 1-15

[34] Ferreira Pires L.: "The Lotosphere Design Methodology: Basic Concepts", Final deliverable, LOTOSPHERE project, ESPRIT Ref: 2304, March 1992

[35] Frontini M., Griffin J., Towers S.: "A Knowledge-Based System for Fault Localisation in Wide Area Networks", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 519-530, North-Holland, 1991

[36] Geihs K., Mann A.: "ODP Viewpoints of IBCN Service Management", Technical Report N0. 43.9104, IBM Deutschland GMBH, Heidelberg

[37] Gering M.: "CMIP versus SNMP", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 347-359, North-Holland, 1993

[38] Goldman J., Hong P., Jeromnimon C., Louit G., Min J., Sen P.: "Integrated Fault Management in Interconnected Networks", in: Proceedings of the IFIP TC6/WG 6.6 First International Symposium on Integrated Network Management, page 333-344, North-Holland, 1989

[39] Goodman R.M., Ambrose B.: "A Hybrid Expert System / Neural Network Traffic Advice System", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 607-616, North-Holland, 1993

[40] Goyal S.K.: "Knowledge Technologies for Evolving Networks", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 439-461, North-Holland, 1991

[41] Graham J.: "The Penguin Dictionary of telecommunications", Penguin books, 1985

[42] Halsall F., Modiri N.,An Implementation of an OSI Network Management System", in: IEEE Network Magazine, page 44-53, July 1990

[43] ISO 7498: "Information Processing Systems - Open Systems Interconnection - Basic Reference Model", Geneva, 1984

[44] ISO 7498-4: "Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 4: Management Framework", Geneva, 1989

[45] ISO 8073: "Information Processing Systems - Open Systems Interconnection - Connection oriented transport protocol specification", Geneva, 1992

[46] ISO 8473: "Information Processing Systems - Data Communications - Protocol for providing the Connectionless-mode Network Service", Geneva, 1988

[47] ISO 8802/5: "Information Processing Systems - Local Area Networks - Token Ring Access Method and Physical Layer Specification", Geneva, 1987

[48] ISO 9542: "Information Processing Systems - Data Communications - End System to Intermediate System Routeing Information Exchange Protocol for use in conjunction with the Protocol for the Provision of the Connectionless-mode Network Service", Geneva, 1988

[49] ISO TR 9575: "Information Processing Systems - Data Communications - OSI Routeing Framework", Geneva, 1990

[50] ISO 9595: "Information Processing Systems - Open Systems Interconnection - Common Management Information Service Definition", Geneva, 1990

[51] ISO 9596: "Information Processing Systems - Open Systems Interconnection - Common Management Information Protocol", Geneva, 1991

[52] ISO 10038: "Information Processing Systems - Data Communications - MAC Bridges", Geneva, 1993

[53] ISO 10040: "Information Processing Systems - Open Systems Interconnection - Systems Management Overview", Geneva, 1992

[54] ISO DIS 10165-1: "Information Processing Systems - Open Systems Interconnection - Structure of Management Information - Part 1: Management Information Model", Geneva, 1993

[55] ISO 10589: "Information Processing Systems - Data Communications - Intermediate to Intermediate System Routing Information Exchange Protocol for use in Conjunction with ISO 8473", Geneva, 1992

[56] ISO 10737: "Information Processing Systems - Open Systems Interconnection - Specification of the elements of Management Information relating to OSI Transport layer Standards", Geneva

[57] ISO 10747: "Information Processing Systems - Data Communications - Protocol for Exchange of Inter-Domain Routing Information among Intermediate Systems to Support Forwarding of ISO 8473 PDUs", Geneva

[58] ISO/TC 97/SC 16 N1719: "OSI Management Framework - Fourth Working Draft", October 1983

[59] ISO/TC 97/SC 21 N382: "Procedures for Management Information Service Standardization", February 1985

[60] ISO/TC 97/SC 21 N975: "OSI Management Framework - Seventh Working Draft", November 1985

[61] ISO/TC 97/SC21 N1388: "Proposal for further consideration of the OSI Management Architecture", Egham, September 1986

[62] ISO/TC 97/SC21 N3003: "Summary of Voting on DIS 7498-4, Information Processing Systems - Open Systems Interconnections - Basic Reference Model - Part 4: Management Framework", August 1988

[63] Jander M.: "SNMP: coming soon to a network near you", Data Communications, November 1992, page 66-76

[64] Jordaan J.F., Paterok M.E.: "Event Correlation in Heterogeneous Networks using the OSI Management Framework", in: Proceedings of the IFIP TC6/WG

6.6 Third International Symposium on Integrated Network Management, page 683-695, North-Holland, 1993

[65] Klerer S.M.: "System Management Information Modelling", in: IEEE Communications Magazine, page 38-44, May 1993

[66] Kobayashi Y., "Standardization Issues in Integrated Network Management", in: Proceedings of the IFIP TC6/WG 6.6 Symposium on Integrated Network Management, page 79-90, North-Holland, 1989

[67] Lewis L.: "A Case-based Reasoning Approach to the Resolution of Faults in CommunicationS Networks", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 671-682, North-Holland, 1993

[68] Lor K. E.: "A Network Diagnostic Expert System for Acculink Multiplexers Based on General Network Diagnostic Scheme", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 659-669, North-Holland, 1993

[69] Masahiko Matsushita: "Telecommunication Management Network", NTT Review, Vol. 3 No. 4, July 1991, page 117 - 122

[70] McCloghrie K., Rose M.T.: "Network management of TCP/IP based internets", in: ConneXions, the Interoperability Report, page 3-9, March 1989

[71] Milham D.J., Willetts K.J.: "BT's Communications Management Architecture", in: Proceedings of the IFIP TC6/WG 6.6 Symposium on Integrated Network Management, page 109-116, North-Holland, 1989

[72] Murrill B.:"OMNIPoint: An Implementation Guide to Integrated Networked Information Systems Management", in: Proceedings of the IFIP TC6/WG 6.6 Third International Symposium on Integrated Network Management, page 405-418, North-Holland, 1993

[73] Nakakawaji T., Katsuyama K., Miyauchi N., Mizuno T.: "MINT: an OSI Management Information Support Tool", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 845-855, North-Holland, 1991

[74] Network Management Forum: "Discovering Omnipoint - a common approach to the Integrated Management of Networked Information Systems", 1993

[75] Pras A.: "Network Management: an Alternative View", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 109-117, North-Holland, 1991

[76] Pras A.: "An Architecture for a Distributed IS-PABX", in Local Communication Systems: LAN and PBX II, Proceedings of the IFIP TC6/WG6.4 Second Int. Conf. on Local Communication Systems, Palma, Spain, June 1991

[77] Pras A.: "Netwerklaag-routering", in Handboek Telematica, Samson, the Netherlands, June 1992

[78] Pras A.: "TMN - Introduction and Interpretation", TIOS 92-16, Memoranda Informatica 92-40, University of Twente, The Netherlands, 1992

[79] Pras A., Togtema J.: "SNMPv2 at Twente University", The Simple Times, February 1994

[80] RACE CFS A150: "General Terminology", Common Functional Specifications, document 11, Issue B, December 1991

[81] RACE CFS H400-H411: "Network Management - Services", Common Functional Specifications, document 7, Issue B, December 1991

[82] RACE CFS H404: "Provisioning services in TMN", Common Functional Specifications, document 7, Issue B, December 1991

[83] RFC 768: "User Datagram Protocol", Postel, J.B., August 1980

[84] RFC 793: "Transmission Control Protocol", Postel J.B., 1981

[85] RFC 903: "Reverse Address Resolution Protocol", Finlayson R., Mann T., Mogul J.C., Theimer M., June 1984

[86] RFC 1021: "High-level Entity Management System (HEMS)", Partridge C., Trewitt G., October 1987

[87] RFC 1022: "High-level Entity Management Protocol (HEMP)", Partridge C., Trewitt G., October 1987

[88] RFC 1028: "Simple Gateway Monitoring Protocol", Davin J., Case J.D., Fedor M., Schoffstall M.L., November 1987

[89] RFC 1052: "IAB recommendations for the development of Internet network management standards", Cerf V.G., April 1988

[90] RFC 1155: "Structure and identification of Management Information for TCP/IP-based internets", Rose M.T.; McCloghrie K., May 1990

[91] RFC 1156: "Management Information Base for network management of TCP/IP-based internets", McCloghrie K.; Rose M.T., May 1990

[92] RFC 1157: "Simple Network Management Protocol (SNMP)", Case J.D., Fedor M., Schoffstall M.L., Davin C., May 1990

[93] RFC 1189: "Common Management Information Services and Protocols for the Internet (CMOT and CMIP)", Warrier U.S., Besaw L., LaBarre L., Handspicker B.D., October 1990

[94] RFC 1213: "Management Information Base for network management of TCP/IP-based internets: MIB-II", McCloghrie K., Rose M.T., March 1991

[95] RFC 1247: "OSPF version 2", Moy J., July 1991

[96] RFC 1253: "OSPF version 2: Management Information Base", Baker F., Coltun R., August 1991

[97] RFC 1441: "Introduction to version 2 of the Internet-standard Network Management Framework", Case J., McCloghrie K., Rose M., Waldbuster S., April 1993

[98] RFC 1447: "Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2)", McCloghrie K., Galvin J., April 1993

[99] RFC 1450: "Management Information Base for version 2 of the Simple Network Management Protocol (SNMPv2)", Case J., McCloghrie K., Rose M., Waldbuster S., April 1993

[100] RFC 1451: "Manager-to-Manager Management Information Base", Case J., McCloghrie K., Rose M., Waldbuster S., April 1993

[101] Rose M.T.: "The Simple Book - Second edition", Prentice-Hall International Editions, 1994

[102] Rose M.T., "Network management is Simple, you just need the right framework", in: Proceedings of the IFIP TC6/WG 6.6 Second International Symposium on Integrated Network Management, page 9-25, North-Holland, 1991

[103] Saal H.: "LAN Downtime: Clear and Present Danger", in Data Communications, 21 March 1991

[104] Schot J.: "The role of Architectural Semantics in the formal approach of Distributed Systems Design", Ph.D. thesis, University of Twente, 1990, ISBN 90-9004877-4

[105] Sinderen M.J. van: "On the design of application protocols", Ph.D. thesis, University of Twente, 1995, ISBN 90-365-0730-8

[106] Skubic J., Burwall H., "Service Management Architecture", in: Proceedings of the XIII International Switching Symposium, Vol. VI: page 155-160, May 1990

[107] Sloman M.S.: "Domain Management for Distributed Systems", in: Proceedings of the IFIP TC6/WG 6.6 Symposium on Integrated Network Management, page 505-516, North-Holland, 1989

[108] Sloman M., Twiddle K.: "Domains - A Framework for Structuring Management Policy", in: Network and Distributed Systems Management, page 433-453, Addison-Wesley Publishing Company, 1994

[109] Smith C., Milham D.J., Mulcahy C., "OSI Systems Management", in: BT Technology Journal, page 27-38, April 1990

[110] Stallings W.: "SNMP, SNMPv2 and CMIP - The Practical Guide to Network Management Standards", Addison Wesley

[111] Steege J.W. ter: "Alternatives for the OSI Systems Management Framework", Msc-thesis University of Twente, Enschede, The Netherlands, 1991

[112] Tucker J.: "A Common Approach to Managed Objects", in: Proceedings of the IFIP TC6/WG 6.6 Symposium on Integrated Network Management, page 159-165, North-Holland, 1989

[113] Vissers C.A., Ferreira Pires L., Quartel D.A.C.: "The Design of Telematics Systems", lecture notes, University of Twente, Department of Computer Science, Enschede, The Netherlands, October 1994

[114] Warrier U.S., Sunshine C.A., "A Platform for Heterogeneous Interconnection Network Management", in: Proceedings of the IFIP TC6/WG 6.6 Symposium on Integrated Network Management, page 13-24, North-Holland, 1989

[115] Westgate J.: "Technical Guide for OSI Management", NCC Blackell, 1992

[116] Worp S.B. van der: "Distributed Transactions in OSI Management", Msc-thesis University of Twente, Enschede, The Netherlands, 1993

[117] Yoshida M., Kobayashi M., Yamaguchi H., "Customer Control of Network Management from the Service Provider's Perspective", in: IEEE Communications Magazine, page 35-40, March 1990

Glossary

The terminology that is used within the network management community is poorly defined. This easily leads to misinterpretations.

This glossary includes a number of terms that have the potential to be misunderstood. For each of these terms the interpretation is given that is valid for this thesis. A number of terms are used in the sense of OSI: these terms are marked with OSI and enclosed between double quotes ("...").

Architecture

The complete set of *architectural concepts*, the rules to combine these concepts plus possibly *architectural models*.
The term architecture is also used in a more restricted sense to denote the functional structure of a distributed system as it appears to its users. In this sense the term architecture denotes the outcome of the architectural phase and is used in contrast to the terms *implementation* and *realization*.

Architectural concepts

The building blocks that are used to create architectural models. Examples are: entity, PDU, protocol, SAP, service, SP, system.

Architectural model

A specific structure of architectural concepts. An example is the OSI Reference Model.

Distributed system

The complete communication network, including all parts that perform communication functions.

Entity OSI

"An active element within a subsystem".
A single entity may be engaged in several *functions*.

Framework

The same as the first interpretation of *architecture*: the complete set of *architectural concepts*, the rules to combine these concepts plus possibly *architectural models*.

Function OSI

"A part of the activity of entities".

Implementation

The outcome of the implementation phase. An implementation unveils the internal structure of the network in terms of *(sub)systems* and protocols. The term implementation is used in contrast to the terms *architecture* and *realization*.

Primary functions
The term ‘primary’ functions is introduced in this thesis in contrast to ‘management’ functions. Primary functions may be considered as the ‘normal’ network functions that satisfy the primary user requirements. Management functions are the functions that ‘support’ and ‘control’ the primary functions.

Protocol OSI
"A set of rules and formats (semantic and syntactic) which determines the communication behaviour of *entities* in the performance of *functions*".

Realization
The outcome of the realization phase. A realization shows the internal structure of a single *system*. The term realization is used in contrast to the terms *architecture* and *implementation*.

Reference model
See *architectural model*.

Service OSI
"A capability of a layer and the layers beneath it, which is provided to entities at the next higher layer at the boundary between these layers."

Subsystem OSI
"An element in a hierarchical division of an open system which interacts directly only with elements in the next higher or the next lower division of that open system."
The intersection of a layer and system.

System OSI
A single node within the network. Examples are: routers and terminals.

Index

A

B

C

D

E

F

N

O

P

Q

R

S

T

U

V

W

X

Abbreviations

ACL - Access Control List
AFNOR - Association Française de Normalisation
API - Application Programming Interface
ASE - Application Service Element
AT&T - American Telegraph and Telephone company
ATM - Asynchronous Transfer Mode
BER - Basic Encoding Rules
BGP - Border Gateway Protocol
BT - British Telecom
CCITT - Comité Consultative Internationale de Telegraphique et Telephonique
CFS - Common Functional Specification
CLNP - ConnectionLess Network Protocol
CMIP - Common Management Information Protocol
CMIS - Common Management Information Service
CMOL - Common Management Over LLC
CMOT - Common Management Over TCP
CO - Command Oriented
CORBA - Common Object Request Broker Architecture
CPU - Central Processing Unit
CRC - Cyclic Redundancy Check
CRS - Configuration Report Server
DCF - Data Communication Function
DCN - Data Communication Network
DIS - Draft International Standard
DME - Distributed Management Environment
DP - Draft Proposal
DQDB - Distributed Queue Dual Bus
EGP - Exterior Gateway Protocol
ES - End System
ETSI - European Telecommunications Standards Institute
FCAPS - Fault, Configuration, Accounting, Performance and Security
FDDI - Fibre Distributed Data Interface
FTAM - File Transfer And Management
IAB - Internet Activities Board (until 1993 Internet Architecture Board)
IBC - Integrated Broadband Communication
IBCN - Integrated Broadband Communication Network
ICMP - Internet Control Message Protocol
IEC - International Electrotechnical Commission
IEEE - Institute of Electrical and Electronics Engineers
IETF - Internet Engineering Task Force
ILM - Intermediate Level Manager
IN - Intelligent Network
I/O - Input / Output
IP - Internet Protocol
IPC - Inter Process Communication
IPX - Internetwork Packet Exchange
ISDN - Integrated Services Digital Network

ISO - International Organization for Standardization
ITU - International Telecommunication Union
ITU-T - ITU Telecommunication Standardization Sector
JTC - Joint Technical Committee
LAN - Local Area Network
LAPB - Link Access Protocol Balanced
LLA - Logical Layered Architecture
LLC - Logical Link Control
LM - Layer Manager
MCF - Message Communication Function
MD - Mediation Device
ME - Managed Element
MF - Mediation Functions
MIB - Management Information Base
MIS - Management Information Service
MISP - Management Information Service Provider
NAU - Network Attachment Unit
NE - Network Element
NEF - Network Element Function
NFS - Network File System
NM Forum - Network Management Forum
NNI - Nederlands Normalisatie Instituut
OMG - Object Management Group
OMNI*Point* - Open Management Interoperability Point
ONA - Open Network Architecture
OO - Object Oriented
OS - Operations System
OSF - Open Software Foundation
OSF - Operations System Function
OSI - Open Systems Interconnection
OSPF - Open Shortest Path First
PC - Personal Computer
PDU - Protocol Data Unit
PF - Presentation Function
PSP - Primary Service Provider
QA - Q Adaptor
QAF - Q Adaptor Functions
QoS - Quality of Service
RACE - Research and development in Advanced Communications technologies in Europe
RARP - Reverse Address Resolution Protocol
REM - Ring Error Monitor
RFC - Request For Comment
RIP - Routing Information Protocol
RM - Reference Model
RPS - Ring Parameter Server
SAP - Service Access Point
SC - Sub-Committee
SG - Study Group
SGMP - Simple Gateway Monitoring Protocol

SIG - Special Interest Group
SMAE - Systems Management Application Entity
SMDS - Switched Multi-megabit Data Services
SMF - Systems Management Functions
SMI - Structure of Management Information
SMO - Systems Management Overview
SNA - Systems Network Architecture
SNMP - Simple Network Management Protocol
SONET - Synchronous Optical Network
SP - Service Primitive
TCP - Transmission Control Protocol
TG - Task Group
TLM - Top Level Manager
TMN - Telecommunications Management Network
TP - Transaction Processing
TR - Technical Report
TSAP - Transport Service Access Point
UDP - User Datagram Protocol
VAS - Value Added Service
VO - Variable Oriented
WG - Working Group
WS - Work Station
WSF - Work Station Functions
XMP - X/Open Management Protocol
XOM - X/Open OSI-Abstract-Data Manipulation

Samenvatting

Netwerkmanagement heeft tot taak de werking van communicatienetwerken te controleren en te verbeteren, alsmede het netwerk aan te passen in het geval de eisen van de netwerkgebruikers veranderen. Management betreft het initialiseren, observeren en modificeren van de netwerkfuncties. Voor het verrichten van management zijn speciale functies nodig, die in dit proefschrift 'managementfuncties' worden genoemd. Om managementfuncties te kunnen onderscheiden van de normale netwerkfuncties die de primaire gebruikerseisen realiseren, wordt in dit proefschrift eveneens de term 'primaire functies' geïntroduceerd.
Managementfuncties kunnen op een handmatige wijze worden uitgevoerd door personen (dit noemen we 'expliciet management'), maar ook op een geautomatiseerde wijze via hard- en software modules ('impliciet management'). In het geval managementfuncties handmatig worden uitgevoerd, zal het merendeel van deze functies worden verricht vanaf speciale managementsystemen die zich bevinden op een beperkt aantal lokaties. Indien managementfuncties automatisch worden uitgevoerd, kan het beter zijn een groot deel van deze functies over het hele netwerk te distribueren en te implementeren als onderdeel van de gemanagede systemen.

Architecturen voor netwerkmanagement bieden de ontwerper de mogelijkheid managementfuncties op een hoog abstractieniveau te beschouwen en een goed beeld te ontwikkelen van de te ontwerpen managementservices en -protocollen. In het kader van dit proefschrift wordt aangenomen dat dergelijke architecturen uit de volgende componenten bestaan:
- een verzameling architecturele concepten,
- regels die aangeven hoe deze concepten gebruikt moeten worden, en
- modellen die de toepassing van deze regels en concepten demonstreren en hierdoor het ontwerp van een specifieke klasse van systemen vereenvoudigen.

Alle bestaande managementarchitecturen, dus ook die van de ISO, ITU-T (voorheen CCITT) en de IETF, zijn ontwikkeld nadat het ontwerp van de primaire functies was afgerond. Een dergelijke aanpak getuigt van een specifieke zienswijze betreffende de rol van management en nodigt uit tot het toepassen van verschillende architecturele concepten voor het ontwerp van primaire functies enerzijds en het ontwerp van managementfuncties anderzijds. In dit proefschrift wordt een alternatieve aanpak voorgesteld, waarin geen principieel verschil wordt gemaakt tussen de ontwerpeisen voor primaire functies en de ontwerpeisen voor managementfuncties. Beide soorten eisen worden geïntegreerd in één enkel ontwerpproces dat gebruik maakt van één architectuurmodel.

Dit proefschrift bestaat uit twee gedeelten. In deel 1 (hoofdstuk 2 - 4) wordt de 'state of the art' van de drie belangrijkste netwerkmanagementarchitecturen

besproken. Deel II (hoofdstuk 5 - 9) laat zien hoe een alternatieve (geïntegreerde) netwerkmanagementarchitectuur ontwikkeld kan worden.

Hoofdstuk 2 analyseert de ISO-managementarchitectuur, zoals deze is vastgelegd in het 'OSI Management Framework' en de 'Systems Management Overview'. Het toont aan dat er, ondanks het vele onderzoek dat op het gebied van ISO-management is verricht, een aantal tekortkomingen van deze managementarchitectuur nog steeds niet is opgelost.

De managementarchitectuur van de ITU-T staat bekend als het 'Telecommunications Management Network' (TMN) en wordt in hoofdstuk 3 besproken. De naam van deze architectuur geeft reeds aan dat deze architectuur primair bedoeld is voor management van telecommunicatie- (b.v. telefonie) netwerken. In feite beschrijft TMN meerdere kleinere architecturen:

- een functionele architectuur,
- een fysieke architectuur,
- een informatie-architectuur, die veel ideeën van ISO management bevat,
- een architectuur van 'logische lagen' (logical layered architecture), inclusief een 'verantwoordelijkheidsmodel' (responsibility model).

Om de korte termijn managementproblemen van het Internet het hoofd te kunnen bieden, heeft de IETF in 1988 het 'Simple Network Management Protocol' (SNMP) gedefinieerd. Hoofdstuk 4 bespreekt deze Internet-managementaanpak. In tegenstelling tot de ISO en de ITU-T heeft de IETF geen afzonderlijk document geproduceerd waarin de gebruikte managementconcepten staan beschreven. De reden hiervoor is dat de IETF gebruik maakt van concepten die reeds beschreven zijn in (verouderde versies van) het 'OSI Management Framework' document, en deze concepten bovendien beschouwde als zijnde vanzelfsprekend. In 1992 begon de IETF met de ontwikkeling van een tweede versie van het SNMP protocol (SNMPv2). In dit protocol zijn een aantal nieuwe concepten geïntroduceerd die helaas nauwelijks worden uitgelegd en daarom moeilijk te begrijpen zijn.

Om managementfuncties beter te kunnen structureren wordt in hoofdstuk 5 een mogelijke classificatie van deze functies voorgesteld. Het hoofdstuk laat zien dat reeds tijdens de verschillende fases van het ontwerpproces over managementfuncties moet worden nagedacht.

In hoofdstuk 6 wordt uitgelegd hoe primaire en managementfuncties op een geïntegreerde wijze ontworpen kunnen worden. Er wordt aangetoond dat het niet altijd mogelijk is het ontwerp van alle managementfuncties af te ronden voordat de operationele fase begint. Dit hoeft geen probleem te zijn, mits er maatregelen worden genomen opdat de netwerkoperator de ontbrekende managementfuncties tijdens de operationele fase handmatig kan verrichten. De ontwerper dient intussen verder te gaan met de ontwikkeling van de overgebleven managementfuncties, zodat in toekomstige generaties van de net-

werksystemen deze functies op een automatische wijze kunnen worden uitgevoerd.

Een alternatieve managementarchitectuur, die zowel primaire als ook managementfuncties bevat, wordt in hoofdstuk 7 ontwikkeld. Dit hoofdstuk geeft een aantal voorbeelden die aantonen dat beide soorten functies gemodelleerd kunnen worden met behulp van de regels en concepten zoals die in het OSI Referentie Model zijn beschreven. Omdat er verschillende klassen van managementfuncties zijn en managementfuncties op diverse manieren gedistribueerd kunnen worden, worden in hoofdstuk 7 meerdere modellen ontwikkeld. Al deze modellen bevatten managementprotocollen, alsmede onderliggende service providers die de uitwisseling van managementinformatie ondersteunen. De managementprotocollen worden in hoofdstuk 9 besproken en ingedeeld in twee basistypes: 'variabel georiënteerd' en 'commando georiënteerd'. Hoofdstuk 8 bespreekt de service providers zoals die voor het vervoer van managementinformatie kunnen worden gebruikt.