CHAPTER X

CONCLUSION
Conclusion

This thesis motivates that in principle there is no difference between the design of primary network functions and the design of network management functions. Both kinds of functions are strongly interwoven and can be described in terms of a single set of architectural concepts and rules; an integrated architectural model can be developed which includes both kind of functions.

Such model shows the relationship between primary and management functions and may be used to construct management simulators (Chapter 7). The management architectures of the ISO, ITU-T and IETF do not describe management functions together with primary functions and do not provide integrated architectural models.

ISO's management architecture violates the idea of layering, by allowing management entities in the application layer to directly manipulate objects in the underlying service provider. Another deficiency of this architecture is that fault management may be problematic because the fault management functions implicitly rely for the exchange of their management information upon the correct behaviour of the managed functions (Chapter 2).

In TMN such implicit dependence between managed and management functions does not exist; to exchange management information TMN has defined a separate Data Communication Network (DCN). A deficiency of TMN is that its architectural concepts are not always properly defined (Chapter 3).

The IETF has not produced a separate standard to define its Internet management architecture. Instead, the IETF has concentrated upon the development of management protocols and MIBs. Many MIBs have already been defined, unfortunately no adequate structure has been proposed to organize the variables contained within these MIBs (Chapter 4).

This thesis demonstrates that primary functions can be developed together with management functions in a cyclic fashion. The first cycle(s) will generally be used to design the primary functions and identify which parts of these functions should be managed. To define the details of the management functions, subsequent cycles will be needed. The complexity of the management functions may be such, that also these management functions should be managed.

The designer may perform a number of additional design cycles to develop these 'management of management' (meta-management) functions (Chapter 6). Management functions can be structured by distinguishing the functions defined in one phase of the design process from functions defined in other phases. The following structure is proposed:
• Service management functions, for management functions introduced in the architectural phase.

• Protocol management functions, for management functions introduced in the implementation phase.

• Element management functions, for management functions introduced in the realization phase.

As opposed to the service and protocol management functions, element management functions are linked to practical realizations and manufacturer dependent. As a consequence, element management functions will not be standardized (Chapter 5).

Two basic types of management protocols can be identified: Variable Oriented (VO) and Command Oriented (CO). With VO management, the emphasis is on the management information within the managed systems. With CO management, the emphasis is on the management interaction between the managing and managed systems. VO management is better suited for centralized and explicit management, whereas CO management is better suited for distributed and implicit management. OSI's Object Oriented (OO) management protocol can be regarded as a mixture of VO and CO management (Chapter 9).

Further Research

It may be interesting to map the SNMPv2 protocol upon the architecture that has been described in this thesis. The purpose of such exercise should be to get a better understanding of SNMPv2. As a result, proposals can be presented to improve this protocol. Practical experience is needed to understand the applicability of the service management, protocol management and element management concept. Special MIBs should therefore be developed, as well as special management applications that take advantage of these MIBs.

It seems worthwhile to build a management simulator (Subsection 7.4.4). The effort to obtain a realistic prototype should not be neglected however, since such simulator requires as input formal descriptions of both primary as well as management functions.