

## 5. Summary and Conclusions

In this final chapter we bring together the work described in the previous chapters of this thesis. Section 5.1 explains the significance of this thesis in terms of the contributions to standards bodies and to the wider telecommunications management community. It also summarises the main findings. Section 5.2 identifies areas in which this work could be developed further.

### 5.1 The Contribution and Main Findings of This Thesis

The key contribution of this thesis was to explain and simplify the TMN architectural framework, to demonstrate its feasibility, implementability and performance when OSI-SM is used as the base technology and to show how distributed object technologies such as OMG CORBA can be used to replace OSI-SM in the future.

Given the fact that the relevant research work was undertaken over a long period of time, a number of the findings have already been published in research papers and contributed to international standards bodies and the wider telecommunications management community. The proposal for the integration of the OSI Directory in the TMN has found its way to the TMN architectural framework [M3010]. The precise details of using the OSI Directory for shared management knowledge and location transparency have been contributed to the Management Knowledge SMF [X750]. It was also proposed to the ITU-T Study Group 4 to remove the mediation function /  $q_x$  reference point and the  $f$  reference point from the TMN architecture. The proposal for removing the MF /  $q_x$  has been adopted in the latest 1997 draft of the TMN framework [M3010]. The proposal for not addressing the  $f$  reference point has not yet been adopted but almost no work exists to date behind its standardisation.

The above findings are of architectural nature and have been backed up by design, implementation and experimental results with real TMN platform infrastructure and prototype TMN systems. These experiments have shown that full scale OSI-SM / TMN technology is feasible, performant and economical to provide both in terms of computing resources and required development time. In addition to the architectural contributions to standards bodies, the OSIMIS TMN platform was made available to selected companies and research institutions as early as in 1991 i.e. version 2.95. Subsequent versions were made publicly available to the wider

telecommunications management community i.e. version 3.0 [Pav93a][Pav93b] and version 4.0 [Pav95b]. The relevant concepts, models, object-oriented design aspects and parts of the actual software itself were subsequently used in a number of commercial TMN offerings. Finally, the relevant models and APIs were provided as input to the NMF TMN/C++ API group at the end of 1995. Their recently proposed solution [Chat97] bears a lot of similarities with the concepts and object-oriented models detailed in Chapter 3 of this thesis.

The research work related to the use of CORBA as the base technology for the TMN was undertaken during the last two years, i.e. in 1996 and 1997. Though some of the results have already been published [Pav97b][Pav97d], the complete approach as presented in Chapter 4 has not yet been made publicly available. Given the growing interest in the use of distributed object technologies in TMN, the author intends to propose the relevant architecture and specification to standards bodies. This work will be specifically proposed to the OMG Telecommunications Special Interest Group; to the ITU-T SG 4 which addresses TMN evolution and its de-coupling from OSI-SM [M3010]; and to the evolving Open Distributed Management Architecture [X703] as an addendum on the potential use of CORBA.

Given the fact there is not yet a complete solution in the research community for replacing OSI-SM with CORBA in TMN environments, the author's proposal constitutes the major contribution of this thesis. It also justifies the fact that the thesis was submitted two years later than initially intended. The rest of the contributions on the TMN architectural framework and its validation through the design and implementation of the OSIMIS platform are equally important. It should be noted that it would have been impossible to propose the complete solution for the use of CORBA in TMN environments without the design and implementation experience from the OSI-SM based approach.

The research contributions of this thesis were summarised at the end of chapters 2, 3 and 4. We re-iterate through the main findings below.

The findings related to the TMN architecture are the following:

- OSI-SM is suitable technology for TMN systems because it satisfies a number of key TMN requirements: object-oriented information modelling aspects; intelligent information retrieval facilities; and fine-grain control event of event dissemination.
- OSI-SM does not support location transparency but it can be combined with the OSI Directory which can support name resolution for OSI-SM management applications. The key advantage of the OSI Directory is its federated nature which can support wide-area geographic distribution.
- There is no point in endorsing “less capable” TMN network elements through the existence of the lightweight  $Q_x$  interface: there are far too many possibilities for different  $Q_x$  interfaces which need specialised mediation devices. In addition, the cost of the full-scale  $Q_3$  interfaces is small as it was explained in Chapter 3 of this thesis.
- There is no need for lightweight TMN F interfaces to drive WS applications: the overhead of full-scale  $Q_3$  interfaces can be very easily accommodated by inexpensive laptop and desktop computers, which can provide views of telecommunications activity from diverse geographic locations. Given the prominence of the WWW as the ubiquitous network GUI, there is the possibility of encapsulating CMIS/P or CORBA IIOP messages in HTTP. There are various different ways of how this can be done and, as such, the TMN F interface should not be standardised.
- Given the previous findings, the TMN becomes a fairly simple architectural framework in which  $Q_3$  is the intra-TMN and X the inter-TMN interface. The key TMN characteristic becomes its layered hierarchical nature (element, network, service and business management) and the fact that specifications are produced only for inter-layer and inter-TMN interfaces, allowing flexibility for different realisations.

The findings related to the object-oriented realisation of the OSI-SM / TMN framework are the following:

- CMIS/P is a fairly straightforward protocol which needs only a connection-oriented reliable transport service and a presentation facility. Lightweight versions are also possible based on string encodings. A CMIS API can be used as a building block for higher-level infrastructures. These can hide its details and complexity and provide a development environment for managed and managing objects, in a similar fashion to emerging distributed object technologies such as OMG CORBA. A key ingredient for such infrastructures is an object-oriented polymorphic ASN.1 API.
- There exist two types of object-oriented manager infrastructures: those that model whole agents (the Remote MIB) and those that model individual managed objects (the Shadow MIB). Shadow objects may be generic, weakly-typed or specific, strongly-typed. In the latter case they may be enriched with behaviour which exploits the semantics of particular managed object classes. The RMIB and the strongly-typed SMIB are the best approaches. The SMIB approach entails a manager mapping of GDMO to O-O programming languages. It is also possible to map the RMIB and SMIB approaches onto scripting languages such as Tcl or, better, Java.
- Managed objects may be shielded from underlying protocol and service access details through generic object-oriented manager infrastructure which maps GDMO to O-O programming languages. Polymorphic methods may be used for the enrichment of automatically produced “stub” managed objects with behaviour. Aspects such as name resolution, scoping, filtering, atomicity, persistence, access control and event dissemination can be provided in a transparent fashion to managed object logic and behaviour. Finally, SMFs model resource independent generic aspects and can be supported in a fully re-usable fashion.
- TMN applications may be organised in a single or multi-threaded fashion. In the former case, asynchronous APIs should be used for remote invocations while sequential activities should be “conscious” about their duration. Long-lasting activities should be delegated to other operating system processes so that the “core” application remains responsive. The advent of kernel-based multithreading and relevant support in modern operating systems suggests that multi-threading is the way forward. On the other hand, concurrency control remains an issue that requires special support and care in complex applications.

- A detailed performance analysis of the OSIMIS platform, which validated the object-oriented ideas for the realisation of the OSI-SM / TMN framework, showed that full-scale TMN technology is relatively inexpensive in terms of computing resources and performant. The infrastructure overhead for a TMN OS is about 1.5 Mb at run-time, the data overhead for a simple MO is 0.5 Kb while an echo action takes around 10-11 msec on a LAN using a typical pair of UNIX workstations. Packet sizes are reasonable for management operations (in the order of 100 bytes / packet) but a number of packets are necessary for establishing a management association. The major performance overhead is due to the OSI protocol stack rather than the object-oriented application framework.

Finally, the findings related to the use of OMG CORBA as the base technology for the TMN are the following:

- When considering OSI-SM in the ODP framework, it is possible to consider an agent application as an ODP computational object which contains managed objects as ODP information objects. The whole agent becomes an engineering object that supports a CMIS-like computational interface. Managed objects become plain engineering objects without computational interfaces, accessed in a local manner by the agent. Until now, the OSI-SM to ODP mappings in the literature considered always managed objects as separate computational objects.
- ANSA and the OSF DCE failed to satisfy the following important requirements for telecommunications management: an object-oriented information and computational model with inheritance; support for the *any* type in their IDL so that dynamic, weakly-typed interfaces are possible; support for recursive structures so that CMIS-like filters can be expressed; and support for a dynamic invocation interface which is necessary for generic, semantic-free applications such as MIB browsers. In addition, both ANSA and DCE supported only C language APIs.
- CORBA is an ODP-influenced technology that has come of age. It supports interface inheritance, the *any* type, a dynamic invocation interface and multiple O-O programming language bindings. It still lacks though: full support for polymorphism through the redefinition of operations in derived interfaces; fine-grain control of events based on filtering; and asynchronous APIs.

- The JIDM GDMO to CORBA IDL mappings can be used as the basis for a native CORBA-based architecture. The problem is that the notion of attributes is lost which makes difficult to support “get all” attribute operations. In addition, in IDL attributes there is no automatic support for pretty-printing, comparison and subsequently filtering.
- According to pure ODP principles, managed objects should not exhibit hierarchical names and references to them should be discovered through the trader. This approach requires federated trading for non-hierarchical service spaces, which is as yet an unsolved problem.
- A more pragmatic approach is to retain the TMN hierarchical naming principles, organise managed objects in “agent” domains according to containment relationships and use the CORBA name services to get access to the “virtual MIT” root object. Managed objects could resolve names of subordinate objects to references. This is a minimal approach, with no multiple attribute access, scoping, filtering, multiple object access and fine-grain event management.
- The missing OSI-SM functionality may be added through an “attribute repository” that can be used to support multiple attribute access and filtering, the last through hand-coded comparison methods. Support for filtering makes possible to provide OSI-SM-like event management facilities through an Event Processing object and EFD / log objects. Multiple object discovery and access through scoping and filtering may be provided by a Management Broker object, in an orthogonal fashion to managed objects.
- Migration from OSI-SM to a CORBA-based TMN can be handled gracefully by implementing management brokers at various stages: first as CORBA to OSI-SM gateways, then as dual CORBA and OSI-SM agents, subsequently as CORBA agents and finally as proper brokers for MOs with native CORBA IDL interfaces. This approach will retain investment on OSI-SM-based TMN technology for as long as necessary.
- A performance evaluation of the CORBA-based framework using a commercial CORBA platform showed very good access times, i.e. 4-5 msec for an echo action through a native IDL interface on a LAN using a typical pair of workstations. On the other hand, the data overhead for a managed object instance and the smallest application size were at least double those of OSIMIS. Finally, packet sizes are comparable to the OSI-SM ones but become much bigger when interoperable object references are communicated.

- The advantage of using CORBA is not so much its slightly better performance but its better distribution paradigm, the software portability and the possibility that it may become the ubiquitous distributed object technology. In the latter case, its use in the TMN may result in economies of scale.
- The use of CORBA does not require any changes to the TMN architectural framework. CORBA protocols will be endorsed as valid options for the Q<sub>3</sub> and X interfaces. Information model specifications will still be in GDMO while the use of CORBA or OSI-SM will become an engineering decision. Interoperability between the two will be supported through gateways in both directions.

It is finally worth mentioning that if CORBA with C++ language bindings was available around 1992-93, the author would have used it as the basis for an object-oriented TMN platform. The fact that ODP-influenced technology at the time had a number of limitations (i.e. ANSA, DCE) led the author to the design and implementation of the OSI-SM based TMN platform. This was essentially a “management-oriented DPE” and the advent of technologies like CORBA justified this initial vision and direction. The existence of an OSI-SM based TMN platform allowed research and experimentation with TMN systems in the first half of the nineties, which would have otherwise been impossible.

## 5.2 Potential Future Work

There are various ways in which this work can be taken further, we summarise potential future directions below.

- The Lightweight CMIP approach [Pav95c] was never more than a specification document. The recent success of the equivalent Lightweight Directory Access Protocol [LDAP], on which LCMIP is based, and the management needs of mobile network environments suggest that this approach could be taken further. An implementation and subsequent detailed performance measurements and comparison to the full Q<sub>3</sub> approach is necessary in order to quantify potential advantages.
- The OSI-SM performance measurements in section 3.8 of Chapter 3 represent only a first approach towards a performance analysis of OSI upper layer protocols. The increasing popularity of the current TMN approach in which OSI-SM is used as the base technology suggests that a more detailed analysis would be welcome. This could highlight those aspects of the upper layer stack that cause the overhead and propose subsequent optimisations. The existence of the full source code of OSIMIS / ISODE makes possible to undertake such a performance study.
- In the same fashion, the performance analysis of the CORBA-based framework in section 4.5 of Chapter 5 only scratched the surface of the problem. A detailed performance analysis is necessary in order to attribute the overheads to various parts of the system and propose optimisations. This work though requires access to the source code of a CORBA platform.
- The specific, strongly-typed Shadow MIB approach is pertinent to OSI-SM based TMN platforms, e.g. OSIMIS, NMF TMN/C++, and does not exist in CORBA. Its key advantage is that shadow objects can be enhanced with class-specific behaviour for caching, adaptive polling, etc. It would be interesting to see how such an approach can be provided over the CORBA strongly-typed client API. This work will require access to an IDL compiler and may result in changing the IDL to O-O language mappings.
- Federated naming services in CORBA can be provided in a similar fashion to the OSI Directory due to the hierarchical nature of the name space. Federated trading and federated event services though are much more difficult to provide. Research is necessary to address the optimal provision of such federated services.

- Management is impossible without security. While management issues have been addressed in OSI-SM environments, it is interesting to investigate how similar facilities can be supported in CORBA-based environments. A particularly interesting issue is the provision of access control. This can be relatively easily provided for agent-administered managed objects but becomes a much more difficult proposition for managed objects with native CORBA IDL interfaces. A related issue is the scalability and performance of the relevant solution.
- The advent of languages like Java that can support code mobility opens up new possibilities for management. The impact of this new paradigm to TMN is a whole new area of interesting research, pointing to “programmable” network elements with very simple native interfaces and an environment that can host mobile software entities.
- The TMN was developed as a framework addressing only the operation, administration and maintenance activities of telecommunications networks. It has become clear though that a more general framework is necessary for the integration of all distributed telecommunications software, including both management and service control. The evolution of the TMN together with other related frameworks such as IN and TINA towards a unifying framework is an interesting research issue.
- One of the main research interests of the author lies in performance and quality of service management for multi-service telecommunications networks. The existence of environments like the one presented in this thesis will make possible to address this problem with experimental management systems, in addition to analytical studies and simulation.

