ETSI TS 132 102 V5.1.0 (2002-09)

Technical Specification

Universal Mobile Telecommunications System (UMTS);
3G Telecom Management Architecture
(3GPP TS 32.102 version 5.1.0 Release 5)

Reference
RTS/TSGS-0532102v510

Keywords
UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, send your comment to:
editor@etsi.fr

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under www.etsi.org/key .

Contents

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document identifies and standardises the most important and strategic contexts in the physical architecture for the management of UMTS. It serves as a framework to help define a telecom management physical architecture for a planned UMTS and to adopt standards and provide products that are easy to integrate.

The requirements identified in this document are applicable to all further development of UMTS Telecom Management specifications as well as the development of UMTS Management products. This document can be seen as guidance for the development of all other Technical Specification addressing the management of UMTS, except 3GPP TS 32.101 [2].

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] ITU-T Recommendation M.3010 (2000): "Principles for a telecommunications management network".

[2] 3GPP TS 32.101: "3G Telecom Management principles and high level requirements".

[3] Void

[4] ITU-T Recommendation X.200 (1994): "Information technology – Open Systems Interconnection – Basic reference model: The basic model".

[5] Void

[6] Void

[7] Void

[8] Void

[9] TMF GB910. Smart TMN Telecom Operations Map (Release 1.1).

[10] TMF GB909. Smart TMN Technology Integration Map (Issue 1.1).

[11] ITU-T Recommendation M.3013 (2000): "Considerations for a telecommunications management network".

[12] 3GPP TS 23.002: "Network architecture".

[13] 3GPP TS 23.101: "General UMTS Architecture".

[14] 3GPP TS 32.111: "3G Fault Management".

[15] OMG "Unified Modelling Language Specification, Version 1.4, September 2001".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following definitions apply:

Architecture: The organisational structure of a system or component, their relationships, and the principles and guidelines governing their design and evolution over time.

Closed interfaces: Privately controlled system/subsystem boundary descriptions that are not disclosed to the public or are unique to a single supplier.

De facto standard: A standard that is widely accepted and used but that lacks formal approval by a recognised standards organisation.

Information Object : defined in 3GPP TS 32.101 [2].

Information Service: defined in 3GPP TS 32.101 [2].

Interface standard: A standard that specifies the physical or functional interface characteristics of systems, subsystems, equipment, assemblies, components, items or parts to permit interchangeability, interconnection, interoperability, compatibility, or communications.

Interoperability: The ability of two or more systems or components to exchange data and use information.

Intra-operability: The ability to interchange and use information, functions and services among components within a system.

IRPAgent: The IRPAgent encapsulates a well-defined subset of network (element) functions. It interacts with IRPManagers using an IRP. From the IRPManager's perspective, the IRPAgent behaviour is only visible via the IRP.

IRPManager: The IRPManager models a user of the IRPAgent and it interacts directly with the IRPAgent using the IRP. From the IRPAgent perspective, the IRPManager behaviour is only visible via the IRP.

IRP Information Model: defined in 3GPP TS 32.101 [2].

IRP Information Service: defined in 3GPP TS 32.101 [2].

IRP Solution Set: defined in 3GPP TS 32.101 [2].

IRPManager: The IRPManager models a user of the IRPAgent and it interacts directly with the IRPAgent using the IRP. Since the IRPManager represents an IRPAgent user, they help delimit the IRPAgent and give a clear picture of what the IRPAgent is supposed to do. From the IRPAgent perspective, the IRPManager behaviour is only visible via the IRP.

IRP Information Model: An IRP Information Model consists of an IRP Information Service and a Network Resource Model (see below for definitions of IRP Information Service and Network Resource Model).

IRP Information Service: An IRP Information Service describes the information flow and support objects for a certain functional area, e.g. the alarm information service in the Fault Management area. As an example of support objects, for the Alarm IRP there is the "alarm information" and "alarm list".

IRP Solution Set: An IRP Solution Set is a mapping of the IRP Information Service to one of several technologies (CORBA/IDL, SNMP/SMI, CMIP/GDMO etc.). An IRP Information Service can be mapped to several different IRP Solution Sets. Different technology selections may be done for different IRPs.

Managed Object : defined in 3GPP TS 32.101 [2].

Management Infrastructure: The collection of systems (computers and telecommunications) a UMTS Organisation has in order to manage UMTS.

Market Acceptance: Market acceptance means that an item has been accepted in the market as evidenced by annual sales, length of time available for sale, and after-sale support capability.

Modular: Pertaining to the design concept in which interchangeable units are employed to create a functional end product.

Module: An interchangeable item that contains components. In computer programming, a program unit that is discrete and identifiable with respect to compiling, combining with other modules, and loading is called a module.

Network Resource Model (NRM): defined in 3GPP TS 32.101 [2]..

Open Specifications: Public specifications that are maintained by an open, public consensus process to accommodate new technologies over time and that are consistent with international standards.

Open Standards: Widely accepted and supported standards set by recognised standards organisation or the commercial market place. These standards support interoperability, portability, and scalability and are equally available to the general public at no cost or with a moderate license fee.

Open Systems Strategy: An open systems strategy focuses on fielding superior telecom capability more quickly and more affordably by using multiple suppliers and commercially supported practices, products, specifications, and standards, which are selected based on performance, cost, industry acceptance, long term availability and supportability, and upgrade potential.

Physical Architecture: A minimal set of rules governing the arrangement, interaction, and interdependence of the parts or elements whose purpose is to ensure that a conformant system satisfies a specified set of requirements. The physical architecture identifies the services, interfaces, standards, and their relationships. It provides the technical guidelines for implementation of systems upon which engineering specifications are based and common building blocks are built.

Plug&play: Term for easy integration of HW/SW.

Portability: The ease with which a system, component, data, or user can be transferred from one hardware or software environment to another.

Proprietary Specifications: Specifications, which are exclusively owned by a private individual or corporation under a trademark or patent, the use of which would require a license.

Reference Model: A generally accepted abstract representation that allows users to focus on establishing definitions, building common understandings and identifying issues for resolution. For TMN Systems acquisitions, a reference model is necessary to establish a context for understanding how the disparate technologies and standards required to implement TMN relate to each other. A reference model provides a mechanism for identifying the key issues associated with applications portability, modularity, scalability and interoperability. Most importantly, Reference Models will aid in the evaluation and analysis of domain-specific architectures.

Scalability: The capability to adapt hardware or software to accommodate changing workloads.

Service Specific Entities: Entities dedicated to the provisioning of a given (set of) service(s). The fact that they are implemented or not in a given PLMN should have limited impact on all the other entities of the PLMN.

Solution Set: defined in 3GPP TS 32.101 [2].

Specification: A document that prescribes, in a complete, precise, verifiable manner, the requirements, design, behaviour, or characteristics of a system or system component.

Standard: A document that establishes uniform engineering and technical requirements for processes, procedures, practices, and methods. Standards may also establish requirements for selection, application, and design criteria of material.

Standards Based Architecture: An architecture based on an acceptable set of open standards governing the arrangement, interaction, and interdependence of the parts or elements that together may be used to form a TMN System, and whose purpose is to insure that a conformant system satisfies a specified set of requirements.

Support object : defined in 3GPP TS 32.101 [2].

System : Any organised assembly of resources and procedures united and regulated by interaction or interdependence to accomplish a set of specific functions.

System Architecture (SA): A description, including graphics, of systems and interconnections providing for or supporting management functions. The SA defines the physical connection, location, and identification of the key nodes, circuits, networks, platforms, etc., and specifies system and component performance parameters. It is constructed to satisfy Operational Architecture requirements per standards defined in the Physical Architecture. The SA shows how multiple systems within a subject area link and inter-operate, and may describe the internal construction or operations of particular systems within the architecture.

UMTS Organisation: A legal entity that is involved in the provisioning of UMTS.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

3G	3rd Generation
AN	Access Network
AS	Application Server
ATM	Asynchronous Transfer Mode
AUC	Authentication Centre
BG	Border Gateway
BGCF	Breakout Gateway Control Function
BSC	Base Station Controller
BSS	Base Station Subsystem
BTS	Base Transceiver Station
CAMEL	Customised Applications for Mobile network Enhanced Logic
CBC	Cell Broadcast Center
CBS	Cell Broadcast Service
CIM	Common Information Model Specification (from DMTF)
CMIP	Common Management Information Protocol
CMIS	Common Management Information Service
CMISE	Common Management Information Service Element
CN	Core Network
CS	Circuit Switched
CORBA	Common Object Request Broker Architecture
CSCF	Call Session Control Function
DCN	Data Communication Network
DECT	Digital Enhanced Cordless Telecommunications
DSS1	Digital Subscriber System 1
EIR	Equipment Identity Register
E-OS	Element Management Layer-Operations System
F/W	Firewall
FM	Fault Management
FTAM	File Transfer, Access and Management
GCR	Group Call Register
GDMO	Guidelines for the Definition of Managed Objects
GGSN	Gateway GPRS Support Node
GMLC	Gateway Mobile Location Center
GMSC	Gateway MSC
GPRS	General Packet Radio Service
GTT	Global Text Telephony
HLR	Home Location Register
HSS	Home Subscriber Server
HTTP	HyperText Transfer Protocol
HW	Hardware
I-CSCF	Interrogating CSCF
IDL	Interface Definition Language
IIOP	Internet Inter-ORB Protocol
IM	Information Model
IM-MGW	IP Multimedia Media Gateway
IMS	IP Multimedia Subsystem
INAP	Intelligent Network Application Part
IP	Internet Protocol
IRP	Integration Reference Point
IS	Information Service
ISDN	Integrated Services Digital Network

IWU	Inter Working Unit
LCS	Location Services
LMU	Location Measurement Unit
MD	Mediation Device
ME	Mobile Equipment
MGCF	Media Gateway Control Function
MIB	Management Information Base
MMI	Man-Machine Interface
MML	Man-Machine Language
MMS	Multimedia Messaging Service
MNP	Mobile Number Portability
MNP-SRF	Mobile Number Portability/Signalling Relay Function
MRF	Multimedia Resource Function
MRFC	Multimedia Resource Function Controller
MRFP	Multimedia Resource Function Processor
MSC	Mobile service Switching Centre
MT	Mobile Termination
NE	Network Element
N-OS	Network Management Layer-Operations System
NPDB	Number Portability Database
NR	Network Resource
NRM	Network Resource Model
NSS	Network Switching Subsystem
NW	Network
OMG	Object Management Group
OS	Operations System
OSA	Open Services Access
OSF	Operations System Functions
P-CSCF	Proxy CSCF
PDH	Plesiochronous Digital Hierarchy
PS	Packet Switched
PSA	Product Specific Applications
PSS	Packet Switched Service
PSTN	Public Switched Telephone Network
QA	Q-Adapter
QoS	Quality of Service
RNC	Radio Network Controller
RNS	Radio Network System
RSVP	Resource ReserVation Protocol
S-CSCF	Serving CSCF
SDH	Synchronous Digital Hierarchy
SGSN	Serving GPRS Support Node
SGW	Signalling Gateway
SLA	Service Level Agreement
SLF	Subscription Locator Function
SIM	Subscriber Identity Module
SMLC	Serving Mobile Location Center
SMI	Structure of Management Information
SMS	Short Message Service
SNM	Sub-Network Manager
SNMP	Simple Network Management Protocol
SS	Solution Set
SS7	Signalling System No. 7
SW	Software
TA	Terminal Adapter
TE	Terminal Equipment
TM	Telecom Management
TMN	Telecommunications Management Network as defined in ITU-T Recommendation M.3010 [1].
UE	User Equipment
UML	Unified Modelling Language
UMTS	Universal Mobile Telecommunications System
USAT	USIM/SIM Application Toolkit

USIM	UMTS Subscriber Identity Module
UTRA	Universal Terrestrial Radio Access
UTRAN	Universal Terrestrial Radio Access Network
VHE	Virtual Home Environment
VLR	Visitor Location Register
WBEM	Web Based Enterprise Management
WS	Workstation

4 General

4.1 UMTS

4.1.1 UMTS Reference Model

A Universal Mobile Telecommunications System is made of the following components:

- one or more Access Networks, using different types of access techniques (GSM, UTRA, DECT, PSTN, ISDN,...) of which at least one is UTRA;
- one or more Core Networks;
- one or more Intelligent Node Networks, service logic and mobility management, (IN, GSM...);
- one or more transmission networks (PDH, SDH etc) in various topologies (point-to-point, ring, point-to-multipoint etc) and physical means (radio, fibre, copper etc).

The UMTS components have signalling mechanisms among them (DSS1, INAP, MAP, SS7, RSVP etc.).

From the service perspective, the UMTS is defined to offer:

- service support transparent to the location, access technique and core network, within the bearer capabilities available in one particular case;
- user to terminal and user to network interface (MMI) irrespective of the entities supporting the services required (VHE);
- multimedia capabilities.

4.1.2 UMTS Provisioning Entities

Two major entities, which cover the set of UMTS functionalities involved in the provision of the UMTS services to the user, are identified as follows:

Home Environment. This entity holds the functionalities that enable a user to obtain UMTS services in a consistent manner regardless of the user's location or the terminal used.

Serving Network. This entity provides the user with access to the services of the Home Environment.

4.1.3 UMTS Management Infrastructure

Every UMTS Organisation has it's own Management Infrastructure. Each Management Infrastructure will contain different functionality depending on the role-played and the equipment used by that UMTS Entity.

However, the core management architecture of the UMTS Organisation is very similar. Every UMTS Organisation:

- provides services to it's customers;
- needs an infrastructure to fulfil them (advertise, ordering, creation, provisioning,...);
- assures them (Operation, Quality of Service, Trouble Reporting and Fixing,...);
- bills them (Rating, Discounting,...).

Not every UMTS Organisation will implement the complete Management Architecture and related Processes. Some processes may be missing dependent on the role a particular UMTS Organisation is embodying. Processes not implemented by a particular UMTS Organisation are accessed via interconnections to other UMTS organisations, which have implemented these processes (called X-interfaces in the TMN architecture).

The Management architecture itself does not distinguish between external and internal interfaces.

4.2 TMN

TMN (Telecommunications Management Network), as defined in [1], provides:

- an architecture, made of OS (Operations Systems) and NEs (Network Elements), and the interfaces between them (Q, within one Operator Domain and X, between different Operators);
- the methodology to define those interfaces;
- other architectural tools such as LLA (Logical Layered Architecture) that help to further refine and define the Management Architecture of a given management area;
- a number of generic and/or common management functions to be specialised/applied to various and specific TMN interfaces.

The UMTS Management Architecture is largely based on TMN, and will reuse those functions, methods and interfaces already defined (or being defined) that are suitable to the management needs of UMTS. However, the UMTS Management needs to explore the incorporation of other concepts (other management paradigms widely accepted and deployed) for the new challenges UMTS faces.

5 General view of UMTS Management Physical architectures

Telecom Management Architectures can vary greatly in scope and detail. The architecture for a large service provider, with a lot of existing legacy systems and applications, upon which many services are based, will be of high complexity. In contrast, the architectural needs of a start-up mobile operator providing its services to a small group of value added Service Providers will be much less and will probably focus on more short-term needs.

A mobile network operator has to manage many different types of networks as radio networks, exchanges, transmission networks, area networks, intelligent nodes and substantial amounts of computer hardware/software. This wide variety of network equipment will most probably be obtained from a variety of equipment vendors. The nature of a mobile radio network will be heterogeneous and will present a number of operational difficulties for the service provider on enabling effective and efficient network management.

The standardisation work for the management of UMTS has adopted the top-down approach and will from business needs identify functional and informational architectures. The physical architecture will have to meet these requirements and as there are many ways to build a UMTS it will vary greatly from one TMN solution to another. There will be many physical implementations, as different entities will take different roles in a UMTS.

It is obvious that it will not be meaningful or even possible to fully standardise a common Telecom Management physical architecture for UMTS. This document will identify and standardise the most important and strategic contexts and serve as a framework to help define a physical architecture for a planned UMTS.

6 Basic objectives for a UMTS Physical Architecture

The management of UMTS will put a lot of new requirements to the management systems compared to the second generation of Mobile telephony. Some of the challenging requirements affecting the physical architecture are:

- To be capable of managing equipment supplied by different vendors.
- To enable TM automation in a more cost efficient way - TM optimised for maximum efficiency and effectiveness.
- To provide UMTS configuration capabilities that are flexible enough to allow rapid deployment of services.
- To report events and reactions in a common way in order to allow remote control.
- To allow interoperability between Network Operators/Service Providers for the exchange of management/charging information.
- To be scaleable and applicable to both larger and small deployments.
- Accessibility to information.
- To profit from advances and standards in IT and datacom industry.

The second generation of mobile networks can - from management point of view - be characterised as the era of net-element vendor-dependent NE managers. The different OSs had very low interoperability with other systems and functional blocks could rarely be re-used. The Mobile Telecom Management Networks were far away from the TMN vision where one vendor's OS should be able to manage other vendor's net elements.

For UMTS Management it is clearly stated the necessity of cost-effective solutions and better time to market focus. Interoperability, scalability and re-use are keywords for the new generation of management systems.

Many of the new requirements on the management of UMTS can only be solved by defining and establish a suitable physical architecture. Thou it is not possible to standardise the one single UMTS TM physical architecture, it is evidently so that the success of a Telecom Management Network of a UMTS will heavily depend on critical physical architectural issues. This document will identify those architectural critical issues.

7 TM Architectural aspects

7.1 Architectural relationship

The basic aspects of a TM architecture, which can be, considered when planning and designing a TM network are:

- The functional architecture.
- The information architecture.
- The physical architecture.

The management requirements - from the business needs - will be the base for the functional architecture, which will describe the functions that have to be achieved. The information architecture defines what information that has to be provided so the functions defined in the functional architecture can be achieved. The physical architecture has to meet both the functional architecture and the information architectures. Other constraints from realty will also have impact to the physical architecture as cost, performance, legacy systems and all preferences any operator will have on a big capital investment as a TM network.

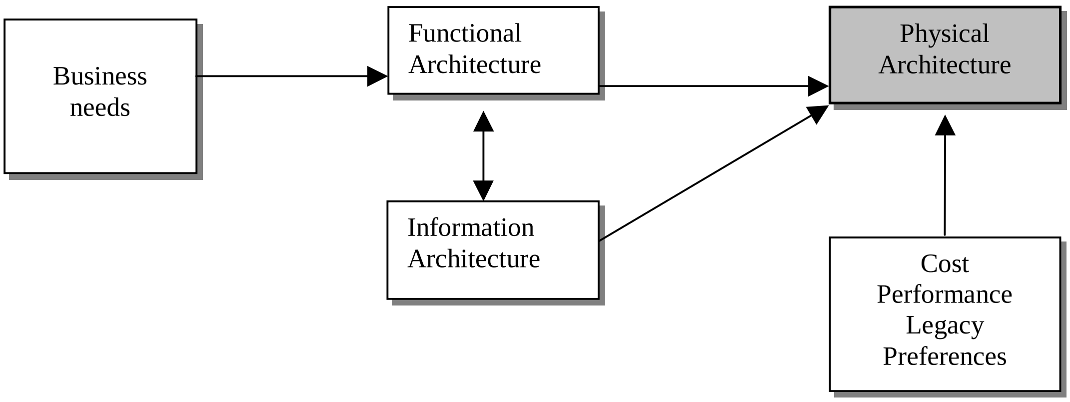


Figure 7.1: Architectural relationship

7.2 Architectural constraints

Large software systems, such as a network management system, are a capital investment that operators cannot afford to scrap every time its requirements change. Network operators are seeking cost-effective solutions to their short-term needs. All these reality-related issues are vital constraints that should be addressed in the definition of the architecture.

The standardisation of UMTS will bring new and different services that will add new demands on network management. Every UMTS organisation will include different functionality depending on the role-played and the equipment used by that UMTS entity. Regulation may force some of the roles that shall be taken. The need to link systems across corporate boundaries will be a consequence of this.

The rapid evolution of new services and technologies will also put requirements on the UMTS physical management architecture to accommodate market and technology trends. To future-proof investments and continuously be able to take advantage of new technologies are important constraints to the physical architecture.

A UMTS TMN should also adopt an architecture that will achieve scalability and extensibility of systems and networks so the TMN can grow as the services expand over time. To start with a small TMN and easily be able to expand the TMN after new requirements will be important issues for most UMTS operators.

The Telecom Management Network will be just one part of the overall business of a company. System management, general security issues and development strategies can be the target for company policies. System architectures and technology choices, as well as the availability of off-the-shelf commercial systems and software components that fulfil the requirements established in this specification, may be critical to an operator's implementation of the specified UMTS management architecture.

7.3 Interoperability

7.3.1 Introduction

The new requirement on a UMTS TMN will imply a focus change from net element management towards management of information "information management". Network providers make use of different information in several different ways which also may vary from network to network and from time to time. Basic information as alarms is of course essential information for localising faults but may also be the key information to be able to set up a service with a service level agreement.

Numerous of different interfaces can be identified in a UMTS network in the areas of network element management, network management and service management. The most important and complex of these interfaces will be standardised but many interfaces of less importance are unlikely to be fully standardised and will be up to the individual operator and vendor to develop. To adopt mainstream computing technologies, re-use widely used protocols, standards and an open system architecture will be essential to secure interworking between all physical entities in a UMTS.

Low-cost and general access to management systems information will be needed. Obviously this is the critical issue and challenging task in the heterogeneous, distributed and complex network of a UMTS.

7.3.2 Interfaces

A UMTS will consist of many different types of components based on different types of technologies. There will be access-, core-, transmission- and service node networks and many of the UMTS components have already been the targets for Telecom Management standardisation at different levels. Many of these standards will be reused and the management domain of a UMTS will thereby consist of many TMNs. The architecture of UMTS TMNs should support distributed TMNs and TMN-interworking on peer-to-peer basis.

The Telecom Management Architecture can vary greatly in scope and detail, because of scale of operation and that different organisations may take different roles in a UMTS (see clause 5). The architecture of UMTS TMNs should provide a high degree of flexibility to meet the various topological conditions as the physical distribution and the number of NEs. Flexibility is also required to allow high degree of centralisation of personnel and the administrative practices as well as allowing dispersion to administrative domains (see further clause 10). The 3G Telecom Management architecture should be such that the NEs will operate in the same way, independently of the OS architecture.

Figure 7.2 illustrates the basic domains in UMTS (identified in 3GPP Technical Standards [12], [13]), related management functional areas and introduces Interface-N (Itf-N).

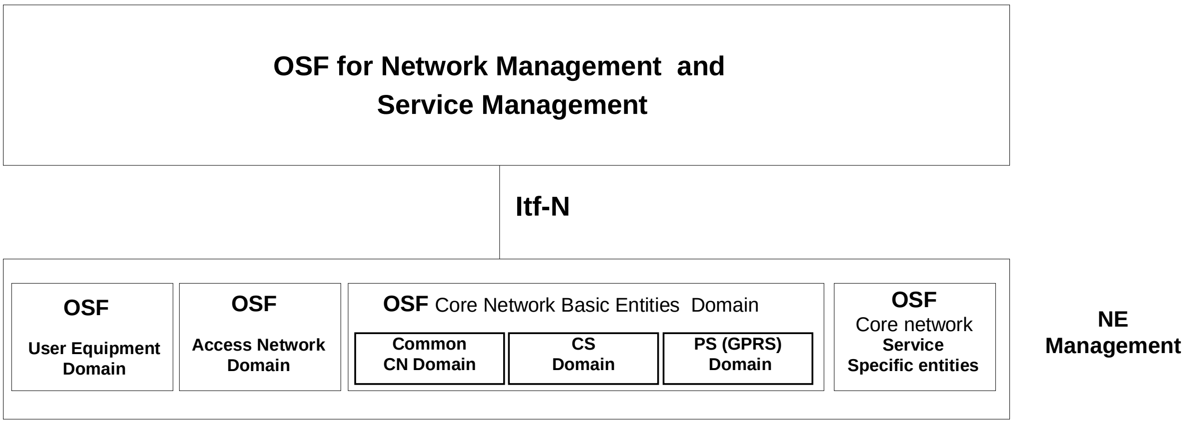


Figure 7.2: Overview of UMTS Telecom Management Domains and Itf-N

Itf-N between the NE OSFs and NM/SM OSFs could be used by the network- and service management systems to transfer management messages, notifications and service management requests via the NE OSF to the Network Elements (NEs).

This interface shall be open and the information models standardised.

Telecom management interfaces may be considered from two perspectives:

1. the management information model;
2. the management information exchange.

The management information models will be standardised in other 3GPP documents but the management information exchange will be further described in this architectural standard.

The management task will vary greatly between different network elements in a UMTS. Some NEs are of high complexity e.g. a RNC, while others e.g. a border gateway is of less complexity. Different application protocols can be chosen to best suite the management requirements of the different Network Elements and the technology used.

Application protocols can be categorised out of many capabilities as:

- Functionality;
- Implementation complexity;
- Processor requirements;
- Cost efficiency;

- Market acceptance, availability of "off the shelf commercial systems and software".

For each Telecom Management interface that will be standardised by 3GPP at least one of the accepted protocols will be recommended. Accepted application protocols (e.g. CMIP, SNMP, CORBA IIOP) are defined in 3GPP TS 32.101 [2], Annex A.

7.3.3 Entities of a UMTS

To provide the mobile service as it is defined in a UMTS, some specific functions are introduced [12]. These functional entities can be implemented in different physical equipments or gathered. In any case, exchanges of data occur between these entities and from the Telecom Management perspective they can all normally be treated as network elements of a UMTS. The basic telecom management functional areas as fault management, configuration management, performance management and security management are all applicable to these UMTS entities. As such they are all the targets for UMTS Telecom Management technical standards.

As discussed in clause 5, there will be many possible ways to build a UMTS and thereby many possible architectures of a mobile system. The entities presented in figure 7.3 should be treated as the fundamental building blocks of any possible implementation of a UMTS.

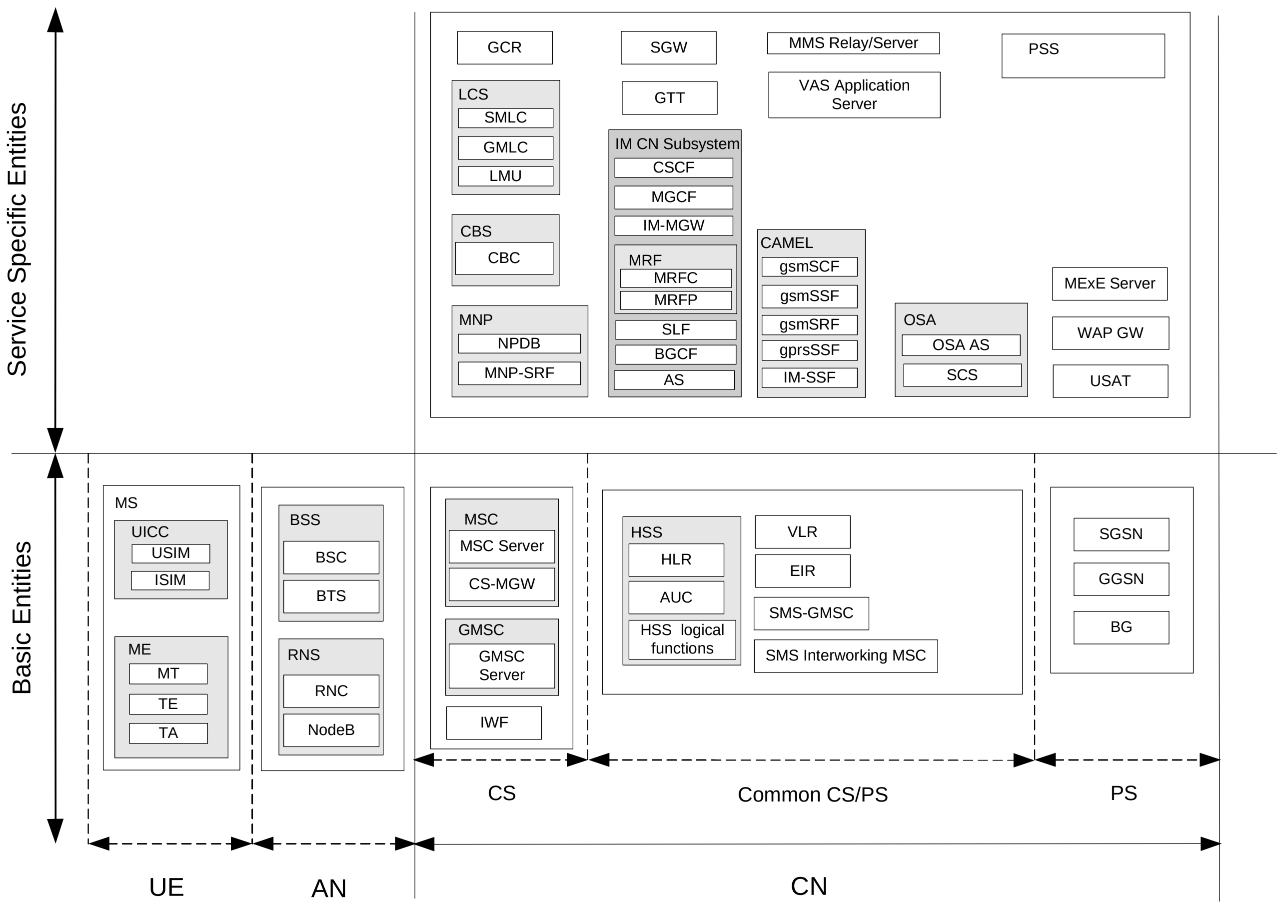


Figure 7.3 Examples of entities of the mobile system to be managed

In figure 7.4 the prime domains for the standardisation effort of 3GPP Telecom Management are shown as shaded.

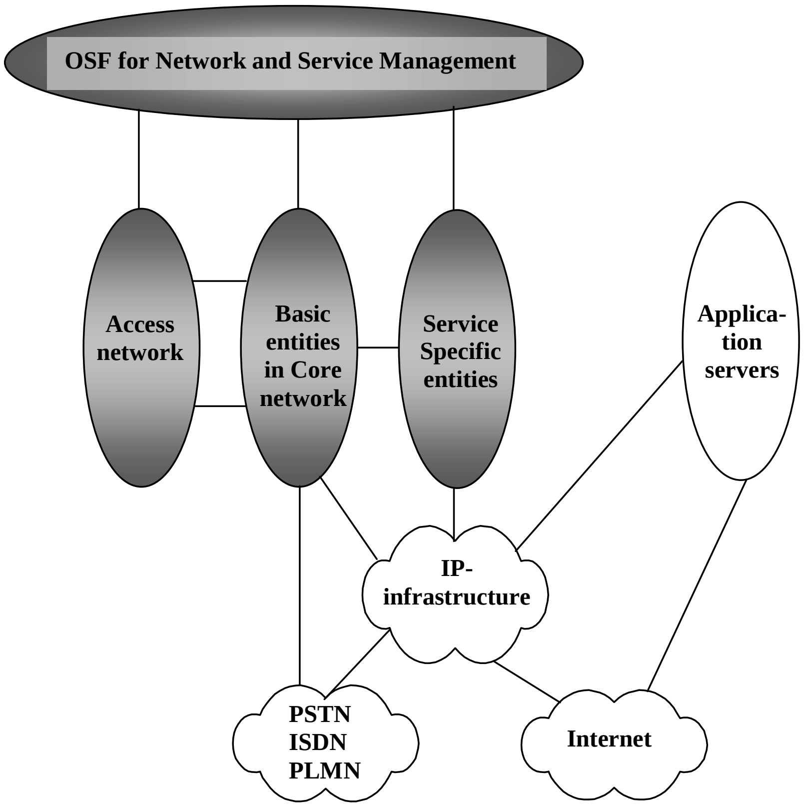


Figure 7.4 : High level UMTS Network architecture

7.3.4 Open systems approach

Even in the second generation of mobile radio networks the operators has to cope with heterogeneous environments in many different ways. No single vendor is likely to deliver all the management systems needed for a mobile operator.

The many different types of network elements, some with very high management complexity as an exchange and some less complex as a repeater system, are generally supported with unique vendor specific management systems with very low interoperability. Duplicated TMN applications is another obvious reality of this generation of management systems. This will be further discussed under Clause 9 (TMN Applications).

The new UMTS requirements call for open systems that can be supported by the marketplace, rather than being supported by a single (or limited) set of suppliers, due to the unique aspects of the design chosen. Open systems architectures are achieved by having the design focus on commonly used and widely supported interface standards. This should ensure costs and quality that are controlled by the forces of competition in the marketplace.

The open systems approach is a technical and business strategy to:

- Choose commercially supported specifications and standards for selected system interfaces.
- Build systems based on modular hardware and software design.

Selection of commercial specifications and standards in the Open systems approach should be based on:

- Those adopted by industry consensus based standards bodies or de facto standards (those successful in the market place).
- Market research that evaluates the short and long term availability of products.

- Trade-offs of performance.
- Supportability and upgrade potential within defined cost constraint.
- Allowance for continued access to technological innovation supported by many customers and a broad industrial base.

7.3.5 Level of openness

The level the interfaces conform to open standards is critical for the overall behaviour. A low level of openness will severely impact on long-term supportability, interoperability, development lead-time, and lifecycle cost and overall performance.

Interfaces are expensive parts in a TMN and interfaces with low level of openness severely impact on development lead-time for the introduction of any system, application component or service. Easy implementation (plug & play) is a requirement for UMTS TMN physical entities and requires a high the level of openness.

7.3.6 Closed interfaces

Many second-generation mobile network physical management entities have vendor controlled system/subsystem boundary descriptions that are not disclosed to the public or are unique to this single supplier - closed interfaces.

In a UMTS network, such interfaces will not fulfil the basic requirements and can not be a part of a UMTS TMN.

Closed interfaces can only be used as internal interfaces where no information what so ever has to be shared to other physical management entities.

7.4 Data communication networks

Within a TMN, the necessary physical connection (e.g. circuit-switched or packet-switched) may be offered by communication paths constructed with all kinds of network components, e.g. dedicated lines, packet-switched data network, ISDN, common channel signalling network, public-switched telephone network, local area networks, terminal controllers, etc. In the extreme case the communication path provides for full connectivity, i.e. each attached system can be physically connected to all others.

The TMN should be designed such that it has the capability to interface with several types of communications paths, to ensure that a framework is provided which is flexible enough to allow the most efficient communications:

- between NE and other elements within the TMN;
- between WS and other elements within the TMN;
- between elements within the TMN;
- between TMNs;
- between TMNs and enterprise.

In this case the term efficiency relates to the cost, reliability and maintainability of the data transported.

Two aspects impact costs. The first is the actual cost to transport data across the network between the TMN and the NE. The second aspect is the design of the interface including the selection of the appropriate communications protocol.

Whatever standardised protocol suite at the networking level that is capable of meeting the functional and operational requirements (including the network addressing aspects) of the Logical and Application Protocol levels of a given UMTS management interface, is a valid Networking Protocol for that interface.

A number of requirements **must** be met by the Networking Protocol, as follows:

- Capability to run over all supported bearers (leased lines, X.25, ATM, Frame Relay,...)
- Support of existing transport protocols and their applications, such as OSI, TCP/IP family, etc.
- Widely available, cheap and reliable.

The Internet Protocol (IP) is a Networking Protocol that ideally supports these requirements. IP also adds flexibility to how management connectivity is achieved when networks are rolled out, by offering various implementation choices. For instance, these may take the form of:

- Dedicated management intranets.
- Separation from or integration into an operator's enterprise network.
- Utilisation, in one way or another, of capacities of the public Internet and its applications or other resources.

7.5 New technologies

Meeting application requirements in the most affordable manner is together with development lead-time important issues identified in early UMTS management standardisation work. But the TMN functional, information and physical architectures should also keep pace with the introduction of new technologies, services and evolving network infrastructures. Technology is advancing so rapidly today that this should be a fundamental part of the physical architecture – to be able to easily adopt new important technologies.

A UMTS will need to incorporate new successful technologies from the IT-world to which TMN standardisation is not fully applicable. Today distributed computing implementations have matured to a point where the goals of TMN can be realised using commonly available technologies for a reasonable cost.

Widely accepted open standards and new IT-technologies will be indispensable to fulfil the challenging managing requirements of UMTS.

New technologies in the IT business as generic application components together with distributed processing technology are new important drivers upon application design of management systems. The possibility to purchase functional components from the open market are of great importance from many aspects as cost-efficiency and time-to-market.

8 UMTS Management Physical architectures

A UMTS Telecom Management Network will consist of many different management layers and many different building blocks. The complexity will vary greatly in detail because every organisation has different needs. The following clause will identify the most critical architectural issues and compliance conditions for a given UMTS Management Interface. It should serve as fundamental requirements for any UMTS entity (network element or management system) being a part of a UMTS TMN.

8.1 Compliance Conditions

For a UMTS entity (Management System or NE) to be compliant to a given UMTS Management Interface, all the following conditions shall be satisfied:

1) It implements the management functionality following the Information Model and flows specified by the relevant 3GPP UMTS Management Interface Specifications applicable to that interface.

2) It provides at least one of the IRP Solution Sets (were available) related to the valid Application Protocols specified by 3GPP UMTS Application Protocols for that interface, [2] Annex C.

3) It provides at least one standard networking protocol.

4) In case the entity does not offer the management interface on its own, a Q-Adapter shall be provided. This Q-Adapter shall be provided independently of any other UMTS NE and/or UMTS Management System.

5) Support for Bulk Transfer Application Protocols specified by the relevant 3GPP UMTS Management Interface Specifications applicable to that interface.

8.2 Network Element (NE) management architecture

Figure 8.1 shows two possible options for management interface from the OS upper layers to NE. Option 1, provides access to the NE via element manager, and Option 2, provides a direct access. It is sufficient to provide one or the other.

Figure 8.1 does not imply and limit the realisation of any OS physical block (e.g. E-OS, N-OS) to just one logical layer. OS physical blocks may span more than one logical layer (ITU-T Recommendation M.3010 (2000) [1]). Different types of network elements, different functional areas, operator and vendor preferences etc will put different constraints on the physical realisation of the OSFs. See further clause 9.

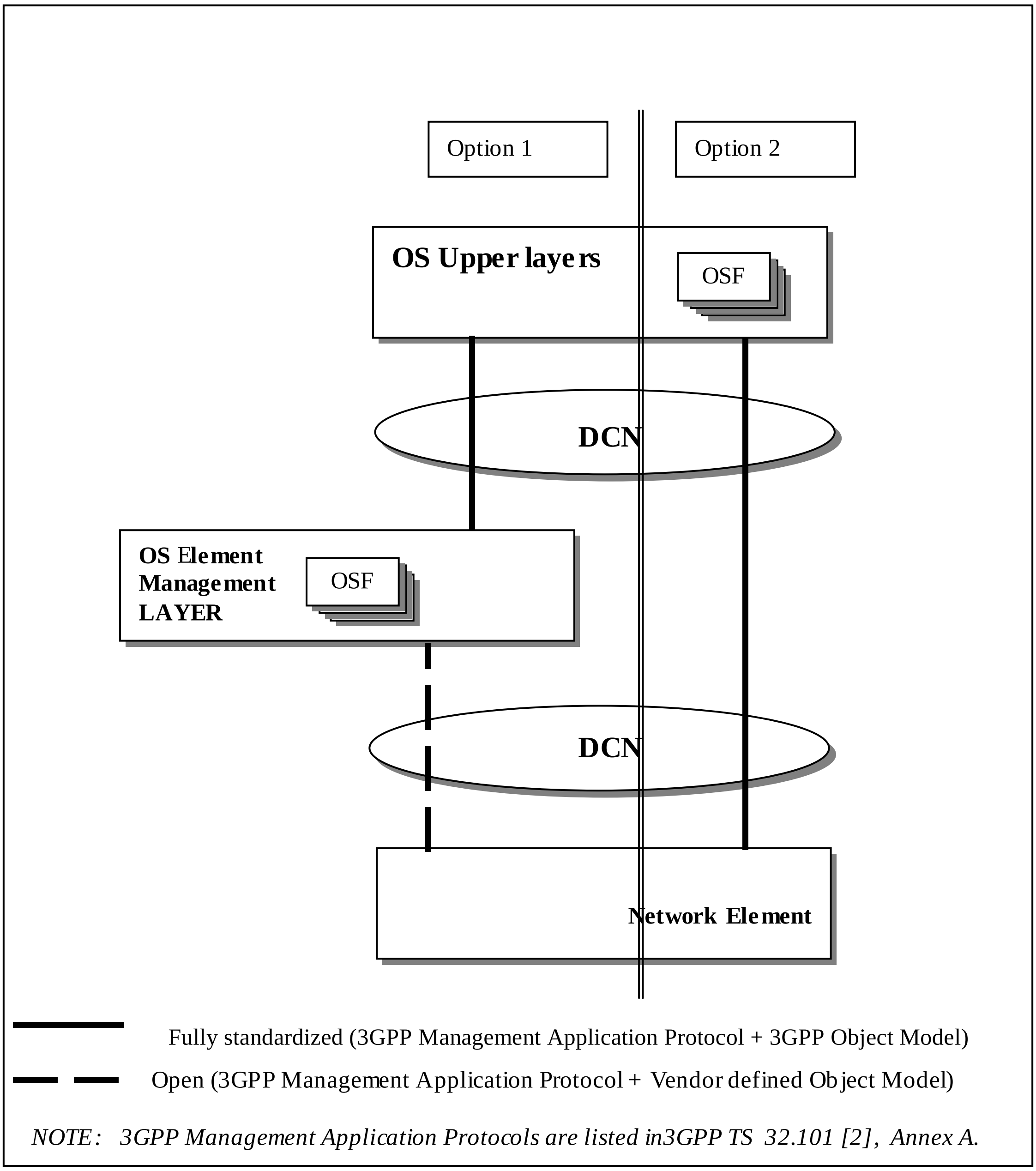


Figure 8.1: Network Element Management Architecture

For a UMTS entity (Network Element or management system) to be compliant to a given UMTS Management Interface the following conditions shall all be satisfied:

Item	Compliance conditions
1	Implements relevant 3GPP IRP Information Services and Network Resource Models For an interface illustrated by the dashed line in figure 4 the object model is not standardised but it shall be open
2	Application protocol (e.g. CMIP,SNMP,CORBA IIOP) (Defined in TS 32.101 [2], Annex A) If 3GPP has specified one or more IRP Solution Sets corresponding to the IRP Information Models in item 1 then at least one of those IRP Solution Sets shall be supported. (Defined in TS 32.101 [2], Annex C)
3	Valid Network Layer Protocol (see Annex B of TS 32.101 [2])
4	Lower protocol levels required by Item 1,2 and 3

8.3 Subnetwork Management Architecture

(Example UMTS RNC / NodeB)

An important special case of the network element management architecture is where one type of network element as the RNC will need management information for co-ordination of a subnetwork of other types of network elements as NodeB.

This management information shared between the RNC and NodeB will not reach the operators and is not considered to be a part of the UMTS TMN. All other management information related to NodeB will transparently be transferred by the RNC towards the UMTS TMN.

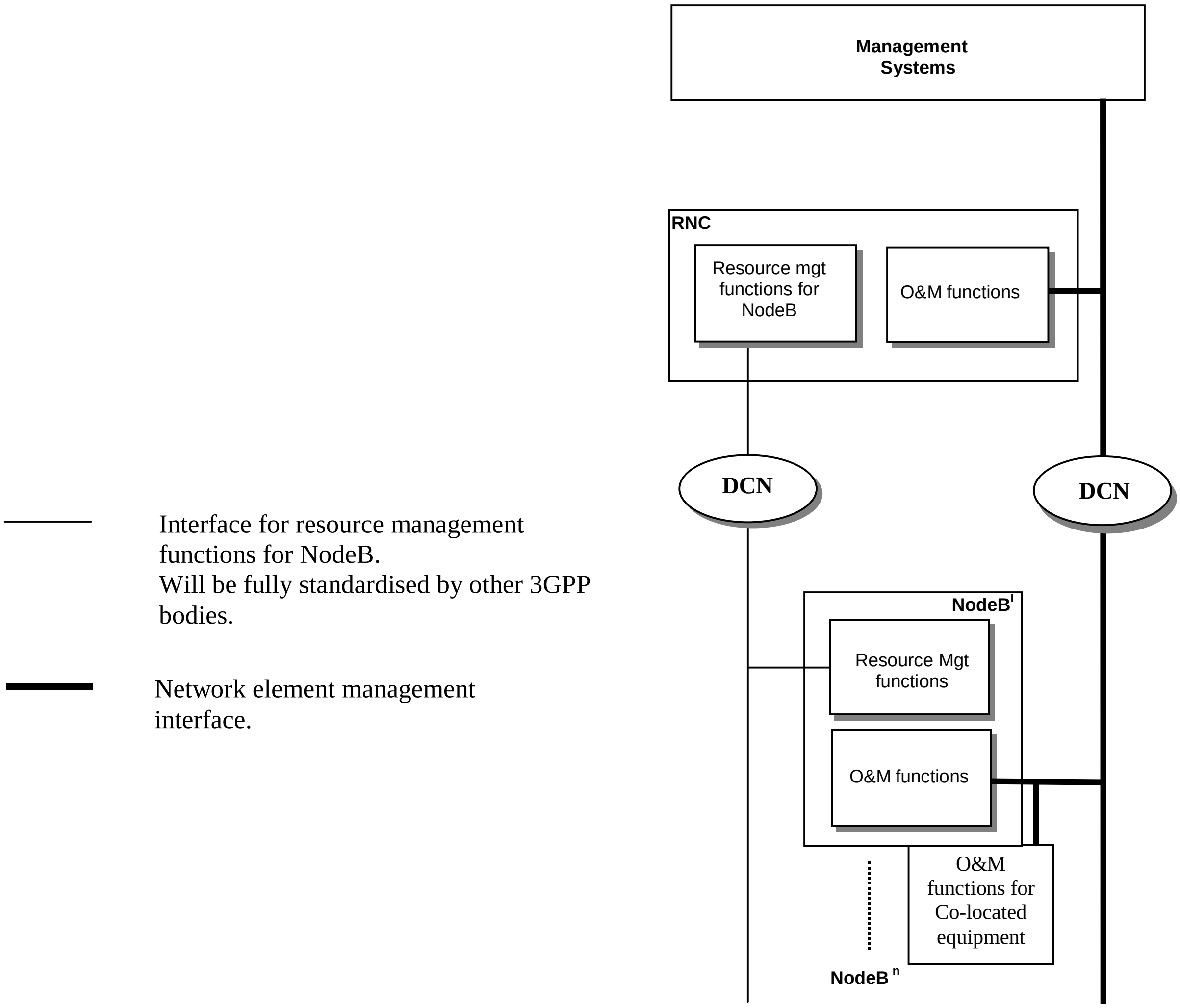


Figure 8.2: Subnetwork Management Architecture

The same compliance conditions apply for the subnetwork management architecture as for the network element management architecture (see clause 8.2).

8.4 Operations Systems interoperability architecture

Interoperability between operations systems is an important issue in a UMTS. Different organisations may take different roles in a UMTS. The need to share information across corporate boundaries will be a consequence of this.

The heterogeneous, distributed and complex network of a UMTS will be a market for many different vendors. All operations systems have to interoperate and shall be able to share information. This is a critical issue in the management of third generation systems.

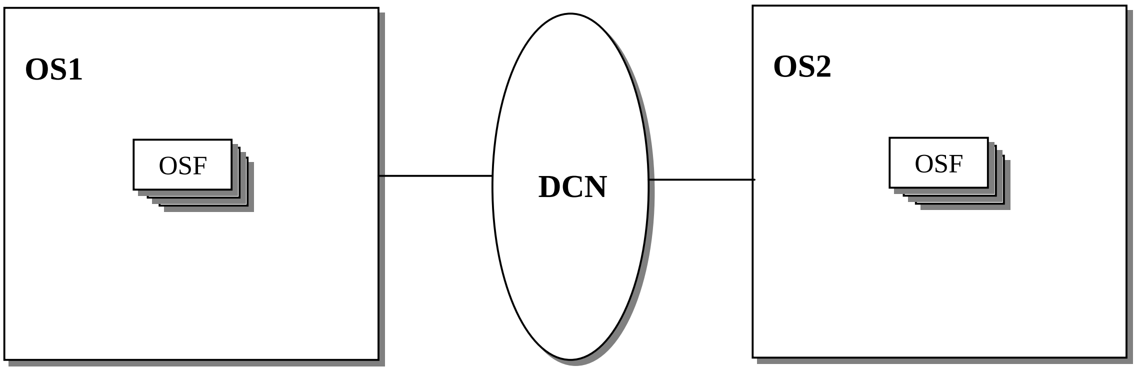


Figure 8.3: Operations Systems interoperability Architecture

For a Operations System to be UMTS TMN compliant the following conditions shall all be satisfied:

Item	Compliance conditions
1	Implements relevant 3GPP IRP Information Services and Network Resource Models
2	Application protocol (e.g. CMIP,SNMP,CORBA IIOP) (Defined in TS 32.101 [2], Annex A) If 3GPP has specified one or more IRP Solution Sets corresponding to the IRP Information Models in item 1 then at least one of those IRP Solution Sets shall be supported. (Defined in [2], Annex C)
3	Valid Network Layer Protocol (see Annex B of TS 32.101 [2])
4	Lower protocol levels required by Item 1,2 and 3

8.5 Operations Systems intra-operability architecture

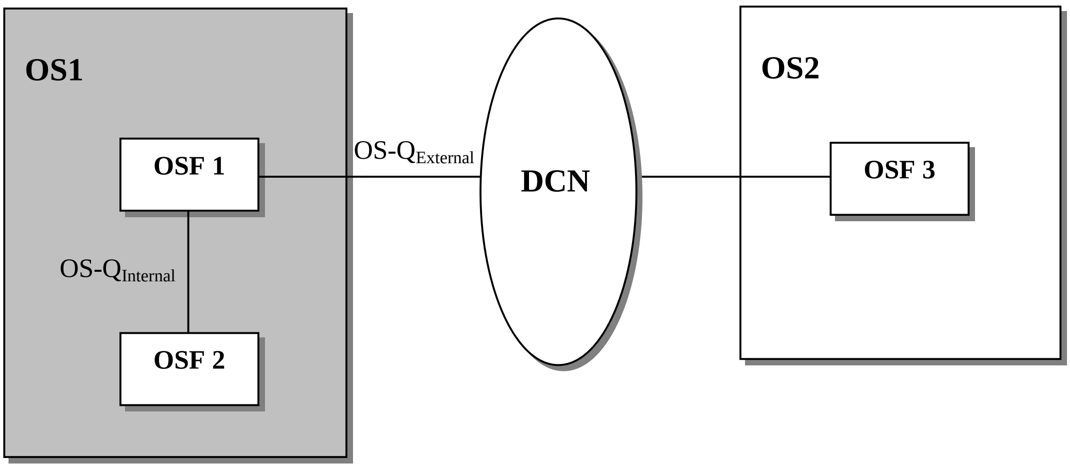


Figure 8.4: Operations Systems intra-operability Architecture

OS-$Q_{Internal}$ indicates an internal flow and is not standardised.

OS-$Q_{External}$ indicates an external flow and shall to be compliant to a given UMTS Management Interface satisfy the following conditions:

Item	Compliance conditions
1	Implements relevant 3GPP IRP Information Services and Network Resource Models
2	Application protocol (e.g. CMIP,SNMP,CORBA IIOP) (Defined in TS 32.101 [2], Annex A) If 3GPP has specified one or more IRP Solution Sets corresponding to the IRP Information Models in item 1 then at least one of those IRP Solution Sets shall be supported. (Defined in TS 32.101 [2], Annex C)
3	Valid Network Layer Protocol (see Annex B of TS 32.101 [2])
4	Lower protocol levels required by Item 1,2 and 3

8.6 Business System interconnection architecture

The business management layer has in the second-generation systems a very low degree of standardisation. Operators have legacy systems or more IT influenced systems often adopted to every organisations different needs. Business systems are not a part of a UMTS TMN.

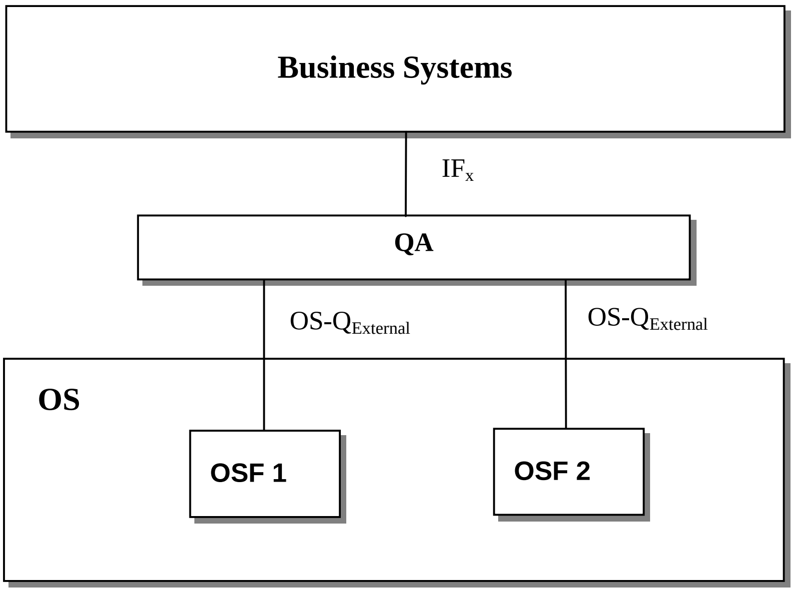


Figure 8.5: Business Systems interconnection architecture

$OS\text{-}Q_{Exteral}$ Indicates an external flow and shall to be compliant to a given UMTS Management Interface satisfy the following conditions:

Item	Compliance conditions
1	Implements relevant 3GPP IRP Information Services and Network Resource Models
2	Application protocol (e.g. CMIP,SNMP,CORBA IIOP) (Defined in TS 32.101[2], Annex A) If 3GPP has specified one or more IRP Solution Sets corresponding to the IRP Information Models in item 1 then at least one of those IRP Solution Sets shall be supported. (Defined in TS 32.101 [2], Annex C)
3	Valid Network Layer Protocol (see Annex B of TS 32.101 [2])
4	Lower protocol levels required by Item 1,2 and 3

IF_X indicates an external flow and shall to be compliant to a given UMTS Management Interface satisfy the following condition:

Item	Compliance conditions
1	Not standardised but open

9 TMN applications

Telecom management applications can be implemented in many different ways depending on constraints presented in previous clauses of the present document. Consistent operational processes are required for the management of the network irrespective of vendor equipment. A mobile operator can because of the very heterogeneous nature of their networks easily end of with dozens of duplicated applications for e g alarm surveillance. Most vendors of network equipment offers dedicated net-element managers and the ones not built with an open system approach will severely limit the possibility to report and manage the network in a consistent way.

Network element vendors with closed and unique net-element managers or operations systems with closed interfaces or interfaces with low level off openness will not fulfil the basic requirements as a part of a UMTS. It will not be possible to design and build the Telecom Management Network to support the operational processes as required. Such physical entities are not under consideration in the present document.

Many TM application functions can be identified as generic functions used by all major types of telecom equipment. Alarm surveillance applications and performance analysing applications are generic necessities to manage most network elements. Security and system management applications are also common to many TM components and may be the scope for overall business policies.

To identify and specify the design criteria that will allow re-usable application components to be developed across multiple telecom business scenarios are important issues to fulfil the basic UMTS Management requirement. "To minimise the costs of managing a UMTS network such that it is a small component of the overall operating cost".

The implication of the top down approach in the standardising work of UMTS is that consistent operational management processes are required irrespective of vendor equipment.

Generic management applications is required to facilitate:

- Reduced management application development costs.
- Simplification of operational processes and associated reduction in costs.
- Reduced time to deploy new services as management systems already exist.
- Consistent representation of basic information.

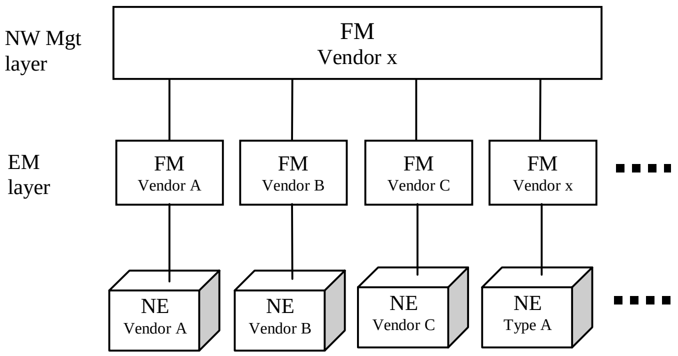


Figure 9.1: Unique NE Fault Management

Figure 9.1 represents a very common situation in the management of second generation of mobile networks. Different vendors supplied their network elements with unique net-element managers. The interfaces were mostly proprietary or unique. The information models for generic information as alarms were rarely standardised. All together the consequence for the operators became very complex. Similar information at many levels, repeated acknowledge of alarms, inconsistent representation of similar information are a few of all the difficulties the operators had to cope with.

Some of the more severe implication of this situation is the difficulty to add more intelligence into the applications to better support the processes of the network providers. The operators who tried to brake up this situation had to put in a lot of effort into software development and proprietary interfaces. The marketplace did not support the needs of the operators.

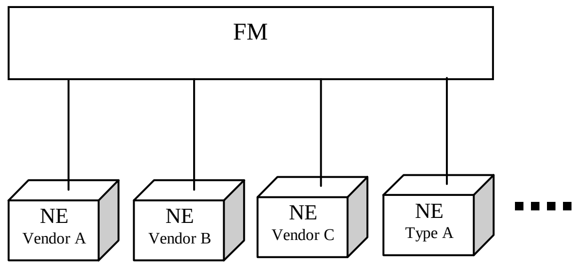


Figure 9.2: Generic Fault Management

Figure 9.2 indicates the situation were the Telecom Management process alarm surveillance is supported by a generic application for Fault Management. A common information model and accessibility to all related information will make it possible to add more intelligence to the management systems and to better support the management task.

TMN application functions as billing information collection or configuration management of a specialised network element are examples of application that can be identified as unique applications. Even these applications will need to interoperate with other applications and will also need the open system approach to be a part of a UMTS TMN. With a network with many different types of network elements a common graphical user interface as a web browser for configuration management applications could be an important issue to create consistent operational processes.

The complexity and heterogeneous nature of UMTS calls for easy integration (plug&play) of HW/SW.

10 Integration Reference Points (IRPs)

10.1 General

Relating to the OSI functional areas "FCAPS", IRPs are here introduced addressing parts of "FCPS" – Fault, Configuration, Performance, and Security management. Comparing with TMF TOM (Telecom Operations Map) [9], the introduced IRPs address process interfaces at the EML-NML (Element Management Layer – Network Management Layer) boundary. In 3GPP/SA5 context, this can also be applied to the Itf-N between EM-NM and NE-NM.

The three cornerstones of the IRP concept are:

- **Top-down, process-driven modelling approach**
  The purpose of each IRP is automation of one specific task, related to TMF TOM. This allows taking a "one step at a time" approach with a focus on the most important tasks.

- **Protocol-independent modelling**
  Each IRP consists of a protocol-independent model (the IRP Information Service) and several protocol-dependent models (IRP solution sets).

- **Standard based protocol dependent modelling**
  Models in different IRP solution sets (CMIP, SNMP, WBEM etc.) will be different as existing standard models of the corresponding protocol environment need to be considered. The means that solution sets largely need to be "hand crafted".

10.2 Integration levels

Virtually all types of telecom/datacom networks comprise many different technologies purchased from several different vendors. This implies that the corresponding management solution need to be built by integrating product-specific applications from different vendors with a number of generic applications that each provide some aspect of multi-vendor and/or multi-technology support. A complete management solution is thus composed of several independent applications.

The following levels of integration are defined:

- **Screen Integration:** Each application provides its own specific Graphical User Interface (GUI) that need to be accessible from a single, unified screen (a common desktop). A seamless integration between the various GUIs is then required. Screen Integration is not specified in the present document.

- **Application Integration:** Applications need to interwork, on a machine-machine basis, in order to automate various end-to-end processes of a communication provider.

10.2.1 Application integration

Interfaces related to application integration can be divided in the following three categories:

- **High-level generic interfaces** between generic applications on the network and service management layers. The same approach and concepts apply for these as the next category:

- **High-level (technology-independent to the extent possible) interfaces** between product-specific and generic applications are needed in order to automate and streamline frequently occurring tasks applicable to several types of network elements. A top-down approach shall be taken when defining these interfaces, where the main input is (1) business processes of a communication provider, and (2) the types of generic applications that are used to implement the process support. The interfaces need to be stable, open and (preferably) standardised. These IRPs are discussed below under the heading Network Infrastructure IRPs.

- **Detailed (product-specific) interfaces** between product-specific applications and the corresponding network elements are of course also needed. These interfaces are defined using the traditional bottom-up approach, where the actual network infrastructure is modelled. This is the traditional TMN approach to element management. The management information in these interfaces is not further discussed in this document, as it is internal to a specific development organisation and does not need to be open. In fact, by publishing the management information in these interfaces, too much of the internal design may be revealed and it may become impossible to later enhance the systems that are using the interfaces. The management services (operations and notifications) and protocol shall however be open and standardised as long as they are independent of the NRM describing the managed NEs/NRs.

10.3 Network infrastructure IRPs

When providing integrated management solutions for multi-vendor networks, there is a strong requirement that the NEs and the management solutions that go together with them are systems integrateable. It is here proposed that the telecom vendors provide a set of Network Infrastructure IRPs.

It should be noted that these IRPs could be provided by either the NE, or the Element Manager (EM) or Sub-Network Manager (SNM) that goes together with the type of NE. There is actually not a clear distinction any more between NE and element management applications, mainly due to the increased processing capacity of the equipment platforms. Embedded Element Managers providing a web user interface is a common example of that.

These IRPs are introduced to ensure interoperability between Product-Specific Applications (PSA) and the types of generic applications shown in the figure below. These IRPs are considered to cover the most basic needs of task automation.

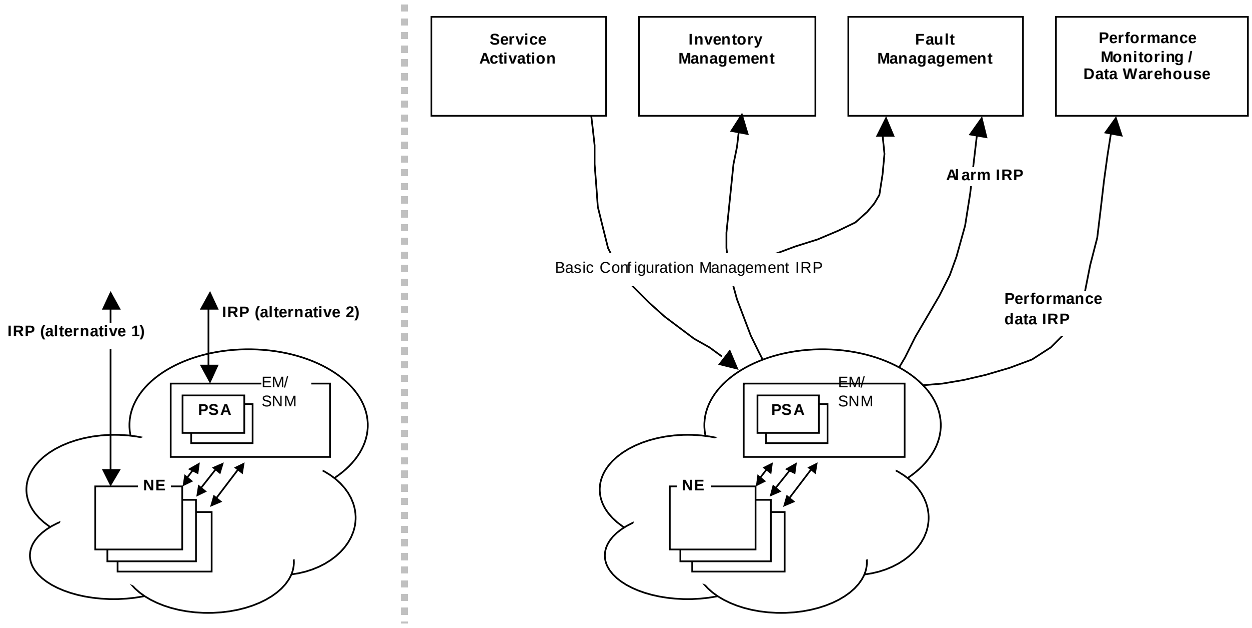


Figure 10.1: IRPs for application integration

The IRPs presented in figure 10.1 are just an example and do not reflect the exact set of IRPs defined by the 3GPP.

Many IRPs have similar needs to use notifications. The corresponding service is formalised as a *Notification IRP*. It specifies: firstly, an interface through which subscriptions to different types of notifications can be set up (or cancelled), and secondly, common attributes for all notifications.

Further, applying a common *Name Convention for Managed Objects* is useful for co-operating applications that require identical interpretation of names assigned to network resources under management.

10.4 Defining the IRPs

It is important to accommodate more than one specific technology, as the technologies will change over time. Applications need to be future-proof. One fundamental principle for achieving this is to clearly separate the semantics of information definition from the protocols definitions (accessing the information) for the external interfaces.

The framework being used to define IRPs allows the implementation of user requirements for each management capability (e.g. configuration management), by modelling the information related to the resources to be managed and the way that the information may be accessed and manipulated. Such modelling is done in a way that is independent of the technology and distribution used in the implementation of a management system.

An IRP for a management capability is composed of 3 types of documents. The first type of document captures the user requirements. The second type of document, known as "Information Service", specifies the information observable and controlled by management system’s client, related to the network resources under management. The IS document also specifies the semantics of the interactions used to carry these information. The third type of document, known as "Solution Set", contains specification of the system in terms of technology choice (e.g. CMIP, CORBA). In this type of specification, the syntax, rather than the semantic are specified. One instance of a Solution Set document is produced per communication technology supported.

The IRP methodology uses the following steps:

a. Capture the management requirements.
b. Specify the semantics of the information to describe the system. Trace back to item (a).
c. Specify the semantics of the interactions between the management system and its clients. Trace back to item (a).
d. Specify the syntaxes of the information and interactions identified in (b) and (c). The specification is technology dependent. Trace back to items (b) and (c)

The set of resources that form an NRM can also be described using the requirement documents and the Information Service (without the part on information access). Both the NRM and IRP Information Service definitions are used to define Solution Sets to develop management capabilities at, for example a CORBA based interface.

As presented above, the Information Service document may contain two parts, the information related to the resources to be managed and the way that the information may be manipulated.

The first part defines the information types within a distributed system. It is in line with the Analysis phase of ITU-T M.3020. From the point of view of the Network Level modelling work it reflects the information aspects (including states and significant transitions) of the managed resources and the management services. It defines information object classes, the relationships between these object types, their attributes and states along with their permitted state transitions. It may also define the allowable state changes of one or more information objects. As recommended in M.3020, UML diagrams (class diagram, state diagram) are used to represent information when appropriate. This rest of the specification is described using an information description specified in natural language with appropriate label keywords (e.g. DEFINITION, ATTRIBUTE, CONSTRAINTS, etc…). A definition of the IS information template is provided in Annex C.

Management service specific information objects may be created by subclassing from the objects in the basic network model, and extending them for that application. In this case, the new management service specific subclass may include other attributes, in addition to those defined in its superclass. Additional relationships and attributes may also be created as needed for that management service. Completely new objects can also be added.

The second part defines interfaces. Each interface contains one or more operations or one or more notifications that are made visible to management service users. An interface encapsulates information exchanged that is atomic in the sense that either all the information exchanged are visible (to management service users) or none. In addition, the specification of the information exchanged is in semantics only. No syntax or encoding can be implied. The operations or notifications are defined with their name, input and output parameters, pre and post conditions, raised exceptions and operation behaviour. These operation and notification specifications refer, through the utilisation of parameter matching, to the information objects. A definition of the IS operations/notifications template is provided in Annex C.

The Solution Set document contains the mapping of the information objects and interactions specified in the IS, into their corresponding syntaxes of a particular chosen technology. The mapping is infrastructure specific and satisfies scenarios where interfaces have been selected, according to mapping choices (driven for example by system performance, development cost, time to market). The mapping is not always one-to-one. General rules valid for all IRP Solution Sets are defined in Annex D. Rules for specific Solution Sets, such as CORBA, are defined in an Annex for each of the Solution Set technologies used by 3GPP.

Managed Object Classes as defined in a CMIP or CORBA Solution Set document represents a mapping into GDMO or IDL of Information Object Classes and other additional objects classes that can be introduced to support interfaces defined in the Information Service. Whether instances of Managed Object Classes are directly accessible or not may not be specified by IRP specifications.

Figure 10.2 shows an example of how an IRP can be structured (the Alarm IRP). Note that Figure 10.2 is only an example of what could be the Alarm IRP, the Alarm IRP specified in GPP TS 32.111 [14] can be different.

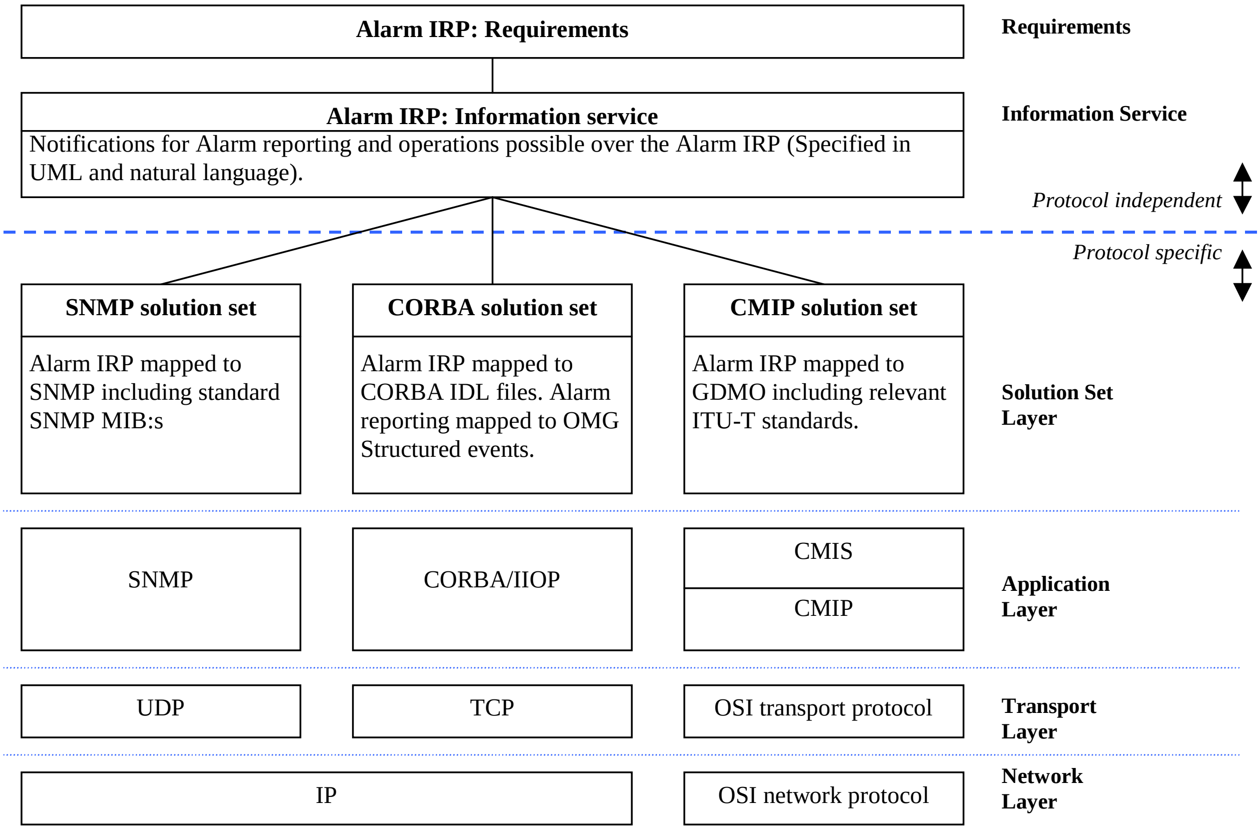


Figure 10.2: Example of an IRP (Alarm IRP)

10.5 Relationships between specifications

This subclause presents the target architecture of SA5 Network Resources Models, Information Services and Information Models. This architecture is based on the concepts of level and partition of information. To achieve this, information object classes and interfaces are defined and grouped into packages which can be related to each others through the import relationship.

Level means that the information services are structured in a way that enables re-utilization between levels, either through inheritance or through a traditional relationship between classes. Four levels are identified namely:

1. A **generic Network Resource Model IS**, also called "Generic NRM", which defines the information object classes and interfaces that are independent of any 1/ protocol (e.g. CORBA / IDL, CMIP / GDMO, etc.) and 2/ "sub-network" (e.g. UTRAN, GERAN, CN). This Network Resource Model contains definitions of the largest subset of information object classes that are common to all the Network Resources Models to be defined in SA5. This Network Resources Model is part of Level 1. For this Information Service, a number of solution sets may be provided;

2. A number of **domain-specific Network Resource Model ISs**. Up to now, three Network Resource Models of this type have been identified: the CN Model, the UTRAN Model and the GERAN Model. They are part of Level 2. These Network Resource Models are specified in corresponding packages and import information object classes from the Generic Network Resources Model defined in Level 1. For each of these Information Services, a number of solution sets may be provided;

3. A number of **function-specific ISs**. Such information services as the BasicCmIRP IS, the NotificationIRP IS and the AlarmIRP IS are part of this level. They are part of Level 3. These Information Services are specified in corresponding packages and import information object classes and interfaces defined in Level 2. For each of these Information Services, a number of solution sets may be provided;

4. A number of (protocol-independent) **Information Models**. Up to now, none of them have been defined. They will be part of Level 4. These Information Models are specified in corresponding packages and import information object classes and interfaces defined both in Level 2 and in Level 3. An example of such Information Model is a "UTRAN Alarm IM" (see Figure 10.3). For each of these Information Models, a number of solution sets may be provided;

These levels provide a means for separation of concerns and re-utilization.

NRM and IRP ISs shall be kept as simple as possible. To achieve this, information object classes and interfaces shall be grouped into packages. The grouping shall be based on semantics, i.e. information object classes and interfaces which participate to the definition of a given IRP or NRM should be gathered into a dedicated package.

Re-utilization of information specification contained in an NRM or IRP IS previously specified shall be possible through the *import* relationship. The import relationship is a means for re-utilization : once a piece of information (i.e. an information object class, an attribute, a relationship or an interface) defined in an NRM or IRP IS is imported in another NRM or IRP IS, it is added to the name space of the importing NRM or IRP IS. Then, the whole information available in a NRM or IRP IS is made up of the information which is owned by the NRM or IRP IS itself (i.e. defined in this document) plus the information which is imported from other NRM(s) or IRP IS(s). This imported information can then be utilised in the importing NRM or IRP IS, for instance, through:

- inheritance (e.g. any information object class defined at Levels 2 to 4 inherits from the information object class Top defined in the generic NRM at Level 1), either directly or indirectly;

- relationship (e.g. any information object class defined at Levels 2 to 4 may have a containment relationship with the information object class IRPAgent defined in the generic NRM at Level 1).

An illustration of this architecture is provided in figure 10.3 below; it uses the UML diagrammatic conventions.

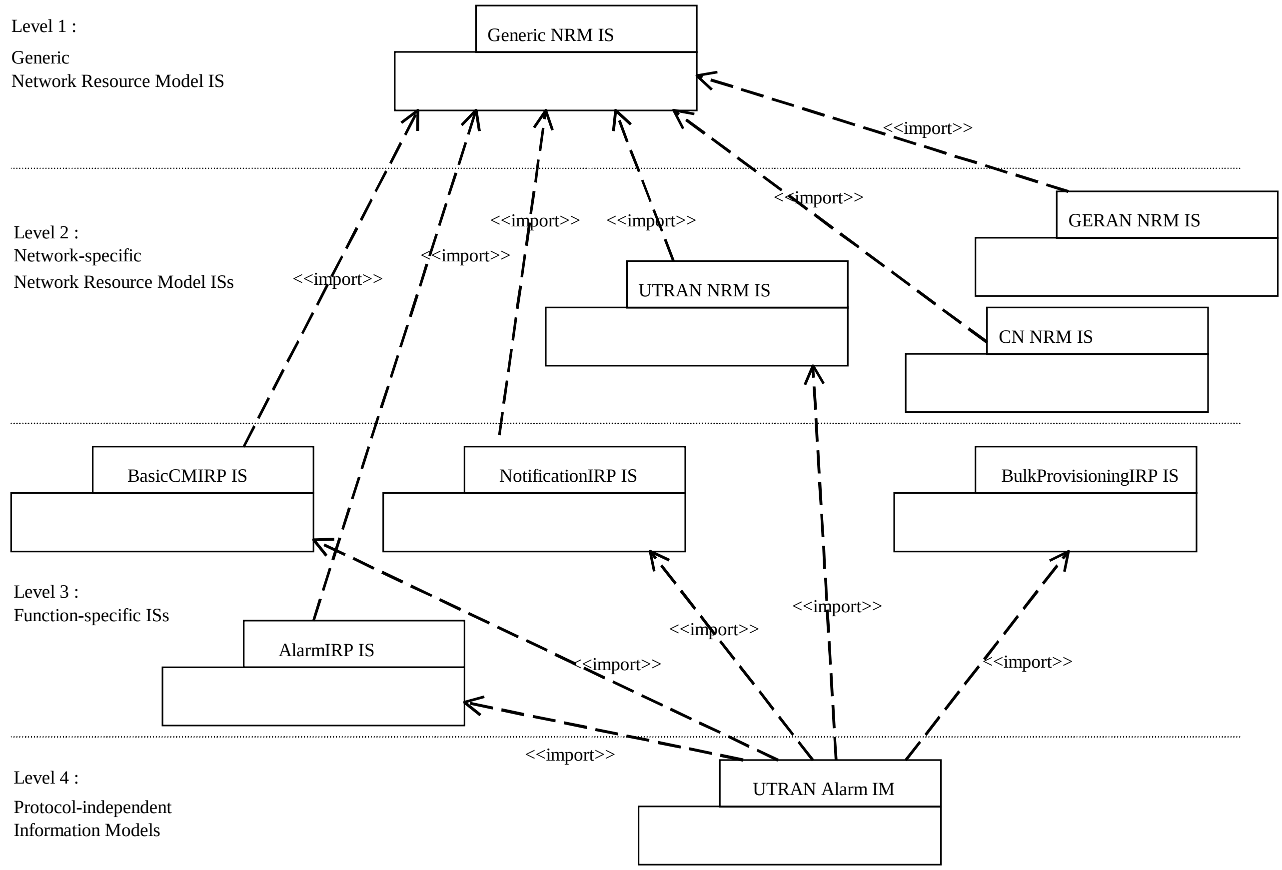


Figure 10.3: Specification architecture (not complete)

In order not to mix up the concept of "information object class" and "interface" with other concepts such as "managed object class" and "manager / agent interface", the former are labelled according to the UML notation capability (cf. stereotype). "Information object class" is defined as a stereotype of "Class" in the UML meta-model. As a consequence, information object classes defined in Information Models are labelled <<InformationObjectClass>>. Similarly, interfaces are labelled <<Interface>>. In Annex C you can find an example of the inheritance between some ISs.

The following piece of information regarding the Semantics of the relationship "import" can be imported from other standard documents:

1. An information object class. The definition of the IOC, the attributes and the roles that the IOC plays in some relationships are imported. The import clause shall specify the TS number from which the IOC is imported and the name of the IOC;

2. An attribute. Two cases can happen :

    2.1. An attribute definition. In this case, the attribute definition is imported. The import clause shall specify the TS number from which the attribute is imported and the name of the attribute;

    2.2. An attribute reference within an IOC definition. In this case, the attribute definition is imported together with its qualifier within the specified IOC. The import clause shall specify the TS number from which the attribute is imported, the name of the IOC and the name of the attribute;

3. A relationship. The definition of the relationship is imported. The import clause shall specify the TS number from which the relationship is imported and the name of the relationship;

4. An interface. The definition of the interface and all its operations or notifications are imported. The import clause shall specify the TS number from which the interface is imported and the name of the interface;

5. An operation or a notification. The definition of the operation / notification is imported. The import clause shall specify the TS number from which the operation / notification is imported, the name of the interface in which the operation / notification is defined and the name of the operation / notification;

A piece of information **must** always be imported from the TS where it is initially defined. It cannot be imported from any other.

10.6 Mandatory, Optional and Conditional qualifiers

This subclause defines a number of terms used to qualify the relationship between the 'Information Service', the 'Solution Sets' and their impact on the IRP implementations. The qualifiers defined in this clause are used to qualify IRPAgent behaviour only. This is considered sufficient for the specification of the IRPs.

Table 1 defines the meaning of the three terms Mandatory, Conditional and Optional when they are used to qualify the relations between operations, notifications and parameters specified in 'Information Service' documents and their equivalents in Solution Set (SS) documents.

Table 1: Definitions of Mandatory, Optional and Conditional Used in Information Service Documents

	Mandatory (M)	Conditional (C)	Optional (O)
Operation and Notification	Each Operation and Notification shall be mapped to its equivalents in all SSs. Mapped equivalent shall be M.	Each Operation and Notification shall be mapped to its equivalents in at least one SS. Mapped equivalent can be M or O.	Each Operation and Notification shall be mapped to its equivalents in all SSs. Mapped equivalent shall be O.
Input and output parameter	Each parameter shall be mapped to one or more information elements of all SSs. Mapped information elements shall be M.	Each parameter shall be mapped to its equivalent in at least one SS. Mapped equivalent can be M or O.	Each parameter shall be mapped to its equivalent in all SSs. Mapped equivalent shall be O.
Information relationship	Each relationship shall be supported in all SS's.	Each relationship shall be supported in at least one SS.	Each relationship shall be supported in all SS's.
Information attribute	Each attribute shall be supported in all SS's.	Each attribute shall be supported in at least one SS.	Each attribute shall be supported in all SS's.

Table 2 defines the meaning of the two terms Mandatory and Optional when they are used to qualify the operations, parameters of operations, notifications and parameters of notifications in Solution Sets.

Table 2: Definitions of Mandatory and Optional Used in Solution Set Documents

Mapped SS Equivalent	Mandatory	Optional
Mapped notification equivalent	IRPAgent shall generate it.	IRPAgent may or may not generate it.
Mapped operation equivalent	IRPAgent shall support it.	IRPAgent may or may not support this operation. If the IRPAgent does not support this operation, the IRPAgent shall reject the operation invocation with a reason indicating that the IRPAgent does not support this operation. The rejection, together with a reason, shall be returned to the IRPManager.
input parameter of the mapped operation equivalent	IRPAgent shall accept and behave according to its value.	IRPAgent may or may not support this input parameter. If the IRPAgent does not support this input parameter and if it carries meaning (i.e., it does not carry no-information semantics), the IRPAgent shall reject the invocation with a reason (that it does not support the parameter). The rejection, together with the reason, shall be returned to the IRPManager.
Input parameter of mapped notify equivalent AND output parameter of mapped operation equivalent	IRPAgent shall generate it.	IRPAgent may generate it.

11 Implementation aspects

UMTS operators might categories and organise its operation systems in many different ways as:

- A national fault and performance OS.
- A national charging, billing and accounting OS.
- Regional configuration OS.
- Regional fault, performance and configuration OS.
- etc.

This geographical dependent categorisation may change after time and the growth of the network. A physical architecture based on an open system design and re-usable application components would ease the work to adopt such structural changes. A management system build for a UMTS shall provide the possibility of layering the applications.

12 UMTS TMN Conformance

The goal of TMN conformance (see M.3010 [1]) is to increase the probability that different implementations within a TMN will be able to interwork, that TMNs in different service/network provider's administrations and customer's system will be able to interwork as much as agreed on.

TMN conformance are testable conditions.

It is only the requirements on the external behaviour that have to be met by the conformance statements.

To finally guarantee interoperability the purchaser/user shall be able to test and verify that any two systems, claiming any type of TMN conformance, interoperate. Interoperability testing shall include:

- Testing of the interface protocols;
- The shared/exposed information over those interfaces;
- The interface functionality of the system.

A UMTS TMN conformant entity shall support necessary information to support such interoperability testing namely:

- Statements made by the supplier of an implementation or system claimed to conform to a given specification, stating which capabilities and options have been implemented.
- Detailed information to help determine which capabilities are testable and which are un-testable.
- Information needed in order to be able to run the appropriate test.
- The system interface documentation shall list the documents that define the specified UMTS information models with the inclusion of the version number and date.
- Necessary information about vendor supplied extensions of a standardised interface

The interface specification shall be documented, publicly available and licensable at reasonable price on a non-discriminatory basis.

Specific conformance guidelines shall be included in the different IRP solution sets. A UMTS TMN conformant entity **must** support information stated in those conformance guidelines.

13 TMN planning and design considerations

A TMN should be designed such that it has the capability to interface with several types of communications paths to ensure that a framework is provided which is flexible enough to allow for the most efficient communications:

- Between one NE and other elements within the TMN;
- Between a WS and other elements within the TMN;
- Between elements within the TMN;
- Between TMNs.

The basis for choosing the appropriate interfaces, however, should be the functions performed by the elements between which appropriate communications are performed. The interface requirements are specified in terms of function attributes needed to provide the most efficient interface.

13.1 Function attributes

a) *Reliability* – The capability of the interface to ensure that data and control are transferred such that integrity and security are maintained.

b) *Frequency* – How often data is transferred across the interface boundary (Normal behaviour).

c) *Quantity* – The amount of data that is transferred across the interface during any transaction.

d) *Priority* – Indicates precedence to be given to data in case of competition for network resources with other functions.

e) *Availability* – Determines the use of redundancy in the design of the communications channels between interfacing elements.

f) *Delay* – Identifies the amount of buffering that may be tolerable between interfacing elements. This also impacts communications channel designs.

Table 3 suggests a possible ranges for these function attributes.

Table 3: Possible ranges for TMN function attributes [1]

<table>
<tr><th colspan="2">Attributes</th><th>Requirements</th><th>Nature of attributes</th></tr>
<tr><td rowspan="3">Performance or grade of service (P)</td><td>Delay (speed)</td><td>Short
Medium
Long</td><td rowspan="3">Objective of design and control (acceptable/unacceptable but available/unavailable)</td></tr>
<tr><td>Reliability (accuracy)</td><td>High
Medium
Low</td></tr>
<tr><td>Availability</td><td>High
Medium
Low</td></tr>
<tr><td rowspan="3">Characteristics of TMN traffic (C)</td><td>Quantity</td><td>Large
Medium
Small</td><td rowspan="3">Condition or parameter of design</td></tr>
<tr><td>Frequency</td><td>Often continuous
Periodic
Sparse</td></tr>
<tr><td>Priority</td><td>High
Medium
Low</td></tr>
</table>

13.2 Functional characteristics

Each major type of telecommunications equipment has functional characteristic needs that can be used to describe the complexity of the interface.

There are, however, a basic group of TMN application functions that cross all major types of telecommunications equipment. There are also unique TMN application functions that are performed by specific categories of major telecommunications equipment. Alarm surveillance is an example of the former, whereas billing information collection is an example of the latter.

Functional characteristics of the elements within a TMN, e.g. OS, DCN and MD also describe the complexity of interfaces between these elements.

13.3 Critical attributes

Attribute values for a given function are generally consistent across the network elements.

When considering a single interface, it is important to identify the controlling attribute ranges for the design of the interface.

If there are conflicting attribute values for different functions in a given network element, more than one instance of an interface may be needed.

Overall TMN attribute values for the interfacing of elements within the TMN depend on the type and number of functions performed within these elements. In this case the functions are not consistent across TMN elements, but are controlled by the individual TMN design of an Administration.

13.4 Protocol selection

In many cases, more than one protocol suite will meet the requirements for the network element or TMN element under consideration. It is the approach for the 3GPP Telecom management standardisation to concentrate on protocol independent information models, allowing the mapping to several protocol suites.

The rational behind this is:

- The blurring of Information and Telecommunication technologies in UMTS, it is required to work on a more open approach (acknowledging the market status and foreseen evolutions).

- The life-cycle of information flows is 10 to 20 years, while the protocols is 5 to 10 years.

- The developments on automatic conversion allows for a more pragmatic and open approach.

The choice of the individual protocol from the recommended family will be left open to the vendors and operators.

To provide the most efficient interface care should be taken to select the protocol suite that optimises the relationship between the total cost to implement that protocol suite, the functional attributes and the data communications channels that carry the information across the interface.

13.5 Communications considerations

DCN architectures should be planned and designed to ensure that their implementation provides appropriate degrees of availability and network delay while minimising cost.

One should consider the selection of communications architectures, e.g. star, multipoint, loop, tree, etc..

The communications channels, e.g. dedicated lines, circuit-switched networks and packet networks used in providing the communications paths, also play an important role.

14 Mediation/Integration

The increase in the need to incorporate a hybrid set of technologies, multiple protocols and heterogeneous resources requires the availability of open management interfaces between the management systems and the different network resources. These interfaces require an underlying mechanism to mediate - interpret, translate, and handle data – between the various data representations and protocols. A set of Technology Integration Points [10] can be identified and will need to be supported.

Software components on the open market as automatic conversion applications, gateways, mediation applications will be valuable products to fulfil the challenging task to incorporate multiple protocols and heterogeneous resources.

Figure 14.1 summarises Technology Integration Points for some technologies:

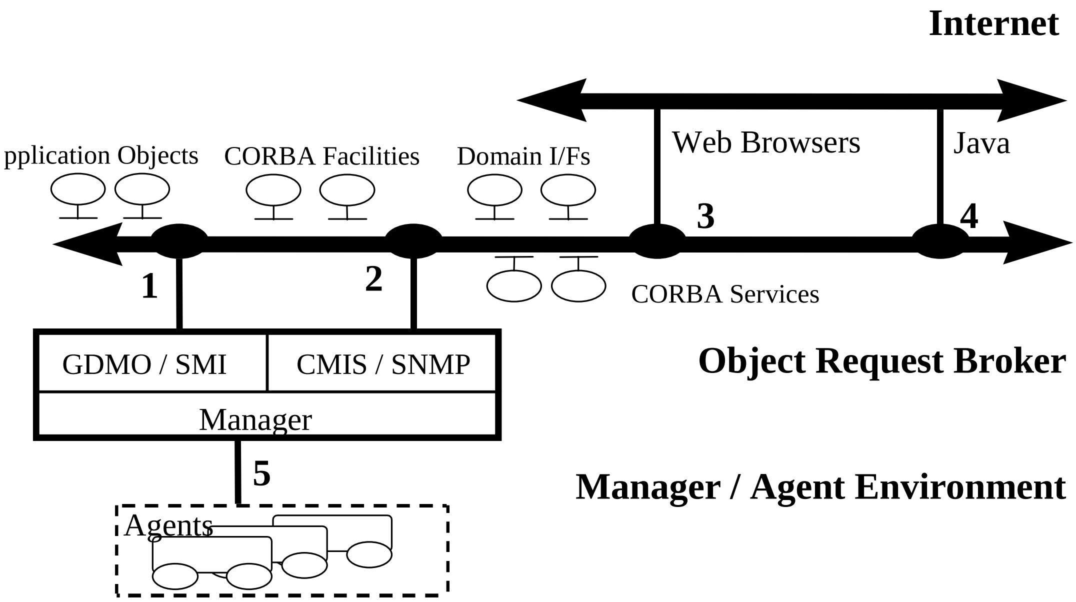


Figure 14.1: Technology Integration Points [10]

Essentially, figure 14.1 indicates that from the technologies selected, three technology areas will need to be integrated. These are:

- Internet/Web based services;
- Object Request Broker (CORBA) based services;
- Telecom based Manager/Agent services (i.e. CMIP/GDMO and SNMP/SMI).

In order to provide adequate points of integration between these areas of technology, five Integration Points (IPs) have been identified - as outlined in table 4 below:

Table 4: Technology Integration Points [10]

	Managed Objects (GDMO/SMI)	Management Services (CMISE/SNMP)	Java Objects	Web Browser (HTTP/HTML)	TMN Agent

CORBA Objects	IP1		IP4	IP3	
CORBA Services		IP2			
TMN Manager					IP5

IP1 Provides mapping of objects defined in CORBA/IDL to managed objects defined in GDMO or SMI.
IP2 Provides mapping of appropriate CORBA Services to CMIS and SNMP services.
IP3 Provides a mapping of Web Browser technology access to CORBA objects (for situations where this may be needed as an addition to/replacement of Browser access to a database).
IP4 Provides a mapping between Java based objects and CORBA objects.
IP5 Provides a high level convenient programming interface for the rapid development of TMN based manager/agent interactions. It also provides a convenient point of integration if it is necessary to separate out the two sides of the manager/agent interface from the point of view of technology selection. For example, allowing the manager role to perhaps be supported in a Web-based environment, but giving a good point of integration with a TMN based agent.

Annex A (informative): Technology considerations

A.1 TMN physical blocks

TMN functions can be implemented in a variety of physical configurations (M.3010 [1]). The relationship of the functional blocks to physical equipment is shown in Table A.1 which names the TMN physical blocks according to the set of function blocks which each is allowed to contain. For each physical block there is a function block which is characteristic of it and is mandatory for it to contain. There also exist other functions, which are optional for the physical blocks to contain. Table A.1 does not imply any restriction of possible implementations, but defines those identified within this annex.

The subclauses below give the definitions for consideration in implementation schemes.

Table A.1: Relationship of TMN physical block names to TMN function blocks

(Note 2 & Note 3)	NEF	TF	OSF	WSF
NE	M	O	O	O (Note 3)
QA, XA, QM, XM		M		
OS		O	M	O
WS				M

M Mandatory
O Optional

NOTE 1: Within this table, where more than one name is possible, the choice of the physical block name is determined by the predominant usage of the block.
NOTE 2: TMN physical blocks may contain additional functionality, which allows them to be managed.
NOTE 3: For the WSF to be present the OSF shall also be present. This means that the WSF shall address an OSF. The local man-machine access is not considered part of the TMN.

A.1.1 Operations System (OS)

The OS is the system, which performs OSFs. The OS may optionally provide QAFs and WSFs.

A.1.2 Transformation

Transformation provides conversion between different protocols and data formats for information interchange between physical blocks. There are two types of transformation: adaptation and mediation that can apply at q or x reference points.

A.1.2.1 Adaptation device

An Adaptation Device (AD), or adapter, provides transformation between a non-TMN physical entity to a NE to OS within a TMN. A Q-adapter (QA) is a physical block used to connect NE-like or OS-like physical blocks with non-TMN compatible interfaces (at m reference points) to Q interfaces. An X-adapter (XA) is a physical block used to connect non-TMN physical entities having a non-TMN communication mechanism in a non-TMN environment to an OS at the edge of a TMN.

A.1.2.2 Mediation Device (MD)

A Mediation Device (MD) provides transformation between TMN physical blocks that incorporate incompatible communication mechanisms. A Q-Mediation Device (QMD) is a physical block that supports connections within one TMN. An X-Mediation Device (XMD) is a physical block that supports connections of OSs in different TMNs.

A.1.3 Network Element (NE)

The NE is comprised of telecommunication equipment (or groups/parts of telecommunication equipment) and support equipment or any item or groups of items considered belonging to the telecommunications environment that performs NEFs. The NE may optionally contain any of the other TMN function blocks according to its implementation requirements. The NE has one or more standard Q-type interfaces and may optionally have F and X interfaces.

Existing NE-like equipment that does not possess a TMN standard interface will gain access to the TMN via a Q Adapter Function, which will provide the necessary functionality to convert between a non-standard and standard management interface.

NEs may be distributed or centralised. Various parts of a NE are not geographically constrained to one physical location. For example, the parts may be distributed along a transmission system. An example of a distributed NE is illustrated in Figure A.1 (M.3013 [11]).

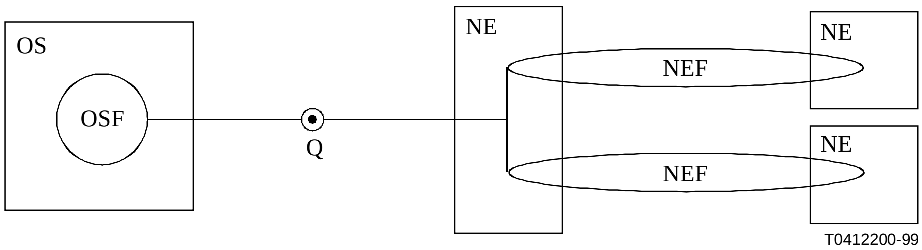


Figure A.1: Distributed network element

A.1.4 Workstation (WS)

The WS is the system, which performs WSFs. The workstation functions translate information at the f reference point to a displayable format at the g reference point, and vice versa.

If equipment incorporates other TMN functionality as well as the WSF, then it is named by one of the other names in Table A.1

A.1.5 Data Communication Network (DCN)

A DCN supporting a TMN has traditionally conformed to the network service of the OSI reference model for ITU-T applications as specified in Recommendation X.200 [4]. ITU-T Recommendation X.25 has been a commonly used packet protocol. However, the evolution of telecommunication services is merging circuit-switched and packet-switched modes with advancing technologies of ISDN, ATM, SDH, and the Internet. A variety of telecommunications services can be employed as long as integrity of information transfer can be preserved.

Within a TMN, the necessary physical connection, such as circuit-switched or packet-switched, may be offered by communication paths constructed with various network components, including dedicated lines, X.25 packet-switched data network, ISDN, common channel signalling network, public-switched telephone network, local area networks, terminal controllers, etc. The facilities can be either dedicated to a DCN or shared resources (for example, using SS No. 7 or an existing X.25 or IP-based packet-switched network).

Equipment supporting an OSF shall provide for two modes of data communication. These are spontaneous transmission of messages (e.g. for the NEF to the OSF) and a two-way dialogue (e.g. as the OSF obtains supporting information from the NEF and sends commands to the NEF or transfer messages to or from another OSF). In addition, an OSF is responsible for assuring the integrity of the data channels through a DCN. Physical connectivity in a local environment may be provided by a variety of subnetwork configurations including point-to-point, star, bus or ring.

The DCN may consist of a number of individual subnetworks of different types, interconnected together. The DCN may be a local path or a wide-area connection among distributed physical blocks. The DCN is technology independent and may employ any single or combination of transmission technologies.

A.1.6 TMN logical layered architecture within the TMN physical architecture

Four specialisations of the OS physical block are defined to support a physical realisation of function blocks in logical layers. The four specialised OS physical blocks are the Business (B-OS), the Service (S-OS), the Network (N-OS) and the Element (E-OS) Operations Systems. These physical blocks are named according to the predominant function block they contain. Specifically, B-OS, S-OS, N-OS and E-OS predominantly contain B-OSF, S-OSF, N-OSF and E-OSF respectively. When physical blocks contain more than one kind of specialised OS function block that provide substantial functionality to the physical block, thus spanning more than one logical layer, the physical block is named according to the highest hierarchically layered function block. For example, a physical block containing both N-OSF and E-OSF, providing substantial network functionality is called an N-OS.

The exchange of management information between logical layers employs the managing roles and managed roles of the TMN interaction model. This allows management activities to be clustered into layers and to be decoupled. The managed roles will be associated with a set of information elements from information model(s) exposing a view at the layer's level of abstraction (e.g. equipment, element, network, service, etc.). Generally, managing and managed roles may be placed in logical layers without restriction. A managed role may be associated with a set of information elements from any layer. Managed roles may be placed in any layer and invoke operations associated with any other managed roles.

A.1.7 Interoperable interface concept

In order for two or more TMN physical blocks to exchange management information they shall be connected by a communications path and each element shall support the same interface onto that communications path.

It is useful to use the concept of an interoperable interface to simplify the communications problems arising from a multi-vendor, multi-capability network.

The interoperable interface defines the protocol suite and the messages carried by the protocol. Transaction-oriented interoperable interfaces are based upon an object-oriented view of the communication and therefore, all the messages carried deal with object manipulations. It is the formally defined set of protocols, procedures, message formats and semantics used for the management communications.

The message component of the interoperable interface provides a generalised mechanism for managing the objects defined for the information model. As part of the definition of each object there is a list of management operations types which are valid for the object. In addition, there are generic messages that are used identically for many classes of managed objects.

In the architecture, what predominantly distinguishes one interface from another is the scope of the management activity that the communication at the interface shall support. This common understanding of the scope of operation is termed Shared Management Knowledge. Shared Management Knowledge includes an understanding of the information model of the managed network (object classes supported, functions supported, etc.), management support objects, options, application context supported, etc. The Shared Management Knowledge ensures that each end of the interface understands the exact meaning of a message sent by the other end.

A.2 TMN standard interfaces

Figure A.2 shows an example of a physical architecture. It represents each of the functions as physical blocks and illustrates how a number of interfaces might share communication paths within a given TMN physical architecture.

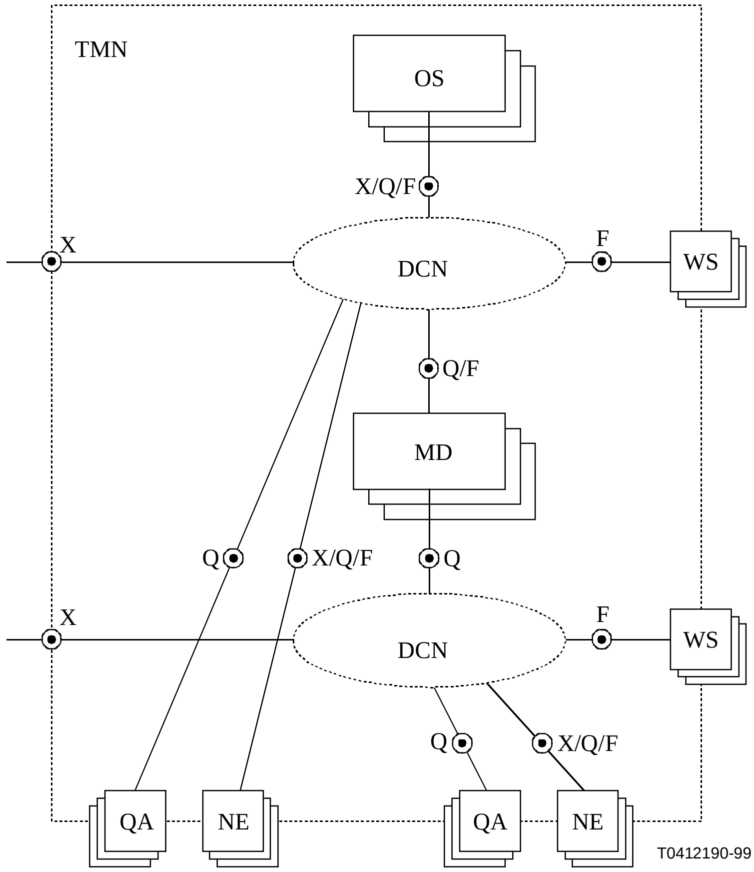


Figure A.2: Examples of interfaces for the TMN physical architecture M.3010 [1]

Figure A.2 shows the interconnection of the various TMN physical blocks by a set of standard interoperable interfaces. The allowable interconnections of these standard interfaces within a given TMN may be controlled by both the actual interfaces provided and/or by security and routing restrictions provided within the various physical block entities (e.g. passwords, log-ons, DCN routing assignment, etc.).

TMN standard interfaces are defined corresponding to the reference points. They are applied at these reference points when external physical connections to them are required.

There is a family of protocol suites for each of the TMN interfaces: Q, X and F. The choice of the protocol is dependent on the implementation requirements of the physical configuration.

A.2.1 Q interface

The Q interface is applied at q reference points.

To provide flexibility of implementation, the class of Q interfaces is made up of the following subclasses:

- the interface Q is applied at the q reference point;
- the Q interface is characterised by that portion of the information model shared between the OS and those TMN elements to which it directly interfaces.

A.2.2 F interface

The F interface is applied at f reference points. The F interfaces connecting workstations to the TMN building blocks containing OSFs or MFs through a data communication network are included in this Recommendation. Connections of implementation specific, WS-like entities to OSs or NEs, are not subject of this Recommendation.

A.2.3 X interface

The X interface is applied at the x reference point. It will be used to interconnect two TMNs or to interconnect a TMN with other network or systems which accommodates a TMN-like interface. As such, this interface may require increased security over the level, which is required by a Q-type interface. It will therefore be necessary that aspects of security are addressed at the time of agreement between associations, e.g. passwords and access capabilities.

The information model at the X interface will set the limits on the access available from outside the TMN. The set of capabilities made available at the X interface for access to the TMN will be referred to as TMN Access.

Additional protocol requirements may be required to introduce the level of security, non-repudiation, etc. which is required.

A.2.4 Relationship of TMN interfaces to TMN physical blocks

Table A.1 defines the possible interfaces, which each named TMN physical block can support. It is based upon the function blocks which Table A.1 associates with each physical block and the reference points between function blocks.

Annex B (informative): Overview of a UMTS Network

Figure B.1 presents an example of a UMTS network, related management areas and introduces some management interfaces. UMTS Service specific entities are not shown.

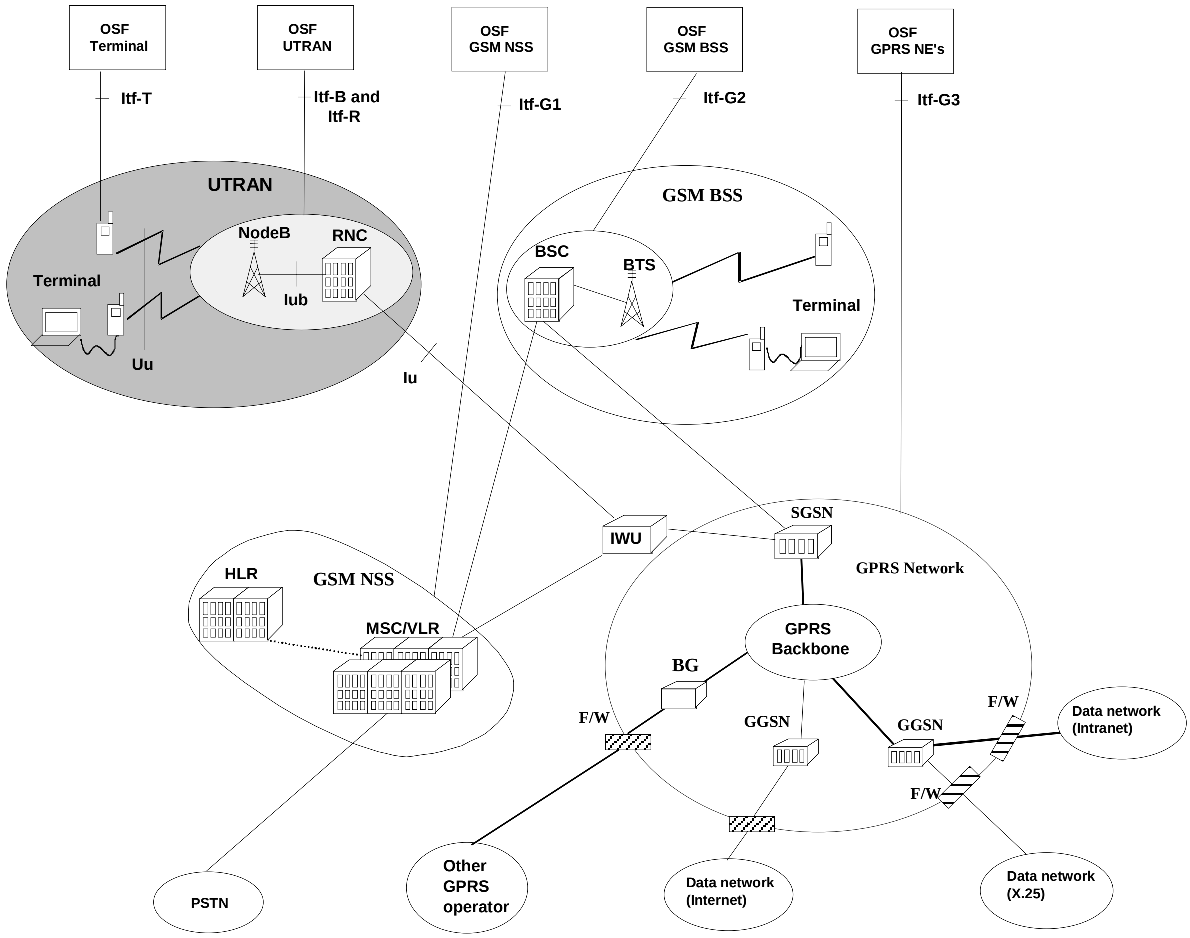


Figure B.1: Overview of a UMTS Network, showing management interfaces and management areas

All the following interfaces are illustrated in figure B.1:

- Itf-T between a terminal and a NE Manager. This interface will in some extent manage the 3G terminal and the USIM of the subscriber. Requirements of this interface are for further study.
- Itf-B and Itf-R between UTRAN and a NE Manager.
- Itf-G1 between GSM NSS and NE Manager.
- Itf-G2 between GSM BSS and NE Manager. This interface is standardised in GSM 12-series specifications.
- Itf-G3 between GPRS NEs and a NE Manager.

Annex C (informative): Information Service template

This annex contains the template to be used for the Information Services documents produced within the 3GPP SA TSG5. This template is based on the latest 3GPP template which **must** be used for any 3GPP Technical Specification.

The introductory clauses of the 3GPP template (from clause 1 to clause 3) are unchanged.

This template is numbered starting with "X" which, in general should correspond to 4 which is the beginning of the main text document. However, if there is a need for a specific IS to introduce additional clauses in the body X may correspond to a number higher than 4. For an NRM only clause X shall be used.

The conclusive clauses/annexes of the 3GPP template are unchanged.

X Information Object Classes

-- 'X' represents a number

X.1 Information entities imported and local labels

-- this clause identifies a list of information entities (e.g. information object class, information relationship, information attribute) that have been defined in other specifications and that are imported in this specification. This includes information entities from other specifications imported for inheritance purpose. Each element of this list is a pair (label reference, local label).The label reference contains the name of the specification where it is defined, the type of the information entity and its name. The local label of imported information entities can then be used throughout the specification instead of the label reference.

-- this information is provided in a table. An example of such a table is given here below :

Label reference	Local label
32.106-5 [10], information object class, Top	Top

X.2 Class diagram

X.2.1 Attributes and relationships

-- this first diagram represents all information object classes defined in this IS with all their relationships and all their attributes. This diagram shall contain relationship names, role name and role cardinality. This shall be a UML compliant class diagram.

-- Characteristics (attributes, relationships) of imported information object classes need not to be repeated in the diagram. Names of information elements (class, attribute) defined in the IS and which scope is local to this IS must be prefixed by a 3 characters prefix uniquely identifying the IS. Information object classes should be defined using the stereotype <<InformationObjectClass>>. On the class diagram, each attribute in an information object class shall be qualified as "protected" by the addition of a symbol "#" before each attribute.

X.2.2 Inheritance

-- this second diagram represents the inheritance hierarchy of all information object classes defined in this IS. This diagram does not need to contain the complete inheritance hierarchy but shall at least contain the parent information object classes of all information object classes defined in this specification. By default, an information object class inherits from the information object class "top". This shall be a UML compliant class diagram.

-- Characteristics (attributes, relationships) of imported information object classes need not to be repeated in the diagram. Information object classes should be defined using the stereotype <<InformationObjectClass>>.

-- Note : some inheritance relationships presented in X.2.2 can be repeated in X.2.1 to enhance readability.

X.3 Information object classes definition

-- each information object class is defined using the following structure :

X.3.a InformationObjectClassName

-- InformationObjectClassName is the name of the information object class

-- 'a' represents a number, starting at 1 and increasing by 1 with each new definition of an information object class

X.3.a.1 Definition

-- The <definition> sub-clause is written in natural language. The <definition> sub-clause refers to the information object class itself. The characteristics related to the relationships that the object class can have with other object classes can't be found in the definition. The reader has to refer to relationships definition to find such kind of information. Information related to inheritance shall be precised here.

X.3.a.2 Attributes

-- The <attributes> sub-clause presents the list of attributes, which are the manageable properties of the object class . Each element is a tuple (attributeName, visibilityQualifier, supportQualifier, readQualifer, writeQualifer)

- The visibilityQualifier indicates whether the attribute is public, private or IRPAgent Internal ("+","—", and "%" respectively). The semantics of public and private are as per the UML specification. The semantic of IRPAgent Internal is defined within the 3GPP UML Repertoire.
- The supportQualifier indicates whether the attribute is Mandatory, Optional, Conditional or not supported ("M"," O"," C", or "—", respectively).
- The readQualifier indicates whether the attribute shall be readable by the IRPManager. The semantics for readQualifier is identical to supportQualifier, for "M, "O", and "—".
- The writeQualifier indicates whether the attribute shall be writeable by the IRPManager. The semantics for writeQualifier is identical to supportQualifier, for "M", "O", and "—".

-- There is a dependency relationship between the supportQualifier and visibilityQualifier, readQualifier, and writeQualifier. The supportQualifier indicates the requirements for the support of the attribute. For any given attribute, regardless of the value of the supportQualifier, at least one of the reqdQualifier or writeQualifier must be "M". The implication of the "O" supportQualifier is that the attribute is optional, however the read and write qualifiers indicate how the optional attribute shall be supported, should the optional attribute be supported. Regardless of the supportQualifier, if an attribute is supported then it shall be supported in accordance with the specified visibilityQualifier.

-- Private or IRPAgent Internal attributes are per definition not readable by the IRPManager. Their readQualifier is hence always "—".

-- Private or IRPAgent Internal attributes are per definition not writable by the IRPManager. Their writeQualifier is hence always "—".

-- The readQualifier and writeQualifier of a supported atrribute, that is public, may not be both "—".

-- The use of "—" in supportQualifier is reserved for documenting support of attributes defined by an «Archetype» IOC. Attributes with a supportQualifier of "—" are not implemented by the IOC that is realizing a subset of the attributes defined by the «Archetype». The readQualifier and writeQualifier are of no relevance in this case. However, a not supported attribute is neither readable nor writable. For this reason the readQualifier and writeQualifier shall be "—" for unsupported attributes.

-- For any IOC that uses one or more attributes from an «Archetype», a separate table shall be used to indicate the supported attributes. This table is absent if no «Archetype» attributes are supported. For example, if a particular IOC has defined attributes (i.e., attributes not defined by an «Archetype») and encapsulates attributes from two «Archetype»s, then the totality of the attributes of said IOC will be contained in three separate tables.

-- This information is provided in a table. An example of such a table is given below:

Attribute name	Visibility	Support Qualifier	Read Qualifer	Write Qualifier
ntfSubscriptionId	+	M	M	O

-- Another example, where the support qualifier is "O" is given here below:

Attribute name	Visibility	Support Qualifier	Read Qualifer	Write Qualifier
ntfSubscriptionId	+	O	M	O

--In this example, the ntfSubscriptionId is an optional attribute. If the implementation chose to support ntfSubscriptionId, then the said implementation is required to support read and may support write.

- Note: this sub-clause does not need to be present when there is no attribute to define.

X.3.a.3 Attribute constraints

-- The <attribute constraints> sub-clause presents constraints between attributes that are always held to be true.). Those properties are always held to be true during the lifetime of the attributes and in particular don't need to be repeated in pre or post conditions of operations or notifications.

- *Note : this sub-clause does not need to be present when there is no attribute constraints to define.*

X.3.a.4 Relationships

-- The <relationship> sub-clause presents the list of relationships in which this class in involved. Each element is a relationshipName.

- *Note : this sub-clause is optional and may be avoided since all relationships are represented in the class diagram in clause.X.2.1.*

X.3.a.5 State diagram

-- The <state diagram> sub-clause contains state diagrams. A state diagram of an information object class defines permitted states of this information object class and the transitions between those states. A state is expressed in terms of individual attribute values or a combination of attribute values or involvement in relationships of the information object class being defined. This shall be a UML compliant state diagram.

X.4 Information relationships definition

-- each information relationship is defined using the following structure :

X.4.a InformationRelationshipName (supportQualifier)

-- InformationRelationshipName is the name of the information relationship followed by a qualifier indicating whether the relationship is Mandatory, Optional or Conditional (M, O, C)

-- 'a' represents a number, starting at 1 and increasing by 1 with each new definition of an information relationship

X.4.a.1 Definition

-- The <definition> sub-clause is written in natural language.

X.4.a.2 Roles

-- The <roles> sub-clause identifies the roles played in the relationship by object classes.. Each element is a pair (roleName, roleDefinition)

-- this information is provided in a table. An example of such a table is given here below :

Name	Definition
isSubscribedBy	This role represents the one who has subscribed

X.4.a.3 Constraints

-- The <constraints> sub-clause contains the list of properties specifying the semantic invariants that must be preserved on the relationship. Each element is a pair (propertyName, propertyDefinition). Those properties are always held to be true during the lifetime of the relationship and don't need to be repeated in pre or post conditions of operations or notifications.

-- this information is provided in a table. An example of such a table is given here below :

Name	Definition
inv_notificationCategoriesAllDistinct	"the notification categories contained in the ntfNotificationCategorySet attribute of ntfSubscription playing the role hasSubscription are all distinct from each other"

X.5 Information attributes definition

-- each information attribute is defined using the following structure :

X.5.1 Definition and legal values

-- This sub-clause contains for each attribute being defined its name, its definition written in natural language and a list of legal values supported by the attribute.

-- In the case where the legal values can be enumerated, each element is a pair (legalValueName, legalValueDefinition), unless a legalValueDefinition applies to several values in which case the definition is provided only once. When the legal values cannot be enumerated, the list of legal values is defined by a single definition.

-- this information is provided in a table. An example of such a table is given here below :

Attribute Name	Definition	Legal Values
ntfSubscriptionId	It identifies uniquely a subscription	N/A
ntfSusbcriptionState	It indicates the activation state of a subscription	"suspended" : the subscription is suspended "notSuspended" : the subscription is active

X.5.2 Constraints

-- The <constraints> sub-clause indicates whether there are any constraints affecting attributes. Each constraint is defined by a pair (propertyName, propertyDefinition). PropertyDefinitions are expressed in natural language.

-- An example is given here below :

Name	Definition
inv_TimerConstraints	"ntfTimeTickTimer is lower than or equal to ntfTimeTick"

X.6 Particular information configurations

-- some configurations of information are special or complex enough to justify the usage of a state diagram to clarify them. A state diagram in this clause defines permitted states of the system and the transitions between those states. A state is expressed in terms of a combination of attribute values constraints or involvement in relationships of one or more information object classes.

Y Interface Definition

-- 'Y' represents a number, immediately following 'X'

Y.1 Class diagram representing interfaces

-- each interface is defined in the diagram. This shall be a UML compliant class diagram.

-- Interfaces are defined using a stereotype <<Interface>>. Each interface contains a set of either operations or notifications which are mandatory or either a single operation or a single notification which is optional. The support of an interface by an information object class is represented by a relationship between the 2 entities with a cardinality (1..1) if all the operations or notifications contained in the interface are mandatory, and (0..1) if the operation or notification contained in the interface is optional. On the class diagram, each operation and notification in an interface shall be qualified as "public" by the addition of a symbol "+" before each operation and notification.

Y.2 Generic rules

-- the following rules are relevant for all IS. They shall simply be copied as part of the template.

- rule 1 : each operation with at least one input parameter supports a pre-condition valid_input_parameter which indicates that all input parameters shall be valid with regards to their information type. Additionally, each such operation supports an exception operation_failed_invalid_input_parameter which is raised when pre-condition valid_input_parameter is false. The exception has the same entry and exit state.

- rule 2 : Each operation with at least one optional input parameter supports a set of pre-conditions supported_optional_input_parameter_xxx where "xxx" is the name of the optional input parameter and the pre-condition indicates that the operation supports the named optional input parameter. Additionally, each such operation supports an exception operation_failed_unsupported_optional_input_parameter_xxx which is raised when (a) the pre-condition supported_optional_input_parameter_xxx is false and (b) the named optional input parameter is carrying information. The exception has the same entry and exit state.

- rule 3 : each operation shall support a generic exception operation_failed_internal_problem which is raised when an internal problem occurs and that the operation cannot be completed. The exception has the same entry and exit state.

Y.b InterfaceName Interface

-- InterfaceName is the name of the interface

-- 'b' represents a number, starting at 3 and increasing by 1 with each new definition of an interface

-- Each interface is defined by its name and by a sequence of operations or notifications as defined herebelow.

-- each operation is defined using the following structure :

Y.b.a Operation OperationName (supportQualifier)

-- OperationName is the name of the operation followed by a qualifier indicating whether the operation is Mandatory, Optional or Conditional (M, O, C)

-- 'a' represents a number, starting at 1 and increasing by 1 with each new definition of an operation

Y.b.a.1 Definition

-- The <definition> sub-clause is written in natural language.

Y.b.a.2 Input parameters

-- list of input parameters of the operation. Each element is a tuple (inputParameterName, supportQualifier, InformationType, inputParameterComment)

-- this information is provided in a table. An example of such a table is given here below :

Parameter Name	Qualifier	Information type	Comment
managerReference	M	ntfSubscriber.ntfManagerReference	It specifies the reference of IRPManager to which notifications shall be sent.

Y.b.a.3 Output parameters

-- list of output parameters of the operation. Each element is a tuple (outputParameterName, supportQualifier, MatchingInformation, outputParameterComment)

-- this information is provided in a table. An example of such a table is given here below :

Parameter Name	Qualifier	Matching Information	Comment
versionNumberSet	M	notificationIRP.irpversion	It indicates one or more SS version numbers supported by the notificationIRP.

Y.b.a.4 Pre-condition

-- a pre-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The pre-condition must be held to be true before the operation is invoked .. An example is given here below :

notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed

-- Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the pre-condition are provided in a table. An example of such a table is given here below :

Assertion Name	Definition
notificationCategoriesNotAllSubscribed	"at least one notificationCategory identified in the notificationCategories input parameter is supported by IRPAgent and is not a member of the ntfNotificationCategorySet attribute of an ntfSubscription which is involved in a subscription relationship with the ntfSubscriber identified by the managerReference input parameter".
notificationCategoriesParameterAbsentAndNotAllSubscribed	" notificationCategories input parameter is absent and at least one notificationCategory supported by IRPAgent is not a member of the ntfNotificationCategorySet attribute of an ntfSsubscription which is involved in a subscription relationship with the ntfSubscriber identified by the managerReference input parameter"

Y.b.a.5 Post-condition

-- a post-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The post-condition must

be held to be true after the completion of the operation. When nothing is said in a post-condition regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the pre-condition. An example is given here below :

subscriptionDeleted OR allSubscriptionDeleted

-- Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the post-condition are provided in a table. An example of such a table is given here below :

Assertion Name	Definition
subscriptionDeleted	"the ntfSubscription identified by subscriptionId input parameter is no more involved in a subscription relationship with the ntfSubscriber identified by the managerReference input parameter and has been deleted. If this ntfSubscriber has no more ntfSubscription, it is deleted as well."
allSubscriptionDeleted	"in the case subscriptionId input parameter was absent, the ntfSubscriber identified by the managerReference input parameter is no more involved in any subscription relationship and is deleted, the corresponding ntfSubscription have been deleted as well."

Y.b.a.6 Exceptions

-- list of exceptions that can be raised by the operation. Each element is a tuple (exceptionName, condition, ReturnedInformation, exitState))

Y.b.a.6.c exceptionName

-- exceptionName is the name of an exception

-- 'c' represents a number, starting at 1 and increasing by 1 with each new definition of an exception

-- this information is provided in a table. An example of such a table is given here below :

Exception Name	Definition
Ope_failed_existing_subscription	**Condition:** (notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed) not verified **Returned information:** output parameter status is set to OperationFailedExistingSubscription **Exit state:** Entry State

-- each notification is defined using the following structure :

Y.b.a Notification NotificationName (supportQualifier)

-- NotificationName is the name of the notification followed by a qualifier indicating whether the notification is Mandatory, Optional or Conditional (M, O, C).

-- 'a' represents a number, starting at 1 and increasing by 1 with each new definition of a notification

Y.b.a.1 Definition

-- The <definition> sub-clause is written in natural language.

Y.b.a.2 Input parameters

-- list of input parameters of the notification. Each element is a tuple (inputParameterName, supportQualifier and filteringQualifier, matchingInformation, inputParameterComment)

-- the filteringQualifier indicates whether the parameter of the notification can be filtered or not. Values are Yes (Y) or No (N). The matchingInformation refers to information in the state "toState".

-- this information is provided in a table. The column "Qualifiers" contains the two qualifiers supportQualifier and filteringQualifier separated by a comma. An example of such a table is given here below :

Parameter Name	Qualifiers	Matching Information	Comment
managerReference	M,Y	ntfSubscriber.ntfManagerReference	It specifies the reference of IRPManager to which notifications shall be sent.

Y.b.a.3 Triggering event

-- the triggering event for the notification to be sent is the transition from the information state defined by the 'from state' sub-clause to the information state defined by the 'to state' sub-clause.

Y.b.a.3.1 From state

-- this sub-clause is a collection of assertions joined by AND, OR, and NOT logical operators. An example is given herebelow :

alarmMatched AND alarmInformationNotCleared

-- Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "from state" are provided in a table. An example of such a table is given here below :

Assertion Name	Definition
alarmMatched	The newly generated network alarm matches with one AlarmInformation (same values for eventType, probableCause, specificProblem attributes) in AlarmList.
alarmInformationNotCleared	The perceivedSeverity attribute of the matched AlarmInformation is not cleared

Y.b.a.3.2 To state

-- this sub-clause is a collection of assertions joined by AND, OR and NOT logical operators. When nothing is said in a to-state regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the from state. An example is given here below :

resetAcknowledgementInformation AND perceivedSeverityUpdated

-- Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "to state" are provided in a table. An example of such a table is given here below :

Assertion Name	Definition
resetAcknowledgementInformation	The matched AlarmInformation identified in inv_alarmMatched in pre-condition has been updated according to the following rule : ackTime, ackUserId and ackSystemId are updated to contain no information; ackState is updated to "unacknowledged";
perceivedSeverityUpdated	The perceivedSeverity attribute of matched AlarmInformation identified in inv_alarmMatched in pre-condition has been updated.

Z Scenario

-- 'Z' represents a number, immediately following 'Y'

-- list of sequence diagrams each describing a possible scenario. This shall be a UML compliant sequence diagram. This is an optional clause.

Annex D (informative): Example of inheritance between ISs

Figure D.1 below illustrates the architecture defined in clause 10.6 with a simplified example. Note that this figure is for illustration only.

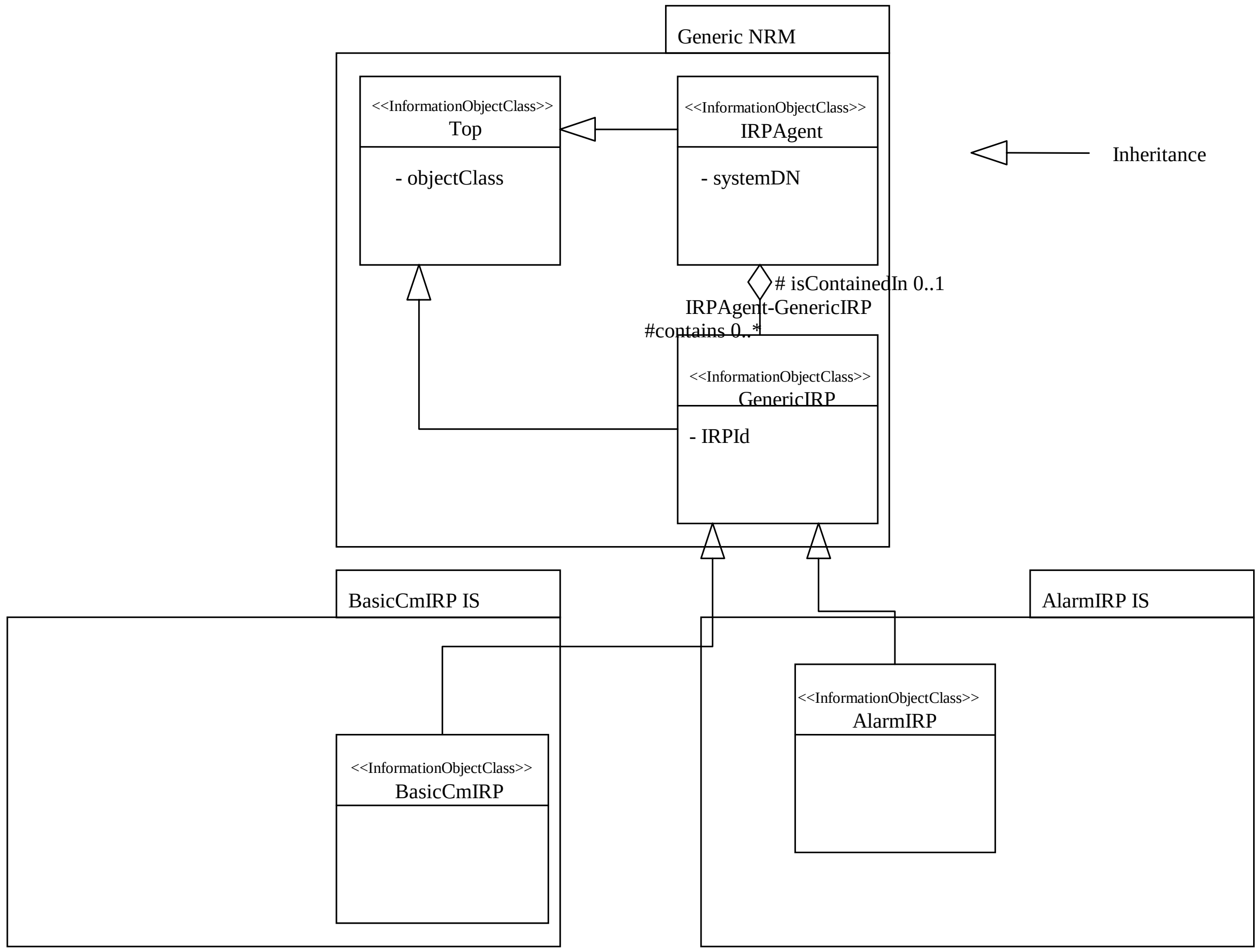


Figure D.1: Example of possible packages together with information object classes and their inter-relationships

The following aspects are illustrated in figure D.1 above:

1. Information object classes that are common to all Network Resources Models / some Information Services are captured in the GenericNRM package : Top, IRPAgent, GenericIRP, together with their attributes and relationships;

2. The information object class BasicCmIRP is defined in the BasicCmIRP IS package. As illustrated in the previous figure, this package imports the GenericNRM package;

3. The information object class AlarmIRP is defined in the AlarmIRP IS package. As illustrated in the previous figure, this package imports the GenericNRM package;

4. As a consequence, every information object class can inherit from the class Top, either directly or indirectly;

5. The IRPAgent class is defined in the GenericNRM;

6. A GenericIRP information object class is defined in the Generic NRM. It represents an abstraction of all the IRPs such as, e.g., BasicCmIRP or AlarmIRP. A containment relationship between IRPAgent and GenericIRP is defined;

7. Both the information object classes BasicCmIRP and AlarmIRP (defined in different Information Services) inherit from GenericIRP. As a first consequence, they inherit the attributes IRPId and IRPVersion (from GenericIRP) and objectClass (from Top). As a second consequence, both BasicCmIRP and AlarmIRP are contained in IRPAgent.

Annex E (informative): General rules for Solution Sets

E.1 Introduction

The intent of this annex is twofold. The first intent is for 3GPP internal use to document how a 3GPP Solution Set is produced and what it shall contain. The second intent with the annex is to give the reader of an Information Service (IS) or a Solution Set a better understanding on how to interpret the IS or Solution Set specifications.

E.2 Solution Set versioning

For further study.

E.3 Referenced Information Service Specification

A sentence shall be included in the clause "Scope" of all Solution Set specifications. The sentence shall read as follows:

"This Solution Set specification is related to Z"

where Z is the 3GPP Information Service specification number including the version, such as "3G TS 32.111-2 V4.1.X" for the case of Alarm Integration Reference Point: Information Service.

NOTE: that 'X', rather than the actual digit, is actually used in the sentence. This is because the value of X is not relevant for the reference purpose since different values of X identify different 3GPP published specifications that reflect only minor editorial changes.

Annex F (normative): Rules for CORBA Solution Sets

F.1 Introduction

The intent of this annex is threefold. The first intent is for 3GPP internal use to document how a 3GPP CORBA Solution Set is produced and how it is structured. The second intent with the annex is to give the reader or implementor of a CORBA Solution Set a better understanding on how to interpret the CORBA Solution Set document. The last and maybe most important intent is to put requirement on an implementor of a CORBA Solution Set.

It can be noted that it is expected that this annex is to be extended in later versions of this document.

F.2 Rules for specification of CORBA Solution Sets

F.2.1 Introduction

This clause identifies rules for specification of CORBA Solution Sets. This clause is mainly for 3GPP internal use. It is only for information for the implementor of a CORBA Solution Set.

F.2.2 Pragma prefix

All IDL-code shall define the pragma prefix using the following statement:

#pragma prefix "3gppsa5.org"

F.3 Implementation aspects of CORBA Solution Sets

F.3.1 Introduction

This clause identifies rules for the implementation of CORBA Solution Sets. This clause is normative for the implementor of a CORBA Solution Set.

F.3.2 IRPAgent behaviour on incoming optional method

The IRPAgent, claiming compliance to a particular SS version of a particular IRP such as the Alarm IRP, shall implement all mandatory and all optional methods. Each method implementation shall have a signature specifying all mandatory and all optional parameters.

- If the IRPAgent does not support a particular optional method, it shall throw the `OperationNotSupported` exception when the IRPManager invokes that method.
- If the IRPAgent have not implemented a particular method (because it is compiled with an IDL version that does not define the method), the CORBA ORB of the IRPAgent shall throw a system exception if the IRPManager invokes that method.

In all the above cases when an exception is thrown, the IRPAgent shall restore its state before the method invocation.

F.3.3 IRPAgent Behaviour on incoming optional parameter of operation

An IRPAgent must implement all optional parameters, as well as mandatory parameters, in all methods.

If the IRPAgent supports the implemented method but does not support its (one or more) optional input parameters, upon method invocation, the IRPAgent shall check if those parameters carry "no information" or absence semantics (defined later in sub-clause "Encoding rule for absence semantics"). If the check is negative, the IRPAgent shall throw the `ParameterNotSupported` exception with a string carrying the name of the unsupported optional parameter.

F.3.4 IRPAgent Behaviour on outgoing attributes of Notification

CORBA SS uses OMG defined structured event to carry notification. The structured event is partitioned into header and body.

The absence semantics of attribute in the header is realised by a string of zero length.

The body consists of one or more name-value pair attributes. The absence semantics of these attributes is realised by their absence.

For optional sub-attributes of an attribute carried by the name-value pair, their absence semantics is realised by the encoding rule of "absence semantics". See sub-clause E.3.5, "Encoding rule of absence semantics".

F.3.5 Encoding rule of absence semantics

The operation parameters are mapped to method parameters of CORBA SS. The absence semantics for an operation (input and output) parameter is method parameter type dependent.

- For a string type, if the parameter is specified as a string type, the absence semantics is a string of zero length. If the parameter is specified as a union structure (preferred), the absence semantics is conveyed via a FALSE Boolean value switch.
- For an integer type, if the parameter is specified as a signed, unsigned, long, etc type, the absence semantics is the highest possible positive number. If the parameter is specified as a union structure (preferred), the absence semantics is conveyed via a FALSE Boolean value switch.
- For a boxed valueType (supported by CORBA 2.3), it is the null value.

The notification parameters are mapped to attributes of the CORBA Structured Events. The absence semantics for a notification parameter is attribute position (within the Structured Event) dependent.

- For the fixed header of the Structured Event header, the absence semantics is realised by a string of zero length.
- For the filterable body fields of the Structured Event body, the absence semantics is realised by the absence of the corresponding attribute.

F.4 IDL file name rule

CORBA IDL uses "#include "X"" statement where X is a name of a file containing IDL statements. In the CORBA Solution Set, IDL statements are specified.

The rule defined here specifies

(a) the IDL statements that shall belong to one file; and

(b) the name of the file.

Rule: In the Annex where IDL statements are defined, use a special marker to indicate that a set of IDL statements shall be contained in one file. The name of the file shall be the name of the first IDL module of that set (of IDL statements). Multiple markers can be used.

Annex G (normative): IRP-IS UML Modelling Repertoire

G.1 Introduction

3GPP SA5 has chosen UML to capture systems behaviour in the IRP IS context.

UML provides a rich set of concepts, notations and model elements to model distributive systems. Usage of all UML notations and model elements is not necessary for the purpose of IRP IS specifications. This annex documents the necessary and sufficient set of UML notations and model elements, including the ones built by the UML extension mechanism <<stereotype>>, for use by 3GPP IRP IS authors. Collectively, this set of notations and model elements is called the 3GPP IRP IS modelling repertoire.

The selection of the UML notations and model elements in this repertoire is based on the needs of the existing 3GPP IRP IS specifications. Future IRP IS releases may require the use of additional UML notations or model elements.

IRP IS specifications shall employ the UML notation and model elements of this repertoire and may also employ other UML notation and model elements considered necessary. However, before any other UML notation and model elements may be employed in an approved 3GPP IRP specification, the other notation and model elements should be agreed for inclusion first in this repertoire.

All quotes are from [15].

Capitalized words are defined by various 3GPP IRP IS specifications or the reference [15].

G.2 Requirements

IRPAgent can be characterized by several different but related models. The models can be exterior or interior to the IRPAgent. Exterior models are use case models and interior models are object models.

Current version of this Annex focuses on the interior model aspects of IRPAgents.

The notation elements captured in this repertoire shall be used to model all aspects of NRM IRP IS (such as GERAN NRM IRP: IS) and (protocol) IRP (such as Alarm IRP: IS).

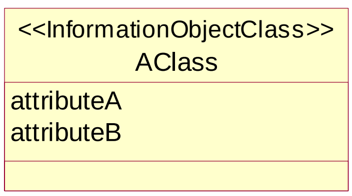
G.3 Model Elements and Notations

G.3.1 Basic model elements

UML defined a number of basic model elements. This subclause lists the selected subset for use in the repertoire. The semantics of the selected ones are defined in [15].

- attribute (Subclause 3.25 of [15]).

  This sample shows two attributes, listed as strings in the attribute compartment of the class AClass.

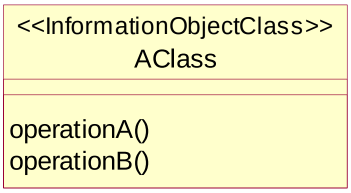


- aggregation (Subclause 3.43.2.5 of [15]).

This sample shows a hollow diamond attached to the end of a path to indicate aggregation. The diamond is attached to the class that is the aggregate.

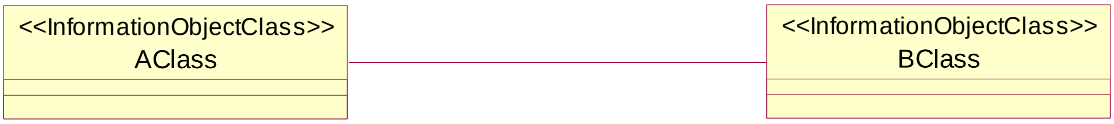
- operation (Subclause 3.26 of [15]).

This sample shows two operations, shown as strings in the operation compartment of class AClass, that the instance of AClass may be requested to perform. The operation has a name, e.g., operationA and a list of arguments (not shown).



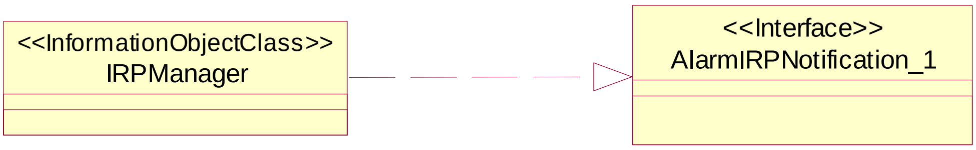
- association, association name (Subclause 3.41 of [15]).

This sample shows a binary association between exactly two model elements. The association can include the possibility relating a model element to itself. This sample shows a bi-directional association in that one model element is aware of the other. Association can be uni-directional (shown with an open arrow at one association end) in that only the source model element is aware of the target model element and not vice-versa.



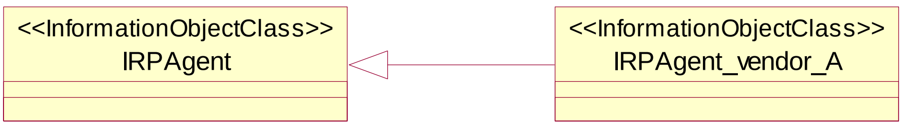
- realization relationship (Subclause 2.5.2.1 of [15]).

This sample shows the realization relationship between a AlarmIRPNotification_1 (the supplier) and a model element, IRPManager, that implements it.



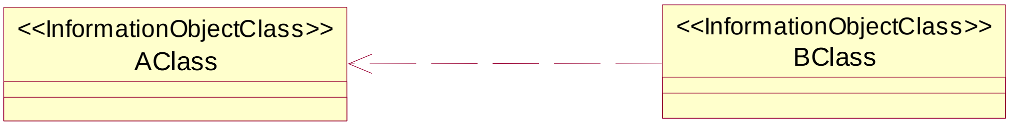
- generalization relationship (Subclause 3.50 of [15]).

This sample shows a generalization relationship between a more general element (the IRPAgent) and a more specific element (the IRPAgent_vendor_A) that is fully consistent with the first element and that adds additional information.



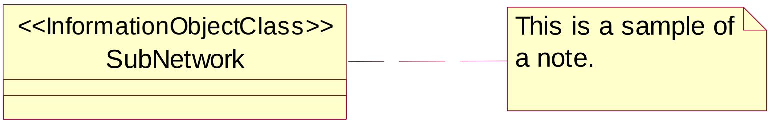
- dependency relationship (Subclause 3.51 of [15]).

This sample shows that BClass instances have a semantic relationship with AClass instances. It indicates a situation in which a change to the target element will require a change to the source element in the dependency.



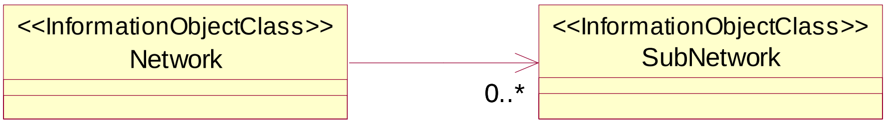
- note (Subclause 3.11 of [15])

  This sample shows a note, as a rectangle with a "bent corner" in the upper right corner. The note contains arbitrary text. It appears on a particular diagram and may be attached to zero or more modelling elements by dashed lines.



- Multiplicity, a.k.a. cardinality (Subclause 3.44 of [15]).

  This sample shows a multiplicity attached to the end of an association path. The meaning of this multiplicity is that one Network instance is associated with zero, one or more SubNetwork instances.



- rolename (Subclause 3.43.2.6 of [15]).

  This sample shows a Person (say instance John) is associated with a Company (say instance XYZ). We navigate the association by using the opposite association-end such as John.theCompany ="XYZ". Use noun for the rolename.

G.3.2 Stereotype

This sub-clause defines all allowable stereotypes that are summarized in the following table. Except <<Interface>>, <<Type>> and <<use>> (which are defined in [15]), all other stereotypes are extensions specifically designed for use in IRP IS specifications.

Table G.1: Stereotypes

Stereotype	Base Class
Interface	Class
Type	Class
ProxyClass	Class
Archtetype	Classifier (subclause 2.5.2.10 of [15])
InformationObjectClass	Classifier
use	Association
may use	Association
may realize	Association
emits	Association
names	Aggregation
%	VisibilityKind (subclause 2.7.2.29 of [15])

G.3.2.1 <<Interface>>

Subclause 2.5.2.25 of [15]:

"An interface is a named set of operations that characterize the behaviour of an element. In the metamodel, an Interface contains a set of Operations that together define a service offered by a Classifier realizing the Interface. A Classifier may offer several services, which means that it may realize several Interfaces, and several Classifiers may realize the same Interface.

…

Interfaces may not have Attributes, Associations, or Methods. An Interface may participate in an Association provided the Interface cannot see the Association; that is, a Classifier (other than an Interface) may have an Association to an Interface that is navigable from the Classifier but not from the Interface."

Subclause 2.5.4.6 of [15]: "The purpose of an interface is to collect a set of operations that constitute a coherent service offered by classifiers. Interfaces provided a way to partition and characterize groups of operations. An interface is only a collection of operations with a name. It cannot be directly instantiated. Instantiable classifiers, such as class or use case, may use interfaces for specifying different services offered by their instances. Several classifiers may realize the same interface. All of them must contain at least the operations matching those contained in the interface. The specification of an operation contains the signature of the operation (i.e., its name, the types of the parameters and the return type). An interface does not imply any internal structure of the realizing classifier. For example, it does not include which algorithm to use for realizing an operation. An operation may, however, include a specification of the effects [e.g., with pre and post-conditions] of its invocation."

G.3.2.1.1 Sample

This sample shows an AlarmIRPOperations_1 <<Interface>> that has two operations. The operation visibility is public (see definition of public visibility applicable to operation in subclause "visibility"). The input and output parameters of the operations are hidden (i.e., not shown). The AlarmIRP has a unidirectional mandatory realisation relationship with the <<interface>>.

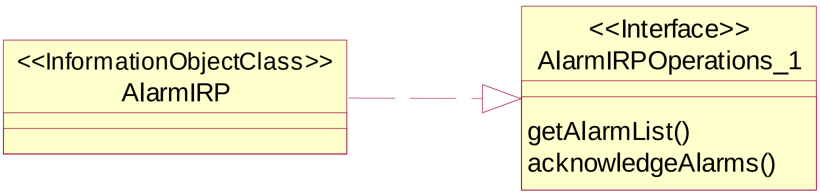


Figure G.1 : <<Interface>> Notation

G.3.2.2 <<Type>>

Subclause 3.28 of [15]: "[A Type is] a domain of objects together with the operations applicable to the objects, without defining the physical implementation of those objects. A Type may not contain any methods, maintain its own thread of control, or be nested. However, it may have Attributes and Associations. The Associations of a Type are defined solely for the purpose of specifying the behaviour of the Type's operations and do not represent the implementation of state data."

G.3.2.2.1 Sample

This sample shows the NotificationIRPNotification <<Type>> that specifies the five parameters (the notification header of Notification IRP). The AlarmIRPNotification_2 <<Interface>> depends (see the dependency relationship, a dashed open arrow line) on this <<Type>> for the construction of the notification emitted via the operation notifyChangedAlarm(). The visibility of attributes and operation in the example is public.

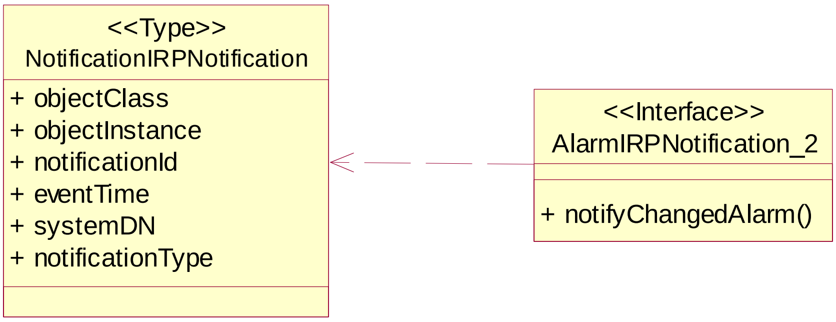


Figure G.2: <<Type>> Notation

G.3.2.3 <<ProxyClass>>

It is a form or template representing a number of <<InformationObjectClass>>. It encapsulates attributes, links, methods (or operations), and interactions that are present in the represented <<InformationObjectClass>>.

The semantics of a <<ProxyClass>> is that all behaviour of the <<ProxyClass>> are present in the represented <<InformationObjectClass>>. Since this class is simply a representation of other classes, this class cannot define its own behaviour other than those already defined by the represented <<InformationObjectClass>>.

A particular <<InformationObjectClass>> can be represented by zero, one or more <<ProxyClass>> or <<Archtype>>. For example, the ManagedElement <<InformationObjectClass>> can have MonitoredEntity <<ProxyClass>> and ManagedEntity <<ProxyClass>>.

The attributes of the <<proxyClass>> are accessible by the source entity that has an association with the <<ProxyClass>>.

G.3.2.3.1 Sample

This shows a <<ProxyClass>> named MonitoredEntity. It represents all NRM <<InformationObjectClass>> (e.g., GgsnFunction <<InformationObjectClass>>) whose instances are being monitored for alarm conditions. The MonitoredEntity plays the role of theMonitoredEntity.

Note that <<MonitoredEntity>> does not define attributeA. The attributeA is already defined by all <<InformationObjectClass>> represented by the <<MonitoredEntity>>, i.e., ClassA and ClassB.

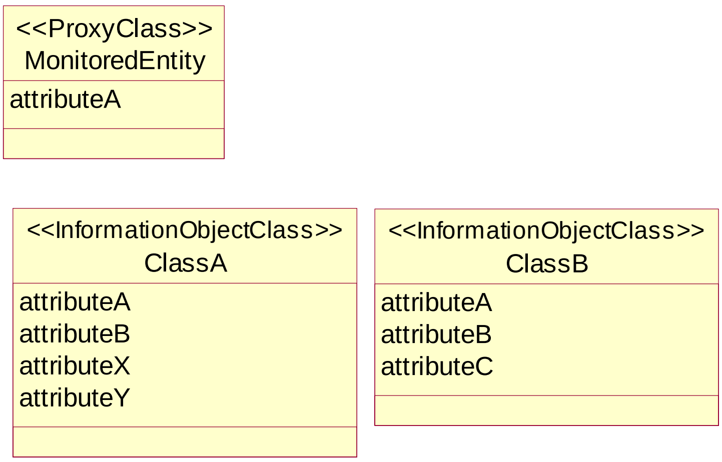


Figure G.3: <<ProxyClass>>

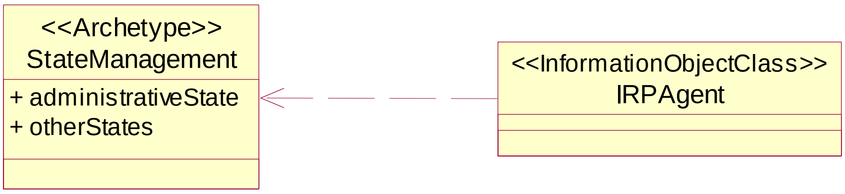
G.3.2.4 <<Archetype>>

It is a form or template representing a number of <<InformationObjectClass>>. It encapsulates attributes, links, operations, and interactions that are typical of the represented <<InformationObjectClass>>.

The semantics of an <<archetype>> is that all attributes, links operations and interactions encapsulated by the <<archetype>> may or may not be present in the represented <<InformationObjectClass>>. The <<Archetype>> represents a place holder class that is most useful in technology neutral analysis models that will require further specification and/or mapping within a more complete construction model.

G.3.2.4.1 Sample

This shows a <<Archetype>> named StateManagement. It also shows a <<InformationObjectClass>> IRPAgent that depends on this StateManagement. Note that the StateManagement has defined a number of attributes, the classes that depend on this StateManagement may or may not use all of the StateManagement attributes. In other words, at least one of the attributes of StateManagement is present in the IRPAgent. The precise set of StateManagement attributes used by the IRPAgent is specified in the IRPAgent specification.



G.3.2.5 <<InformationObjectClass>>

It is the descriptor for a set of network resources and network management capabilities. Each <<InformationObjectClass>> represents a set of instances with similar structure, behaviour and relationships.

This <<InformationObjectClass>> and other information classes such as <<interface>> are mapped into technology specific model elements such as GDMO Managed Object Class for CMIP technology. The mapping of IS modelling constructs to technology specific modelling constructs are captured in the corresponding IRP Solution Set specifications.

The name of a <<InformationObjectClass>> has scope within the 3GPP IRP IS document in which it is specified and the name must be unique among all <<InformationObjectClass>> names within that 3GPP IRP IS document. The IRP IS document name is considered in the similar way as the UML Package-name.

The <<InformationObjectClass>> is identical to UML *class* except that it does not include/define methods or operations.

Subclause 3.22.1 of [15]: "A *class* represents a concept within the system being modelled. Classes have data structure and behaviour and relationships to other elements."

G.3.2.5.1 Sample

This sample shows an AlarmList <<InformationObjectClass>>.

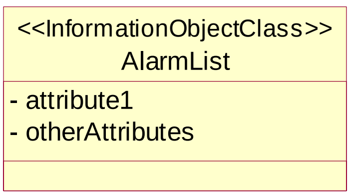


Figure G.4: <<InformationObjectClass>>> Notation

G.3.2.6 <<use>> and <<may use>>

The <<use>> and <<may use>> are unidirectional associations. The target must be an <<interface>>. The <<use>> states that the source class must have the capability to use the target <<interface>> in that it can invoke the operations defined by the <<interface>>. Support of the capability by the source entity is mandatory. The <<may use>> states that the source class may have the capability to use the target <<interface>> in that it may invoke the operations defined by the <<interface>>. Support of the capability by the source entity is optional.

The operations defined by the <<interface>> are visible across the itf-N.

G.3.2.6.1 Sample

This shows that the NotificationIRPAgent shall use the notifyNewAlarm and otherNotifications of AlarmIRPNotification_1 and may use the notifyChangedAlarm of AlarmIRPNotification_2.

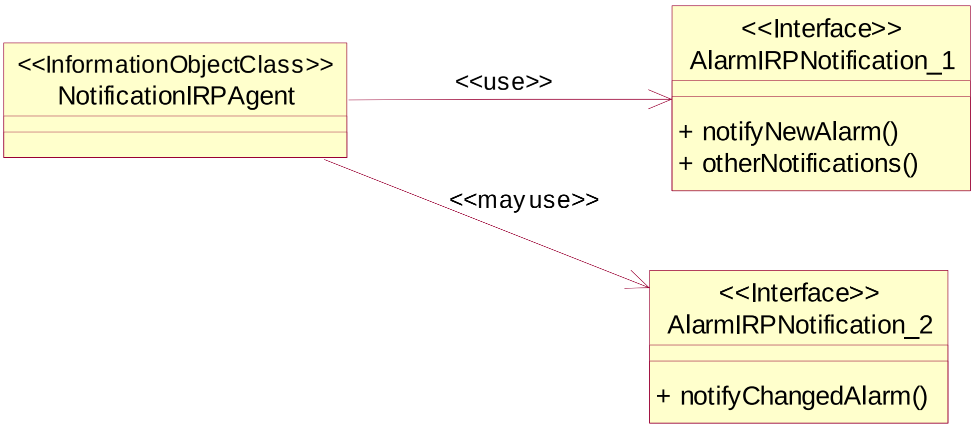


Figure G.5: <<use>> and <<may use>> Notation

G.3.2.7 Relationship realize and <<may realize>>

The relationship realize and <<may realize>> are unidirectional association. The target must be an <<interface>>. The relationship 'realize' shows that the source entity must realize the operations defined by the target <<interface>>. Realization of operations by the source entity is mandatory. The <<may realize>> shows the source entity may realize the operations defined by the target <<interface>>. Realization of the <<interface>> by the source entity is optional.

The operations defined by <<interface>> are visible across the itf-N.

G.3.2.7.1 Sample

This shows that the AlarmList shall realize (or support, implement) the two operations of AlarmIRPOperations_1 and may realize the operation of AlarmIRPOperations_2.

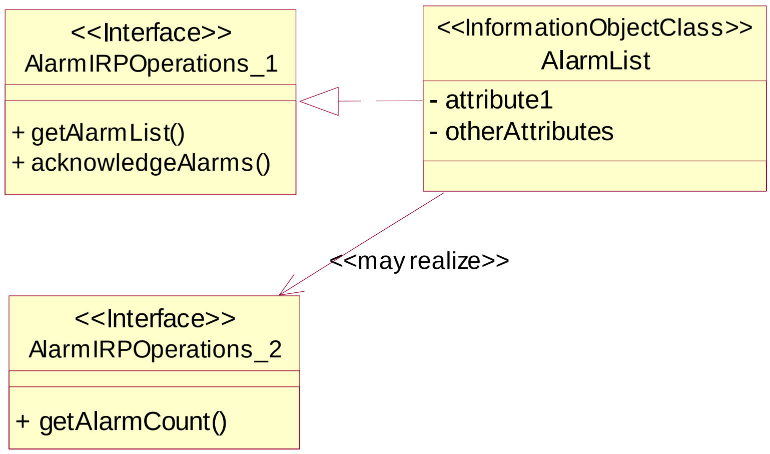


Figure G.6: Relationship realize and <<may realize>> Notations

G.3.2.8 <<emits>>

This is a unidirectional association. The source sends information to target. In the case that the target is NotificationIRPAgent, the information will then carry the semantics of 3GPP notification (e.g., notifyObjectCreation, notifyNewAlarm) such that the target NotificationIRPAgent can construct the relevant 3GPP notification for reception by the NotificationIRPManager.

The visibility of the information passed by <<emits>> is always "IRPAgent Internal" (see subclause on "Visibility").

G.3.2.8.1 Sample

This shows the MonitoredEntity (e.g., a GgsnFunction instance) emits notifications that are received by the NotificationIRPAgent. The emission is not visible across the itf-N.

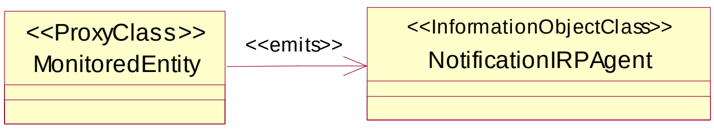


Figure G.7: <<emits>> Notation

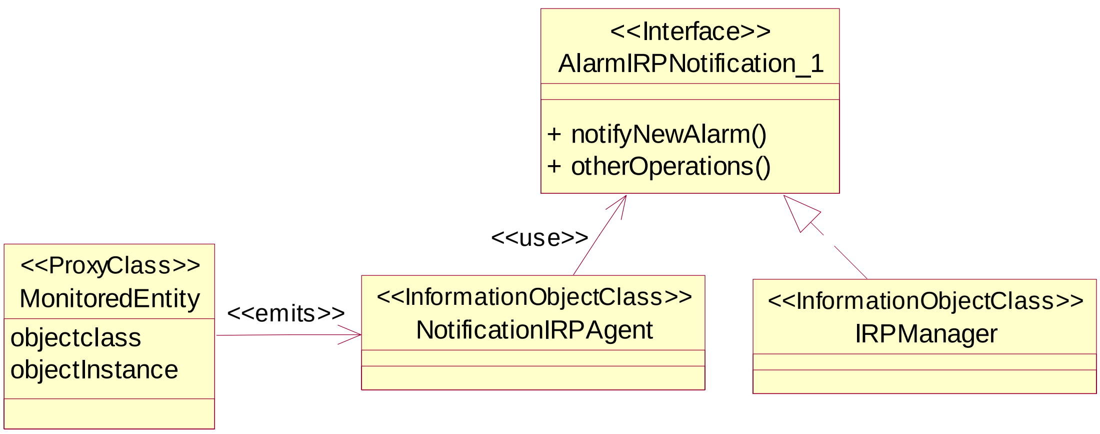


Figure G.8: <<use>>, <<emits>> and realize relationship Notation

G.3.2.9 <<names>>

It specifies a unidirectional aggregation. The target instance is uniquely identifiable, within the namespace of the source entity, among all other targeted instances of the same target classifier and among other targeted instances of other classifiers that has the same <<name>> aggregation with the source.

A source can have multiple <<names>> with multiple targets. The set of <<names>> used between the source and its targets forms the source namespace.

A target can have multiple <<names>> with multiple sources, i.e., a target can participate/belong to multiple namespaces.

By convention, the name of the attribute in the target model element to hold part of the unique identification shall be formed by the name of the target class concatenated with "Id".

When used in specifications, the label <<names>> can be omitted to reduce clutter and to improve readability of class diagrams.

G.3.2.9.1 Sample

This shows that all instances of MscFunction are uniquely identifiable within the ManagedElement namespace. Note the use of the label <<names>> in specifications is optional.

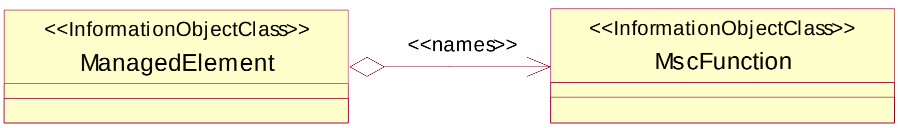


Figure G.9: <<names>> Notation

G.3.3 Visibility

It specifies the accessibility of the operation and attribute. There are three types of visibility, i.e., private, public and IRPAgent Internal.

Table G.2: Private Visibility (notation '-')

Operation	NA
Attribute	It indicates that the attribute is not accessible by other entities, e.g., the IRPManager, other entities not holding the subject attribute

Table G.3: Public Visibility (notation '+')(default)

Operation	It indicates that the operation is visible across the itf-N, e.g., the IRPManager can invoke the operation across the itf-N interface.
Attribute	it indicates that the attribute is accessible across the itf-N, i.e., the IRPManager can invoke an operation to read the attribute and to write to this attribute if the attribute is so qualified. The read or write operation must be directly invoked against the entity holding the subject attribute or against the CM IRP Agent.

Table G.4: IRPAgent Internal Visibility (notation '%')

Operation	It indicates that the operation is not visible across the itf-N, i.e., the IRPManager cannot invoke the operation. However, other entities can invoke the operation. (Note: no Release 5 operations are of this kind.)
Attribute	It indicates that the attribute is not directly accessible across the itf-N, i.e., the IRPManager cannot read/write this attribute. However, other entities can read/write this attribute.

G.3.3.1 Samples

This sample shows four attributes whose visibility are private, public (default notation), public and IRPAgent Internal. It is recommended that within a Class symbol, the use of default notation or not for public visibility should be consistent, i.e., all "publicly visible" attributes shall be shown with the "+" sign or without the "+" sign (default notation).

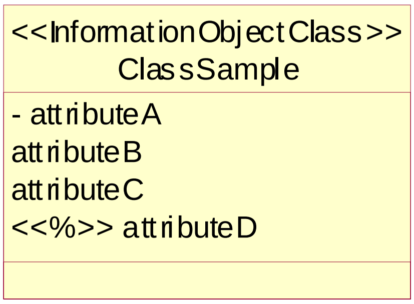


Figure G.10: Visibility of attributes

This sample shows three operations. Two of these operations are accessible by the IRPManager via the itf-N. It is recommended that within a Class symbol, the use of default notation or not for public visibility should be consistent, i.e., all "publicly visible" operation shall be shown with the "+" sign or without the "+" sign (default notation).

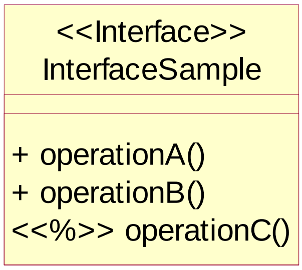


Figure G.11: Visibility of operations

This sample shows one notification whose visibility is public using the non-default public visibility notation. These notifications are accessible by the IRPManager via the itf-N.

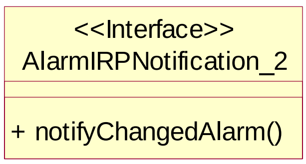


Figure G.12: Visibility of notification

Annex H (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Dec 1999	S_06	SP-99578	--	--	Approved at TSG SA #6 and placed under Change Control	-	3.0.0
Mar 2000	S_07	SP-000015	001	--	resolving remaining R99 inconsistency between 32.101 & 32.102	3.0.0	3.1.0
Mar 2000	S_07	SP-000015	002	--	Correction of IRP-related terminology	3.0.0	3.1.0
Mar 2000	--	--	--	--	Cosmetic	3.1.0	3.1.1
Jun 2000	S_08	SP-000227	003	--	Clarification of compliance conditions	3.1.1	3.2.0
Jun 2000	S_08	SP-000228	004	--	Update ITU-T TMN related reference material	3.1.1	3.2.0
Jun 2000	S_08	SP-000229	005	--	Definition of the Mandatory/Optional/Conditional qualifiers used in the IRPs	3.1.1	3.2.0
Jun 2000	S_08	SP-000230	006	--	Correction of erroneous editing and usage of undefined term	3.1.1	3.2.0
Mar 2001	S_11	SP-010026	007	--	Add UMTS TMN conformance	3.2.0	4.0.0
Jun 2001	S_12	SP-010232	008	--	Correction of ITU-T TMN concerns	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	009	--	Alignment with 3GPP drafting rules regarding headings	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	010	--	Update of TM architectural aspects	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	011	--	General clarifications and enhancements	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	012	--	Alignment with 3GPP drafting rules regarding verbal forms for the expression of provisions	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	013	--	Update and clarify compliance condition for a UMTS entity	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	014	--	Delete OSA definition	4.0.0	4.1.0
Jun 2001	S_12	SP-010232	015	--	Enhancements of the IRP Concept	4.0.0	4.1.0
Sep 2001	S_13	SP-010466	016	--	Update and alignment of compliance conditions for UMTS Management Physical architectures	4.1.0	4.2.0
Sep 2001	S_13	SP-010522	017	--	Specify the Rule for IDL file names	4.1.0	4.2.0
Mar 2002	S_15	SP-020037	018	--	Add the rule on how all SA5 Solution Set specifications indicate a reference to a particular SA5 Information Service specification.	4.2.0	5.0.0
Mar 2002	S_15	SP-020037	019	--	Inclusion of the IMS in the 3G Telecom Management Architecture (32.102)	4.2.0	5.0.0
Sep 2002	S_17	SP-020450	020	--	Correction of diagrams describing entities of the mobile system to be managed	5.0.0	5.1.0
Sep 2002	S_17	SP-020450	021	--	IS Template Changes to support new UML Repertoire/Methodology	5.0.0	5.1.0
Sep 2002	S_17	SP-020450	022	--	Addition of 3GPP UML Repertoire for IRP: IS	5.0.0	5.1.0
Sep 2002	S_17	SP-020479	023	--	Add optional parameters in CORBA Solution Set IDLs	5.0.0	5.1.0

History

Document history		
V5.0.0	March 2002	Publication
V5.1.0	September 2002	Publication