TR 101 665 V1.1.1 (1999-04)

Technical Report

Intelligent Network (IN);
Service capability modelling for IN-CS4

Reference

DTR/NA-060503 (ff000ics.PDF)

Keywords

CS3, IN

ETSI

Postal address

F-06921 Sophia Antipolis Cedex - FRANCE

Office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - 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

Internet

secretariat@etsi.fr
Individual copies of this ETSI deliverable
can be downloaded from
http://www.etsi.org
If you find errors in the present document, send your
comment to: editor@etsi.fr

Copyright Notification

No part may be reproduced except as authorized by written permission.
The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 1999.
All rights reserved.

Contents

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 SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available **free of charge** from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

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 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 Report (TR) has been produced by ETSI Technical Committee Network Aspects (NA).

Introduction

In IN CS-1 and CS-2, the IN Global Functional Plane (GFP) has been defined to help identify a comprehensive set of Functional Entity Actions and information flows on the Distributed Functional Plane (DFP). Another aspect of the IN CS-1 and CS-2 GFP is that it has provided a tool to model (benchmark) services and service capabilities.

However, IN service (capability) modelling using SIBs has shown some shortcomings. Therefore, this work item provides a first informative study on object oriented service modelling. Object oriented methodologies and modelling techniques are promising for IN service modelling. This work item shows an example of a service (video-on-demand) being modelled with techniques like ODP, UML and TINA lookalike techniques.

Service modelling techniques and methodologies investigated in this work item intend to be used for IN CS-4. However, this does not have to prevent us from applying them to IN CS-3 service features and network capabilities.

1 Scope

The present document is a first study on service capability modelling for IN CS-4.

Clause 4 identifies the requirements for IN service modelling. Clause 5 lists some shortcomings of using IN CS-2 SIBs for IN service modelling. Clause 6 explains the rationale for applying object oriented techniques, followed by the introduction of some of these techniques in clause 7. Finally, clause 8 shows as an example how the object oriented approach can be applied to Video-on-Demand.

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.
- A non-specific reference to an ETS shall also be taken to refer to later versions published as an EN with the same number.

[1] EURESCOM Project P607, "Top-down approach applied to multimedia services".

[2] ITU-T Recommendation X.902 | ISO/IEC 10746: "Information technology - Open Distributed Processing - Reference Model; Part 1: Overview, Part 2: Foundations, Part 3: Architecture, Part 4: Architechtural semantics".

[3] ITU-T Recommendation Q.1223 (1997): "Global functional plane for Intelligent Network Capability Set 2".

[4] ITU-T Recommendation Z.100 (1993): "CCITT Specification and description language (SDL)".

[5] ITU-T Recommendation X.680-X.691: "Abstract Syntax Notation One (ASN.1)".

[6] ISO/IEC DIS 10165-4 | ITU-T Recommendation X.722 (1992): "Information technology - Open Systems Interconnection - Structure of Management Information: Guidelines for the definition of managed objects".

[7] ISO/IEC 10165-7 | ITU-T Recommendation X.725 (1995): "Information technology - Open Systems Interconnection - Structure of management information: General Relationship Model".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply.

methodology: systematic working procedure consisting of a set of clearly defined steps and their relation. In each step of the methodology one or more modelling techniques can be used to carry out the activities described by that step

modelling technique: set of rules and tools that allows for the specification of a service/system in an unambiguous way. Modelling techniques can be applied at various levels of abstraction

reference point: interaction points defined in an architecture to allow access to the system

conformance point: reference point at which behaviour may be observed for the purposes of conformance testing

stakeholder: legal unit of any kind (company or person) which owns and/or uses a portion of the total system. A stakeholder can fulfil one or more roles in a community

contract: agreement governing part of the collective behaviour of a set of stakeholders

community: configuration of stakeholders formed to meet an objective. The objective is expressed as a contract which specifies how the objective can be met

domain: set of objects where each object is related by a characterizing relationship to a controlling object. E.g. an administrative domain is defined by the resources (switches, cables, …) owned by a stakeholder

inter-domain reference point: set of interfaces which define the relationships between administrative domains

federation: community of <x> domains

role: behaviour of a stakeholder in relation to another stakeholder in a community (expressed in terms of a contract)

TINA system: all the administrative domains that can inter-operate via interactions specified at one or more of the TINA inter-domain reference points

session: temporary relationship of a group of resources assigned to fulfil a task over a period of time

service component: entity encapsulating data and functionality in the computational viewpoint. It can be mapped to one Computational Object (CO) or a set of COs

3.2 Abbreviations

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

AccCOs	Accounting Computational Objects
API	ApplicationProgramming Interface
ASN.1	Abstract Syntax Notation No.1
BCSM	Basic Call State Model
B-ISDN	Broadband ISDN
CM	Connection Management
CNPs & ANPs	Core and Access NPs
ConS	Connectivity Service
CORBA	Common Object Request Broker Architecture
CPE	Customer Premises Equipment
CS n	Capability Set n (n = 1, 2,...)
CSLN	Client-Server Layer Network
CSM	Communications Session Manager
DC	Data Communication
DCA	Data Communication Application
DFP	Distributed Functional Plane
DPE	Distributed Processing Environment
FDT	Formal Description Techniques
FSM	Finite State Machine
GDMO	Guidelines for the Definition of Managed Objects
GFP	Global Functional Plane
GRM	General Relationship Model
IA	Initial Agent
IDL	Interface Description Language
IN	Intelligent Networks
ISDN	Integrated Services Digital Network
LC	Local Communication
LCA	Local Communication Application
LNFed	Layer Network Federation
MPEG	Motion Picture Expert Group
namedUA	Named User Agent
NPs	Network Providers
ODL	Object Definition Language
ODP	Open Distributed Processing
OLEX	Originating Local EXchange

OMG	Object Management Group
OMT	Object Modelling Technique
PA	Provider Agent
PSTN	Public Switched Telephone Network
QoS	Quality of Service
SC	Service Control
SCF	Service Control Function
SDL	Service Description Language
SF	Service Factory
SIB	Service Independent Building block
SRF	Specialized Resource Function
SS7	Signalling System No.7
SSF	Service Switching Function
SSM	Service Session Manager
STB	Set-Top Box
SubCOs	Subscription Computational Objects
TCon	Terminal Connection
TCSM	Terminal Communications Session Manager
TINA	Telecommunications Information Networking Architecture
TLEX	Terminating Local EXchange
UA	User Agent
UAP	User Application
UML	Unified Modelling Language
USM	User Session Manager
VoD	Video on Demand

4 Requirements for IN CS-4 Service Modelling

Requirements that have been identified as important for modelling techniques and methodologies used for IN service modelling are:

- **stepwise refinement** from service (features) to protocol level should be supported;
- support for various mechanisms of **transparency.** These include:
  - *technology* transparency:
    - the modelling of IN services should not be dependent on the technology used, e.g. the type of network, operation system or programming language. An important type of transparency in the context of IN CS-4 service modelling is network technology transparency. The objective of IN CS-4 is the integration of a number of network technologies, including connection oriented type of networks (e.g. PSTN, PLMN) and connectionless type of networks (e.g., Internet, data networks). Some of the services targeted in IN CS-4 will probably be specific to one network technology, others will require interworking between different network technologies, or their behaviour will depend on the technology the service is invoked from. One could think of a service with user interaction which can be invoked by B-ISDN as well as mobile network users, or of a multi-media conference service over different network technologies. IN could be an integrated architecture for the services on multiple network technologies. It should be investigated to what extent network technology transparency can be achieved;
  - *access* transparency: masking differences in data representation and invocation mechanisms (i.e., providing multiple mappings of the information contents of IN protocol exchanges to programming APIs);
  - *failure* transparency: masking, from an object, the failure and possible recovery of objects (including itself);
  - *migration* transparency: masking, from an object, the ability of a system to change the location of interfaces to that object;
  - *relocation* transparency: masking relocation of an object interface from other interfaces bound to it;

    - *replication* transparency: masking use of a group of mutually behaviorally compatible objects to support an interface;

- **service extensibility**: it should be possible to support additions to the IN service functionality by extending the existing service models and existing service components;

- **service scaleability**: it should be possible to accommodate and allow to scale the service capabilities in terms of the number of users, the number of nodes, the number of administrative domains, etc;

- **service version handling**: service modelling techniques should facilitate the concurrent use of multiple versions of service models and service components;

- **service reusability**: it should be possible to re-use the models of an IN service (capability) during the specification of another IN service, rather than starting the modelling from scratch even in case of overlapping functionality. The same applies to the software components that implement the specification;

- **operation, administration and maintenance**: the methodologies and modelling techniques should support flexibility when it comes to functionality and data model changes during operation, administration and maintenance. For example, functionality changes that occur during maintenance should be easily reflected in the model of the IN service;

- **reduction of service conflict**: the methodologies and modelling techniques should reduce the risks of service interactions;

- **flexible service modification/customization:**

    - the network providers, service providers, customers and users should to be able to:

        - differentiate services to meet specific market needs (network and service providers);

        - customize the services to some extent to their personal needs (users). In the short term it will be limited to data customization and selection from a list of predefined features. In the longer term service packaging might be provided as well;

- **support for a multi-stakeholder environment**. the IN CS-4 service capability modelling should be able to take into account the respective roles of the different stakeholders involved in the service provisioning (retailer, operator, content provider, etc.).

5 Shortcomings of IN CS-2 SIB approach

This clause describes the major shortcomings that have been identified for the IN CS-2 SIB approach (see ITU–T Recommendation Q.1223 [3]):

- the Service Independent Building blocks (SIBs) at Global Functional Plane (GFP) is hard to refine to a Distributed Functional Plane (DFP) level;

- the refinement of the Service Plane to the GFP SIBs contains only one intermediate step, namely that of High Level SIBs. As a result, the refinement of the Service Plane to the GFP is not very precise. In order to achieve a more stepwise refinement, the concept of High Level SIB is not sufficient, but a more iterative refinement is required;

- modelling the GFP by means of CS-2 SIBs does not provide a complete service model, since the service data is subordinate to the functions performed by an IN service. Data needs more explicit attention, e.g., to enable better management of service data;

- the SIB approach applied in IN CS1 and CS2 GFP modelling differs from modelling techniques often applied in the IT world. However, alignment with products originating from the IT world (e.g., billing and customer care) becomes increasingly important;

- the specification techniques used in the Global Functional Plane and the Distributed Functional Plane do not allow any mechanisms of consistency checking between specifications and thus limit service compositions and reusability. More generally, there is no global specification methodology advocated by IN.

6 Rationale for using object orientation in IN CS-4 service modelling

Object oriented modelling is based on the principle of 'objects'. Objects are components that are self-contained, i.e. their properties can be described independent from the outside world. The definition of an object comprises attributes and methods. Attributes describe the data in the object; methods the operations that can be imposed on that data. Objects can invoke methods in other objects. Other very important concepts of object oriented modelling include inheritance and encapsulation. A detailed description of object oriented principle can be found in "Object-Oriented Modelling and Design".

Given the wide scope and the complexity of services envisioned for IN CS-4 (resulting from the integration of mobile, broadband and Internet services) the advantages of using the object oriented paradigm within the standardization of IN CS-4 are threefold:

- object orientation methods seem suitable for specification of IN services throughout the whole design. However, it should be investigated whether IN service modelling really is an application area where OO methods can be successfully applied as well;
- object oriented methods are already widely applied in software engineering;
- an object oriented Distributed Processing Environment (DPE) can be used as a uniform computational model providing transparent distribution for the network intelligence (service logic, service creation and management). A migration scenario could be the interworking between a CORBA-based DPE (for the network intelligence) and the existing IN equipment (SSPs) through a gateway (SS7-IDL) interface as illustrated in figure 1:

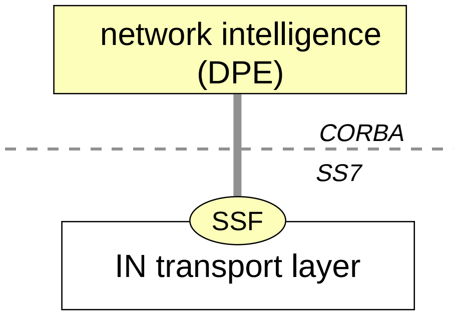

Figure 1: A possible evolution scenario for IN

The advantages of using an object-oriented DPE (e.g. a CORBA-based platform) for the network intelligence include: enhancement of service reuse and genericity (for composition and decomposition), ease of development, deployment and integration, dynamic binding and reconfiguration of service components.

CORBA (Common Object Request Broker Architecture) (see CORBA, www.omg.org) allows for invocation of object methods in a distributed environment. This means that CORBA will hide the location of the object that is addressed. The interfaces between objects specified with CORBA IDL provide technology transparency, because the interface will hide how objects have been implemented. It is however worth noting that, currently, one of the major drawbacks for use of such environments in the context of IN is the real-time requirements!

The definition of interactions between objects across a logical interface is usually referred to as an API. Figure 2 illustrates the use of APIs within an IN context.

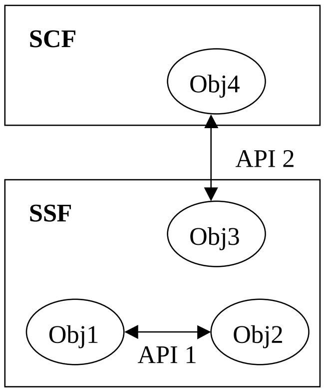

Figure 2: An example of the use of APIs in an IN context

In Figure 2, Object 1 and Object 2 reside in the SSF. Object 1 could e.g. be the Call Model object and Object 2 the Trigger Status Table. When the SCF arms a particular Event Detection Point (edp), Object 1 would invoke a method to Object 2 changing the status of a particular trigger. The API call could therefore be 'arm edp2'.

When Object 3 in the SCF invokes a method on Object 4 (API2 call) residing in the SSF, the method crosses an external interface and as a result it **can** be mapped onto a protocol operation. The properties of the API call, i.e. its contents, will need to populate the physical protocol operation. Therefore a one-to-one mapping exists between the contents of the API call and the attributes of e.g. an INAP operation. For example, imagine Object 3 represents a 'service access' object and Object 4 an IN service logic. Upon receipt of information from the user, Object 3 might invoke a method in Object 4 like 'receipt of user dialled digits'. This method could be realized as the INAP operation 'AnalysedInformation'.

7 Methodologies and modelling techniques

Subclauses 7.1 and 7.2 introduce ODP and UML as general methodologies/modelling techniques for service modelling. Subclause 7.3 introduces TINA as a possible approach for service modelling based on ODP and UML principles.

7.1 Open Distributed Processing (ODP)

7.1.1 Introduction to ODP

The reference model of ODP is a generic distributed object-oriented methodology that is suitable for both traditional telecommunications applications (such as IN) and information processing applications. It is based on two powerful trends in software technology:

- object-oriented specification techniques provided at different abstraction levels allowing a high degree of stepwise refinement and consistency checking;

- object-based distributed processing environments (e.g. CORBA-based platforms) allowing the provision of distributed services, and most importantly, enabling a distribution transparency and interworking within distributed systems.

The reference model of ODP prescribes that multiple descriptions, with different abstraction levels, should be provided for any service under study or development. Five abstraction levels – called *viewpoints* in the ODP terminology – have been defined within ODP and are considered to encompass the different areas of concerns that need to be covered in the service development process. Specifications have to be provided in each of these viewpoints using a corresponding viewpoint model or an adequate viewpoint language. The five ODP viewpoints are: *enterprise*, *information*, *computational*, *engineering* and *technology*.

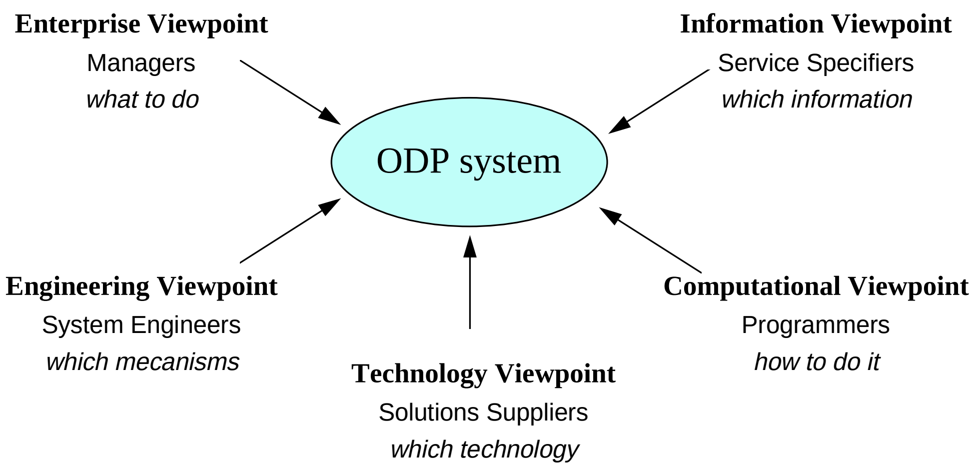

Figure 3: Different abstraction levels (viewpoints) in ODP

Enterprise Viewpoint

An enterprise specification of an IN service describes the service from the perspective of the organizations and people that will use or operate that service (see TINA-C, "Overall Concepts and Principles of TINA"). The concepts used for enterprise specifications include:

- it's the requirements that state the desired capabilities of the service at a strategic level (why it is being considered);
- the stake-holders (and other participating institutions);
- the roles (who will be involved and in what role);
- the legal environment (obligations of the stakeholders);
- the set of rules and regulations that govern the exploitation of the service (the rules that should be obeyed when the service will operate).

Usually, an enterprise specification is written in a textual format or using a high-level graphical notation.

Information Viewpoint

An information specification provides a highly abstract model of real world's entities and their relationships along with the constraints that govern their behaviour. An information specification typically covers:

- the specification of the information structure of the service (objects);
- the specification of the dependencies between the information objects (associations);
- the specification of the operations and constraints that govern the possible dynamic evolution of the service considered as collection of objects.

The information viewpoint could be developed using e.g. UML (see subclause 7.2), especially using the capabilities for data modelling (static model).

Guidelines for mapping Enterprise viewpoint to Information viewpoint

Mapping rules can be established to translate the enterprise specification into information terms. It should be noted that these are only guidelines (see ITU-T Recommendation X.902 | ISO/IEC 10746 [2]).

Computational Viewpoint

A computational specification represents an abstract implementation of the service under consideration in terms of interacting objects, whereby location, access, distribution and failure are transparent. Put simply, at this abstraction level, a service is represented as a dynamic configuration of interaction objects.

An important feature of computational specification is the ability to capture the real-time and probabilistic aspects. Binding objects, for instance, describe the behaviour of communication which complies with certain QoS constraints. These non-functional requirements are particularly relevant when it comes to the support of multimedia interactions in a distributed environment.

The specification of the computational viewpoint would typically contain the well-known functional entities such as SCF, SSF, SRF, etc. as computational objects, perhaps further decomposed depending on the distribution needs. The interactions between the computational objects would be in terms of 'operations' (i.e. INAP operations) and 'flows' (i.e. voice, video and data).

The computational viewpoint could be developed using e.g. UML for specification of the computational objects and e.g. IDL for the specification of interfaces between objects.

Mapping from Information viewpoint to Computational viewpoint

Though ODP does not recommend any mapping solution between information and computational object types, the following simple guideline is a starting point: there is a one-to-one mapping. A computational object type specification is obtained by taking an information object type specification and adding the description of the operations for which it can have a server role. However, considerations regarding the potential for distribution will modify this mapping.

Engineering Viewpoint

An engineering specification determines how computational, distribution-free descriptions can be realized in terms of generic system components and communication protocols (such as SS7). It therefore focuses on how interaction between objects is achieved and which resources are needed to do so. From an engineering viewpoint, an ODP system is considered as a collection of computer systems. The details of the underlying communication networks, operating systems and hardware are hidden by a uniform and basic distributed environment.

In ODP terminology the SSF-FSMs and the BCSMs could be considered as *stubs* or *protocol objects*. The signatures of the operations would correspond to the ASN.1 definitions of the INAP operations.

The Engineering specification would be very similar to the protocol specification.

It could be developed using SDL (see ITU-T Recommendation Z.100 [4] and ITU-T Recommendation X.680-X.691 [5]). SDL is used to describe the objects and their behaviour, and ASN.1 to describe the signatures of the operations visible at external interfaces (hence corresponding to protocol messages). The reason for using ASN.1 and not IDL is that a large number of protocols with which IN needs to interwork are already specified using ASN.1.

The consistency between the Computational and the Engineering specifications can be maintained by using tool support. For example, the computational objects, specified using UML/OMT, can be 'pasted' in the SDL specification as SDL objects (types). The tool will maintain such links between UML/OMT and SDL objects, and consistency checks can be done between the two models. Also, SDL process diagrams can be automatically generated from UML/OMT statechart descriptions.

Technology Viewpoint

The technology viewpoint describes the implementation of the system in terms of the hardware and software components (see "Common ETSI Approach to Standards Development, Report from the Board Ad-Hoc Group on New Working Methods"). It may need to consider cost and availability constraints. Selections influence the performance and quality of service of the system. Because it is directly concerned with implementation, the technology viewpoint is outside the scope of standardization.

7.1.2 Evaluation of ODP

The benefits of ODP for IN service modelling include:

- *stepwise refinement of specifications*: shows some possible scenarios of specification refinement in ODP. Note that specification refinement and transformations can be made in an iterative manner;

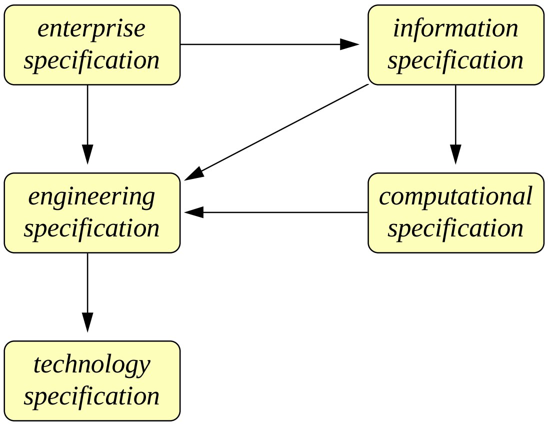

Figure 4: Possible scenarios of specification refinement in ODP

- *transparency mechanisms*: ODP provides various mechanisms of transparency at different levels of abstraction. These transparency mechanisms include: access transparency, location transparency, failure transparency, migration transparency and transaction transparency;
- *service portability* and *Reusability of service components*: these two requirements can be easily supported in an object-oriented DPE, such as a CORBA-based platform;
- *feature Interaction*: ODP strongly advocates use of Formal Description Techniques (FDT) in different viewpoints. Use of FDTs enhances the description of service behaviour and considerably reduces the risks of service interactions;
- *naming and Trading Functions*: these are parts of the functions defined by ODP. The trader function provides the means for advertising service offers and the means to discover service offers through service requests.

ODP does not provide any method on how to develop each viewpoint or which language should be used to describe it. Therefore ODP should be applied in conjunction with the appropriate modelling techniques for each of the viewpoints.

7.2 Unified Modelling Language

7.2.1 Introduction to UML

The Unified Modeling Language (UML) (see UML, www.omg.org), a de facto standard from OMG, is a further development and merging of Booch, OMT, and OOSE. The result is a single, widely recognized modelling language for users of these and other methods. UML provides a standard modelling language, not a standard process.

UML is a modelling language that incorporates the object-oriented community's consensus on core modelling concepts. It allows deviations to be expressed in terms of its extension mechanisms. The developers of UML had the following objectives in mind during its development:

- provide sufficient semantics and notation to address a wide variety of contemporary modelling issues in a direct and economical fashion;
- provide sufficient semantics to address certain expected future modelling issues, specifically related to component technology, distributed computing, frameworks, and executability;
- provide extensibility mechanisms so individual projects can extend the metamodel for their application at low cost. Users should not be forced to adjust the UML metamodel itself;
- provide extensibility mechanisms so that future modelling approaches could be grown on top of the UML;
- provide sufficient semantics to facilitate model interchange among a variety of kinds of tools;
- provide sufficient semantics to specify the interface to repositories for the sharing and storage of model artifacts.

7.2.2 Evaluation of UML

UML is particularly used in the information and computational viewpoints of ODP. Although UML could also be applied to other ODP viewpoints, there is currently limited application to these. Since a large number of protocols with which IN needs to interwork are already specified in ASN.1 and SDL, it might be considered for backwards compatibility purposes to use ASN.1 and SDL rather than UML in the engineering viewpoint.

7.3 TINA

7.3.1 Introduction to TINA

TINA provides an approach based on ODP and UML and OMG CORBA concepts. This means e.g. that TINA applies the five view points defined ODP. In this section the technology viewpoint is not considered since it is not relevant with respect to IN service modelling.

7.3.2 TINA business modelling

TINA aims to provide a common architecture that can be applied to different service scenarios. It ensures that TINA entities (business roles) can interact in a multi-stakeholder/multi-vendor environment. The TINA reference architecture identifies a number of generic reference points. These reference points specify the set of interfaces (interactions) that can take place between various TINA-based entities (businesses). The Reference Point concept describes a service architecture in terms of a 'business model' and 'computational object model'. Note that TINA uses the term "business modelling", whereas ODP uses the term "enterprise viewpoint".

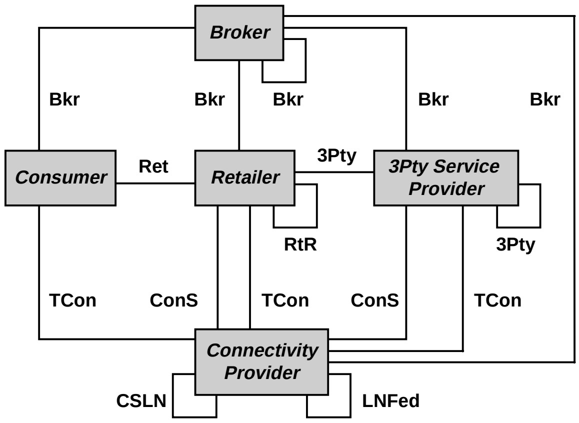

Figure 5: General TINA Business Model

The TINA business model (see TINA-C, "Business Model and Reference Points"), shown in Figure 5, defines the service related inter-domain reference points. It identifies the various business roles that can be performed by TINA stakeholders and the relationships (interactions) between them. The TINA business model is equivalent to what is called "enterprise model" in ODP. It is noted that while a stakeholder in any business (role) can participate in one or more of the businesses defined, regulatory restrictions may bar a stakeholder from operating a monopolized service portfolio. The following is a description of these businesses:

- **Consumer**: a consumer is the end user of the service(s) provided in a TINA system. Consumers may range from private individuals to large companies. Consequently, CPE (Customer Premises Equipment) will range from PCs or STBs to large corporate networks;

- **Broker**: a broker provides stakeholders with interface reference information on how to access services provided by other stakeholders in the TINA system. The user-provider relationship with a broker enables users to retrieve, store, or update information the broker manages;

- **Retailer:** the retailer provides (sells/resells) various services to consumers. A retailer business can be anything from a large corporation to a small niche company. Consumers can have access to more than one retailer offering the same services. Retailers will generally interact with other stakeholders, such as third party service providers and connectivity providers;

- **Third Party Service Provider**: a third party service provider provides services to retailers or other third party service providers. This stakeholder resembles a wholesaler or supplier having agreements with retailers and other third party service providers. The difference between a third party service provider and a retailer is that this stakeholder can not have a contractual relationship with consumers. Thus, a retailer acts as a mediator between consumers and third party service providers;

- **Connectivity Provider**: the connectivity provider is the manager (owner) of a transport network controlled by the TINA Connection Management (CM). Connectivity providers offer an interface to retailers and third-party service providers, which enables them to request connections between arbitrary terminating points in the network.

The business model also defines the relationships (inter-domain reference points) that can appear between administrative domains (owned by different stakeholder). Subclause 8.2 shows these relationships for the Video on Demand example service. The inter-domain reference points specify the generic interactions, with each reference point having an Access and Usage Part. The Access Part contains the interfaces that are needed to establish a contractual relationship over an administrative domain boundary. A common set of capabilities is defined for the access part of all inter-domain reference points. These include, security (authentication/authorization), billing/accounting, subscription, service instantiation, QoS (Quality of Service), and negotiation of Usage Part interfaces. The Usage Part contains the prescribed interfaces used for the generic services offered across a reference point. The usage part also includes interfaces used for management purposes.

7.3.3 TINA information modelling

For information modelling TINA uses TINA-C, "Information Modelling Concepts", a notation which is a form of GDMO (Guidelines for the Definition of Managed Objects) with GRM (General Relationship Model). See also TINA–C, "Overall Concepts and Principles of TINA", ISO/IEC DIS 10165-4 | ITU-T Recommendation X.722 [6] and ISO/IEC 10165-7 | ITU-T Recommendation X.725 [7]. GDMO and GRM were chosen by TINA because these standards are widely used in the telecommunications management community and because a large number of GDMO specifications were available for re-use. TINA has used only those aspects of GDMO and GRM that are suitable for (TINA) information modelling. The TINA specific notation is called quasi GDMO-GRM. OMT graphical notation (see "Object-Oriented Modelling and Design") has been adopted for the diagrammatic representation of information specification.

7.3.4 TINA computational modelling

TINA applies the TINA ODL (Object Definition Language) for computational modelling. ODL enhances OMG IDL (Interface Definition Language) (see "The OMG Object Model"). An example of an enhancement is the specification of objects that are made up of one or more interfaces. OMG IDL does not have the notion of objects with multiple interfaces.

7.3.5 Evaluation of TINA

TINA is an application of ODP on telecommunications services. TINA is a promising approach for service modelling and is likely to meet most of the service modelling requirements. Although TINA concepts seem to be successfully applicable to IN service modelling, there is currently limited application. Moreover, support of some IN specific requirements (real-time constraints) and impact on legacy systems needs further study.

8 Example: modelling Video-on-Demand with TINA concepts

Clause 7 indicated that the TINA approach is based on ODP and UML concepts. This clause shows how the TINA concepts are applied to a service example, i.e. Video on Demand. This material has been derived from Eurescom project P607 [1].

8.1 Introduction to Video-on-Demand

The VoD Service, as described in [1], provides for the transfer of digitally compressed and encoded video/audio information across a telecommunications network. The source is termed a Video Server and the destination will usually be a Set-Top Box (STB). Commands from the user are relayed across a network to the Video Server via a control channel. The reverse video stream is displayed after decoding and converting on a monitor - usually a TV set.

The service allows a user to access a library of films on a remote digital storage system (at the Video Server). The coding scheme employed is Motion Picture Expert group (MPEG) encoded video. The Video Server acts as a repository of such MPEG encoded videos and, on request, sends the selected video to the STB on an 'on demand' broadband channel.

The initial request is transmitted by the user via his STB, which sends messages in response to user prompts. The STB offers the user basically the same functionality as a VCR, facilities such as 'fast forward/reverse', 'stop' and 'pause', with the additional capability to decode the video and signalling information and present it to the user.

The user's first point of interaction is with a Level-1 Gateway. This can be viewed as a service supplied by a Broker or Service Gateway Provider. When connected to this Gateway, the user can select the VoD service from a menu containing a number of different service types. The user can then select a particular VoD Service Provider (VoD SP) from a list of available VoD SPs. Based on this selection, the user will then be connected to a Level-2 Gateway and to a Video Server. The service requires a control channel from the user's STB to the different gateways and a broadband channel from the Video Server to the STB.

NOTE: The Service Provider may not be the Content Provider.

The integrated IN/B-ISDN service architecture for the VoD service is represented in Figure 6.

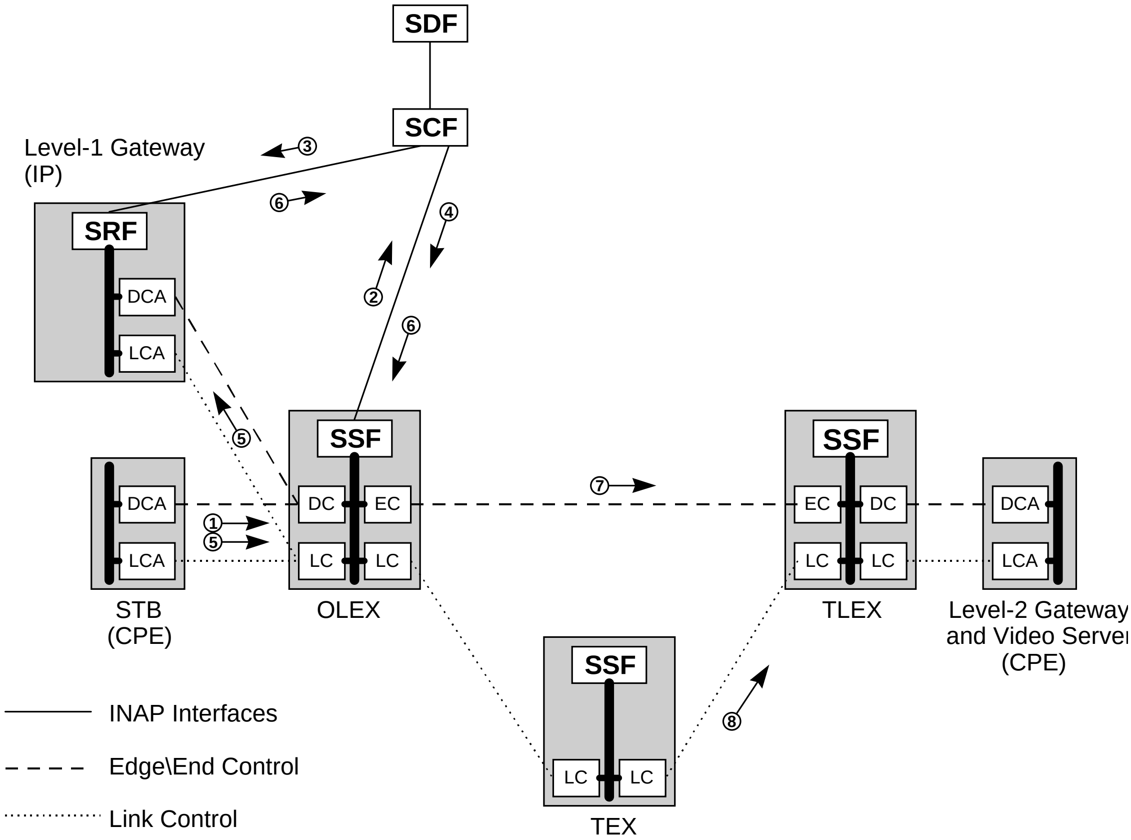

Figure 6: VoD Service Scenario in the Integrated IN/B-ISDN Service Architecture

Following service provisioning steps have been identified:

Step 1: the DC in the OLEX receives a call request from the DCA in the STB. IN call handling is then invoked via the SSF;

Step 2: the Originating Local Exchange (OLEX) establishes an association with the IN SCF and requests access to the appropriate Level-1 Gateway via its SSF. User authentication information may also be requested and processed at this point;

Step 3: the request is processed by the SCF and the SDF, where location and routing information is stored. The SCF then locates the Level-1 Gateway (IP/SRF);

Step 4: the SCF instructs the OLEX via the SSF to set-up call (DC-DCA) and bearer (LC-LCA) relationships with the Level-1 Gateway. The bearer will contain the control channel required to transport user selections to the Gateway. Some application software for the STB may be down–loaded from the Gateway at this point if necessary. A broadband connection is established if the selection includes video clips;

Step 5: the user is now connected to the Level-1 Gateway via the STB and may select a Video Server;

Step 6: the address information from the selection is conveyed to the SCF by the SRF in the Level-1 Gateway. The SCF then forwards this information to the OLEX via the SSF;

Step 7: at this stage the OLEX DC and LC can release the connection to the Level-1 Gateway. The OLEX EC will then transfer the call from the user to the Service Provider (Level-2 Gateway) via the TLEX. A bearer is also established via the LC to provide a user control channel. The user now has access to the Level-2 Gateway and can Select the video for viewing. Once the selection is made, a broadband channel is established from the Video Server to the STB.

The control channel allows user features such as 'pause' and 'rewind'. When the STB - VoD Server session is finished, the OLEX is instructed to release the call to the TLEX.

8.2 Video-on-Demand business model

Based on the business roles described in subclause 7.3.2, the business model proposed for the VoD Service is shown in Figure 7.

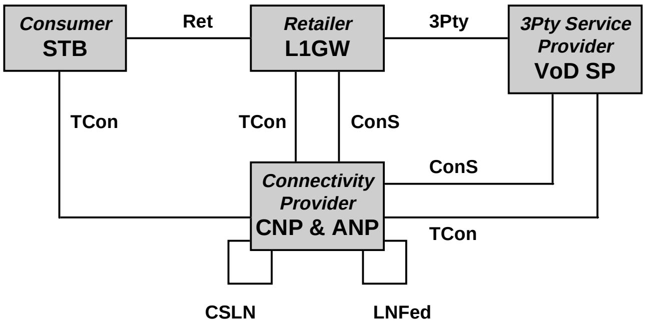

Figure 7: Video-on-Demand business model

The following businesses (stakeholders) were identified:

- **Consumer**: end-user (STB);
- **Retailer**: Level-1 Gateway (L1GW);
- **3Pty Service Provider**: Level-2 Gateway and Video Server (VoD SP);
- **Connectivity Provider**: federation of Network Providers (NPs), i.e. Core and Access NPs (CNPs & ANPs).

The following relationships (inter-domain reference points) have been identified between administrative domains in the VoD service:

- **Ret** (Retailer): defines the reference point (relationship) between a stakeholder in the consumer business and a stakeholder in the retailer business, i.e. [STB - L1GW];
- **3Pty** (Third Party): defines the reference point between a stakeholder in a third party service provider business and stakeholders in the broker businesses, i.e. [L1GW - VoD SP];
- **ConS** (Connectivity Service): defines the reference point between stakeholders that provide network connectivity services and stakeholders that use these services on behalf of their customers, i.e. [L1GW - NPs] or [VoD SP – NPs];
- **TCon** (Terminal Connection): defines the reference point between a stakeholder in the connectivity provider business and a stakeholder which can terminate a stream flow in its domain, i.e. [STB - NPs] and [NPs – VoD SP];
- **LNFed** (Layer Network Federation): defines the reference point between stakeholders in the connectivity provider business that enables the mutual provisioning (federation) of transport facilities;
- **CSLN** (Client-Server Layer Network): defines the reference point between stakeholders in connectivity provider business that have a client-server relationship with each other, i.e. between a CNP operating an ATM VP switched network and an ANP operating an ATM VC switched network.

8.3 Video-on-Demand computational model

The VoD computational model is based on the TINA service architecture (see TINA-C, "Service Architecture"). This subclause shows as an example of computational modelling four service provisioning steps based on the VoD service architecture:

Step 1: User Access, Authentication, and Authorization;

Step 2: VoD Service Selection and Instantiation;

Step 3: VoD Service Provider Selection;

Step 4: VoD Content Selection.

NOTE: In this VoD service scenario the subscriber has no contractual relationship with a VoD SP. Instead, the subscriber negotiates the terms of the service with the SP indirectly via the L1GW. Some sub-contractual relationships should exist between the various stakeholders to support accounting capabilities (billing and charging) i.e. relationships between the L1GW, VoD SP and NPs. For example, a customer who subscribes to several SPs receives one bill from the L1GW instead of separate bills from each SP. Also, user requests for information on overall charges will be co-ordinated by the L1GW.

The following subclauses provide a more detailed description of how the service provisioning steps can be realized by the proposed VoD service architecture.

8.3.1 Introduction of components in the VoD computational model

This subclause shows the overall computational model for Video-on-Demand and briefly introduces the different components used in the Video-on-Demand computational model.

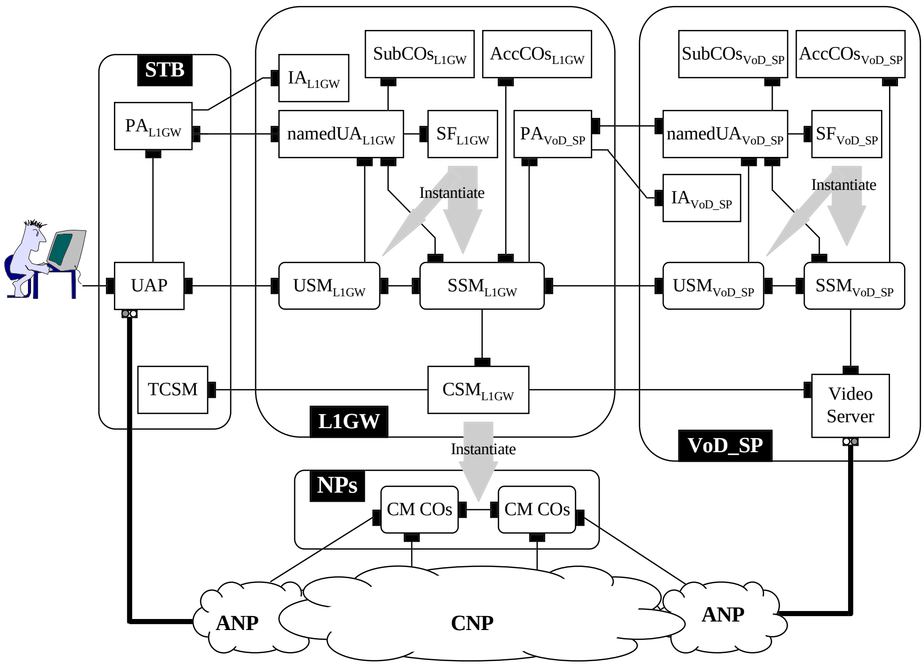

Figure 8: Overall Video-on-Demand computational model

UAP **User Application:** a session component is used to model applications in the user domain. The UAP supports the user in subscribing to a provider. Once subscribed, the UAP supplies authentication information required by the PA to set-up an Access Session with a UA. Within the Access Session the UAP is used to convey user requests to create (instantiate) a new Service Session. It also enables users to customize their service profiles, e.g. set service specific preferences, and schedule service to be activated at a particular time. The UAP also supports generic session control capabilities, such as suspending or resuming the user's participation in a Service Session.

PA **Provider Agent:** a service independent session component. It facilitates the set-up of a trusted relationship between the user and provider, by interacting with an Initial Agent (IA) and being referred to a named User Agent (namedUA).

IA **Initial Agent:** a user and service independent session component. It is the initial access point to a provider's domain. An IA reference is returned to the PA when it wishes to contact and establish an access session with a provider. Through interactions with the PA, the IA supports the capabilities to authenticate the user and set-up a trusted relationship between the user and the provider.

UA **User Agent:** the UA is a service-independent SC that is used to represent a user in the provider's domain. Within the Access Session it acts as a single contact point to control and manage (create/suspend/resume) a new User Service Session and a Service Session by interacting with the SF to create a User Session Manager (USM) and Service Session Manager (SSM). The UA also provides access to the user/provider contract information;

namedUA (Named User Agent): a specialization of the UA for a user who is an end-user or subscriber of the provider (in terms of the generic user/provider paradigm).

SF **Service Factory**: a service-specific SC that is capable of creating (instantiating) the Service Session objects (typically USM and SSM SCs) of a particular service type.

SSM **Service Session Manager:** an SSM supports service capabilities that are shared among users in a Service Session. User dependencies in services are encapsulated in the USM. The SSM supports operations to track and control the resources in a Service Session. It also supports management capabilities associated with the Service Session, such as accounting.

USM **User Service Session Manager**: the USM represents and holds the context of the user in a Service Session. It contains the information and service capabilities which are local to the user.

SubCOs **Subscription Computational Objects**: functionality including:

- providing access to subscription information pertaining to a user (subscriber), e.g. personal and service specific user profile;
- managing and negotiating subscription contracts and user (subscriber) profiles;
- handling templates that represent capabilities provided by the service provider;
- managing all subscription-related information, i.e. user (subscriber) profiles.

AccCOs **Accounting Computational Objects**: a set of computational objects that are used to manage the billing and charging aspects of a service session.

TCSM **Terminal Communications Session Manager**: a service-independent SC, which manages and controls stream bindings (communications streams) within the user's domain. The TCSM provides interfaces to allow the CSM to set-up, modify, and remove stream bindings in the user's domain. The user's domain is often referred to as a 'terminal', irrespective of whether it is a single terminal point or a gateway to other communications or computing resources.

CSM **Communications Session Manager**: a service-independent SC, which manages end-to-end stream bindings. The CSM provides interfaces to allow USM/SSMs to set-up, modify, and remove stream bindings.

8.3.2 User access, authentication and authorization

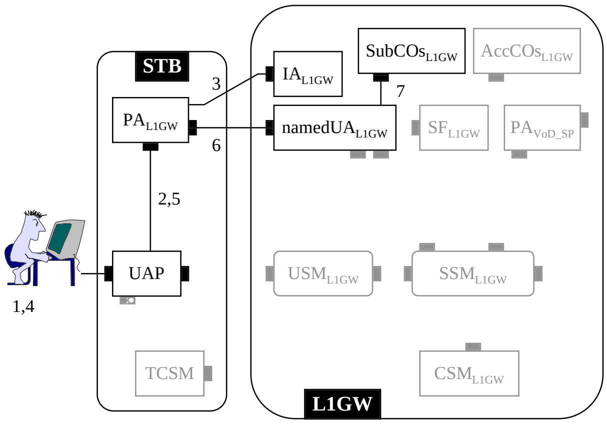

Figure 9: User Access, Authentication and Authorization

1) **STB - UAP**: the user activates the UAP (via the STB) to initiate an access session - when started the UAP prompts the user to enter his user ID and Password;

2) **UAP - PA$_{L1GW}$**: the UAP obtains (see note) an interface reference to the PA$_{L1GW}$ and then invokes an operation on this interface to request access to the domain of the Broker/Retailer, i.e. L1GW;

NOTE: How this is achieved has not been specified by TINA.

3) **PA$_{L1GW}$ - IA$_{L1GW}$**: it is assumed that the user is known to the Broker, i.e. there has been contact between them previously. During the initial contact, i.e. subscription, the PA$_{L1GW}$ obtained an interface reference to the IA$_{L1GW}$ of the L1GW and a namedUA$_{L1GW}$ was assigned to the user. The PA$_{L1GW}$ now requests from the IA$_{L1GW}$ the interface reference to the named UA$_{L1GW}$. Included in the request is a reference to its own interface which the named UA$_{L1GW}$ will now use during the access session. The IA$_{L1GW}$ authenticates the user and returns the interface request. A trustworthy relationship has now been established between the user and the Broker;

4-6) **STB - UAP - PA$_{L1GW}$ - namedUA$_{L1GW}$**: the user requests the list of services available to him. The UAP forwards this request to the namedUA$_{L1GW}$ via the PA$_{L1GW}$ using the interface reference that IA$_{L1GW}$ provided;

7) **namedUA$_{L1GW}$ - SubCOs$_{L1GW}$**: the namedUA$_{L1GW}$ then requests the list from the SubCOs$_{L1GW}$ (which stored the user's profile). This is then relayed back along the operations path to the user (STB).

At this point the user can edit his user profile (change ID, password, negotiate new services, etc.).

8.3.3 VoD service selection and instantiation

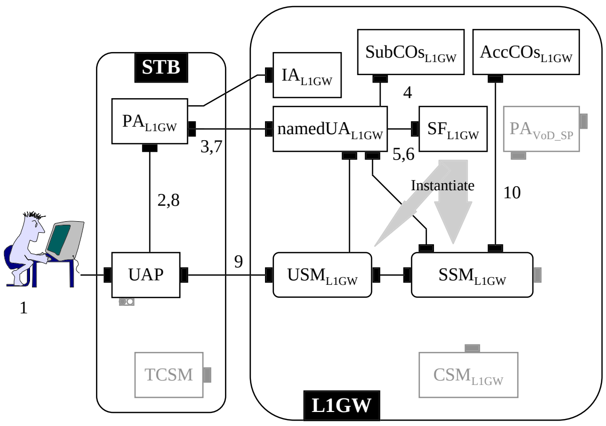

Figure 10: VoD Service Session and Instantiation

1-3) **STB - UAP - PA_{L1GW} - $namedUA_{L1GW}$**: from the list of services the user selects the VoD Service. The UAP forwards the service request to the $namedUA_{L1GW}$ via the PA_{L1GW};

4) **$namedUA_{L1GW}$ - $SubCOs_{L1GW}$**: the $namedUA_{L1GW}$ then invokes the appropriate operations in the $SubCOs_{L1GW}$ to get a reference to an instance of a VoD Service based on the service specific user profile information, i.e. a short list of preferred VoD SPs;

5) **$namedUA_{L1GW}$ - SF_{L1GW}**: the $namedUA_{L1GW}$ then sends the request to the SF_{L1GW} to instantiate the SSM_{L1GW} and USM_{L1GW}. Once instantiated, the service specific information will be used to configure the service;

6-8) **SF_{L1GW} - $namedUA_{L1GW}$ - PA_{L1GW} - UAP**: The SF_{L1GW} returns the SSM_{L1GW} and USM_{L1Gw} interface references to the $namedUA_{L1GW}$. These are then forwarded to the service session related UAP via the PA_{L1GW};

9) an interface between the USM_{L1GW} (and SSM_{L1GW}) and the UAP is established and the list of VoD SPs is sent to the UAP. The user can then browse through this list;

10) **SSM_{L1GW} - $AccCOs_{VoD\_SP}$**: an interface between the SSM_{L1GW} and the $AccCOs_{L1GW}$ is established at this point to manage the billing of the Service Session.

This is the last step of the Access Session. Now that the Service Session SCs have been instantiated, the Usage Part of the Service begins. At this point the user can edit his service specific profile, i.e. display settings, types of VoD SP.

8.3.4 VoD service provider selection

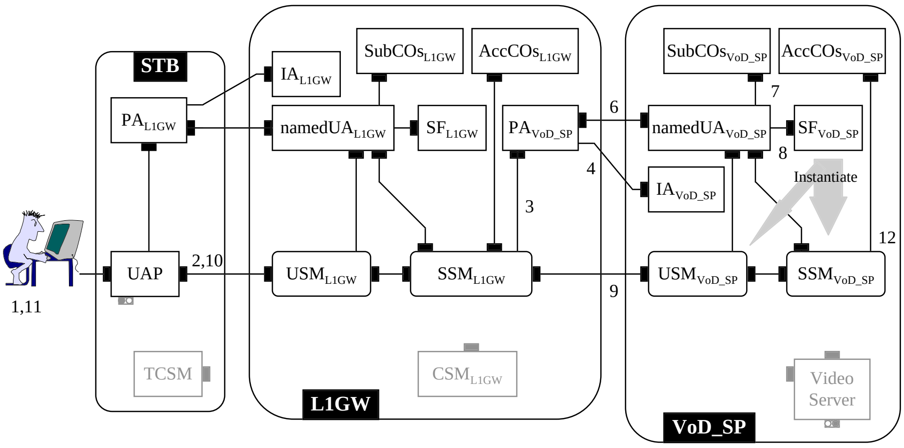

Figure 11: VoD Service Provider Selection

1-3) **STB - UAP - $USM(SSM)_{L1GW}$ - PA_{VoD_SP}**: the user selects a VoD SP from the available list. The request is forwarded to the $USM(SSM)_{L1GW}$ via the UAP. The $USM(SSM)_{L1GW}$ obtains the interface reference to the PA_{VoD_SP} and then invokes an operation on this interface to log-in to the domain of the VoD SP. Additional authentication information may be required at this stage;

4) **PA_{VoD_SP} - IA_{VoD_SP}**: it is assumed that the L1GW has a subscription contract with the VoD SP, i.e. PA_{VoD_SP} has obtained an interface reference to the IA_{VoD_SP} of the VoD SP and as a result a named UA_{VoD_SP} was assigned to the L1GW. The PA_{VoD_SP} now requests from the IA_{VoD_SP} the interface reference to the named UA_{VoD_SP}. Included in the request is a reference to its own interface which the named UA_{VoD_SP} will now use during the access session. The IA_{VoD_SP} authenticates the user and returns the interface request. A trusted relationship has now been established between the L1GW and the VoD_SP;

5-6) **SSM_{L1GW} - PA_{VoD_SP} - $namedUA_{VoD\_SP}$**: the SSM_{VoD_SP} then sends a service request to the named UA_{VoD_SP} via the PA_{VoD_SP} to accept the registration of a (new) user. The SSM_{VoD_SP} also forwards service session information, i.e. the user's personal and service specific profile (see subclauses 2.3.1.1 and 2.3.1.2);

7) **$namedUA_{VoD\_SP}$ - $SubCOs_{VoD\_SP}$**: the $namedUA_{VoD\_SP}$ then invokes the appropriate operations in the $SubCOs_{VoD\_SP}$ to obtain a reference to an instance of a VoD Service session based on the service specific user profile information;

8) **namedUA$_{VoD_SP}$ - SF$_{VoD_SP}$**: the named UA$_{VoD_SP}$ then sends the request to the SF$_{VoD_SP}$ to instantiate and configure the USM$_{VoD_SP}$ and SSM$_{VoD_SP}$ based on this reference;

9-11) **USM(SSM)$_{VoD\_SP}$ - USM(SSM)$_{L1GW}$ - UAP - STB**: an interface between the USM(SSM)$_{VoD\_SP}$ and USM(SSM)$_{L1GW}$ is established and a list of available content (relating to the service preferences) is sent to the user. The user can now browse through this list;

12) **SSM$_{VoD_SP}$ - AccCOs$_{VoD_SP}$**: an interface between the SSM$_{VoD_SP}$ and the AccCOs$_{VoD_SP}$ is set-up at this point to co-ordinate the billing of the Service Session.

8.3.5 VoD content selection

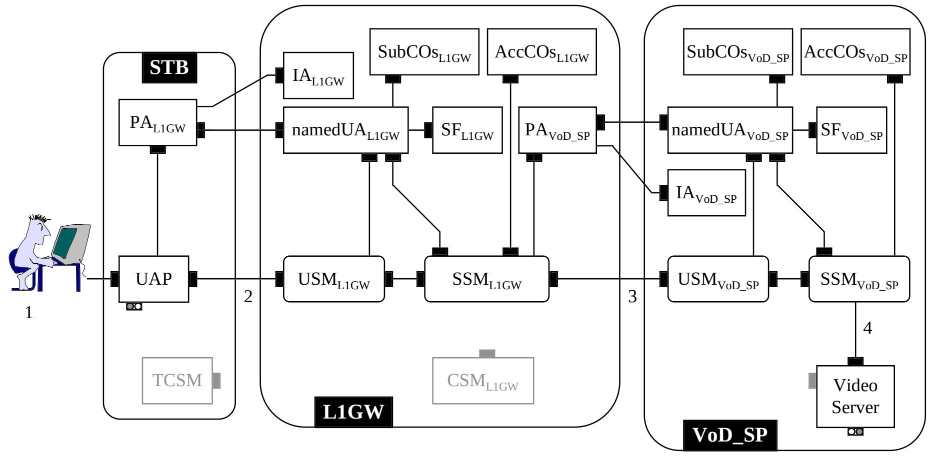

Figure 12: VoD Content Selection

1-4) **STB - UAP - USM(SSM)$_{L1GW}$ - USM(SSM)$_{VoD\_SP}$ - Video Server**: the user selects a video from the VoD SP. The request is sent by the UAP to the USM(SSM)$_{VoD\_SP}$ via the USM(SSM)$_{L1GW}$. The SSM$_{VoD_SP}$ obtains the interface reference to the Video Server which is relayed back along the operations path to the SSM$_{L1GW}$.

Prior to video selection, service specific control and management operations may be available to the user, e.g. negotiation of QoS and charging. As this might involve interactions between the VoD SP and NPs, the L1GW may act on behalf of the user in these negotiations. In addition, the USM(SSM)$_{VoD\_SP}$ can invoke an appropriate interface to the SubCOs$_{VoD_SP}$ to check if the user is allowed to take part in this session, i.e. restricted access for named user.

9 Conclusions

The present document contains a first study on service modelling for IN CS-4. The document described the requirements important for IN service modelling. Then the shortcomings of the SIB approach used for service modelling in IN CS-2 were illustrated, followed by the rationale for using object orientation in IN CS-4 service modelling. The present document introduced some methodologies/modelling techniques that are increasingly applied for service modelling: ODP and UML. Also the TINA approach, which applies a combination of ODP and UML was briefly introduced. Finally, the Video-on-Demand service wastaken an example to show how TINA concepts can be used for service modelling.

The service modelling techniques and methodologies investigated in this work item intend to be used for IN CS-4. However, this does not have to prevent us from applying them to IN CS-3 service features and network capabilities.

The evaluation of ODP showed that ODP does not provide any method on how to develop each viewpoint or which language should be used to describe it. Therefore ODP should be applied in conjunction with the appropriate modelling techniques for each of the viewpoints.

UML is particularly used in the information and computational viewpoints of ODP. The computational viewpoint could be developed using UML for specification of the computational objects and IDL for the specification of interfaces between objects. Although UML could also be applied to other ODP viewpoints, there is currently limited application to these. Since a large number of protocols with which IN needs to interwork are already specified in ASN.1 and SDL, it might be considered for backwards compatibility purposes to use ASN.1 and SDL rather than UML in the engineering viewpoint.

TINA is an application of ODP on telecommunications services. TINA is a promising approach for service modelling and is likely to meet most of the service modelling requirements. Although TINA concepts seem to be successfully applicable to IN service modelling, there is currently limited application. Moreover, support of some IN specific requirements (real-time constraints) and impact on legacy systems needs further study.

The following work is therefore seen as important for progressing service modelling for IN CS-4:

- apply the approach described in the present document on more IN services;
- identify elementary service capabilities that can be re-used in various services. The use of the techniques presented in the present document might impact the IN conceptual model. The challenge of identifying service capabilities is to define them at the appropriate level: a too high level of abstraction makes it difficult to use them as building blocks for service modelling; a too low level of abstraction will result in service capabilities that do not meet the network transparency requirement.

Bibliography

The following material, though not specifically referenced in the body of the present document (or not publicly available), gives supporting information.

- EURESCOM Project P506: "Harmonization/Integration of B-ISDN and IN".
- ITU-T Recommendation Q.1290 (1998): "Glossary of terms used in the definition of intelligent networks".
- TINA-C, "Service Architecture", Version 5.0, June 1997.
- TINA-C, "Overall Concepts and Principles of TINA", Version 1.0, February 1995.
- TINA-C, "Information Modelling Concepts", Version 2.0, April 1995.
- TINA-C, "Business Model and Reference Points", Version 4.0, May 1997.
- TINA-C, "Computational Modelling Concepts", Version 3.2, May 1996.
- "Common ETSI Approach to Standards Development", Report from the Board Ad-Hoc Group on New Working Methods.
- "Object-Oriented Modelling and Design", Rumbaugh et al.
- OMG Document Number 91.9.1, Draft, "The OMG Object Model", September 1991.
- "Telecommunications Architecture for an Evolving Environment (TAEE)", ITU-T, Baseline Document, version 2.

History

Document history		
V1.1.1	April 1999	Publication

ISBN 2-7437-2989-9
Dépôt légal : Avril 1999