Developing Mobile Distributed Intelligent Network Services Using RM-ODP

By
Gaurav S. Rampal

Thesis submitted to the
School of graduate Studies and Research
in partial fulfillment of
the requirements for the degree of
Master of Computer Science

under the auspices of the Ottawa-Carleton Institute for Computer Science

Univerisity of Ottawa.
Ottawa, Ontario, Canada
January 1998.

Acknowledgments

I wish to make a brief mention about the people without whom I would have been unable to come this far. Deep gratitude to:

Professor Luigi Logrippo, my supervisor through this work. I am grateful for support, academic and financial; for freedom to choose a subject of my preference; and for patience shown in times when my work was not upto the necessary standards. Without his help this work could never have been done.

The National Sciences Engineering Research Council of Canada (NCERC) and Motorola Canada for funding of this work.

John Vissers, who introduced me to Intelligent Networks and subsequently gave insight into telecommunication systems by patiently answering numerous of my questions.

George Young, for conceptual aspects behind mobile communications and for related documentation.

Raj Pandya, for introducing future evolution of mobile communication systems - IMT 2000 - and the problems to be solved before successful delivery.

The LOTOS group at the University of Ottawa - Kazi Farooqui for introducing me to ODP; Daniel Amyot for the numerous technical tips; to Jacques for the providing system support; and to Jalel Kamoun and Hichem Ben Fredj for their presence in the lab. The tough and uncompromising questions raised during the group meetings was a wonderful, though at times uncomfortable, way to cut deeper into technical material.

Mayank Tandon who first gave insight into the intricacies and wonders of technology back in 1989. Without his feedback I would probably have dropped out.

Ashish, for his patience in dealing with my strange hours, especially when I was trying to manage work and the writing of the thesis.

My parents, for offering me choices and supporting me even when they were not the most sensible ones.

My grandfather Daulat Ram Kundi and the rest of the Kundi family - Kamlesh, Virbala, Pran, and Aruna for their generosity and support over the past four years.

Ven. Anzan Hoshin Sensei, for the opportunity to practice, for showing me my confusion, my struggle, and the possibilities within. To the Godo, the Ino and the Shuso for all the details in the practice environment, far too numerous to mention here.

Abstract

The Intelligent Network (IN) is a conceptual model for a service development technology to create telecommunication services. In its current form, IN is limited to service creation in isolated networks and cannot support co-operative service development between two or more networks. Rapid development in networking paradigms and standards has led to an urgent need of finding solutions to the problem of interworking heterogeneous networks. Differing abstraction levels make meaningful exchange of information difficult, and IN has not been able to meet this requirement.

The Reference Model for Open Distributed Processing (RM-ODP) is a distributed object based architecture which provides a high level framework for distributed systems. The emphasis is to develop a set of re-usable functional abstractions that can be recombined in various configurations to develop required applications.

This work uses RM-ODP framework to supplement deficiencies evident in IN. Two specific aspects are examined and developed. The first is service portability through service profile modeling. A model for service development in a mobile environment, and related concepts of service profile modeling and transfer are developed. The second, IN domain interworking in the ODP framework. A ODP framework for the modeling of this service profile and its migration as the user moves to different domains is proposed. Our approach allows dynamically configured interworking of domains.

Table of Contents

List of Figures

List of Acronyms

BCP	Basic Call Process
BCSM	Basic Call State Model
BTS	Base transceiver Station
CCAF	Call Control Agent Function
CCF	Call Control Function
CCSN	Common Channel Switching Network
CDNP	Customer Defined Numbering Plan
CS	Capability Set
DFP	Distributed Function Plane
DTMF	Dual Tone Multi-frequency
FE	Functional Entity
FEA	Functional Entity Action
FPLMTS	Future Public Land Mobile Telephone System
GFP	Global Functional Plane
GSL	Global Service Logic
GVNS	Global Virtual Network Service
IF	Information Flows
IMEI	International Mobile Equipment Identifier
IMT 2000	International Mobile Telecommunications 2000
IMUI	International Mobile User Identifier
IP	Intelligent Peripheral
ISCP	Interworking Service Control Point
ISPT	Internetwork Service Profile Transfer
ITU	International Telecommunications Union
ODP	Open Distributed Processing
PCS	Personal Communications Systems
PIC	Point In Call
POTS	Plain Old Telephone System
PP	Physical Plane
RBOC	Regional Bell Operating Company
RM-ODP	Reference Model for Open Distributed Processing
RN	Role Name
SCE	Service Creation Environment
SCF	Service Control Function
SDF	Service Data Function
SDP	Service Delivery Point
SDPID	Service Delivery Point Identifier
SIB	Service Independent Building Block
SMF	Service Management Function
SLP	Service Logic Program
SN	Service Node

SP	Service Plane
SPL	Service Privilege List
SRF	Service Resource Function
SS	Service Set
SS7	Signaling System Number 7
SSF	Service Switching Function
SSP	Service Switching Point
TDP	Trigger Detection Point
TMN	Telecommunication Management Network
UMTS	Universal Mobile Telecommunications System
UPT	Universal Personal Telecommunication
VPN	Virtual Private Network

Chapter 1
Introduction

1.1 Motivation and Problem Context

Computer and telephone networks have proliferated over the past 2 decades. Interworking at physical and routing levels is well established, however substantial heterogeneity exists in the application level.

Telecommunication network technology is now accessible and affordable to a majority of the general population. The average end user of today is not as technologically proficient as an average user of a decade ago. As a consequence, successful service provisioning of today must deliver to the end user simple and intuitive services and simultaneously integrate entrenched systems.

The primary motivation of this work is to devise a service development framework where distributed services can be modeled. Such modeling must be intuitive for the service designer so that mis-alignment between user requirements and delivered services is minimum.

The Intelligent Network (IN) conceptual model developed in the mid-1980's is the primary service development platform available to the traditional telephone company. IN promises the introduction of new services in a quick, cost effective and vendor independent manner. Service providers can meet customer demands by increasing breadth of services offered and reducing delivery schedules.

However, as the past few years have illustrated, IN has not been as effective as hoped. Core conceptual IN ideas were developed when two critical factors of today's scenario were absent: deregulation and a high penetration of mobile users. During the time of initial IN conceptualization it was assumed that the number of service providers would be few, and services would be accessed from fixed terminals only. In today's environment these limitations are clearly evident. With an increase in the number of providers, and users demanding services from different points in the network, co-operation and interworking becomes vital. IN is unable to support mobile users and distributed services. Removing this limitation would help in the evolution of IN to 3rd generation mobile communication systems like IMT-2000 and UMTS. Overcoming this limitation is the target of this work.

The reference model for Open Distributed Processing (RM-ODP) is a reference architecture for targeting interoperability of disparate systems. In this work, concepts borrowed from the RM-ODP standards are used to provide a common interaction model. It is demonstrated that concepts borrowed from RM-ODP, can be used to overcome the limitations of IN service development platform.

1.2 Objectives

The intent is to apply ODP distributed processing technology for the development of telephony services. Specifically stated the objectives of this work are:

1. Support for development of distributed services.

Identifying a suitable model for the development of services distributed across domains. Current IN standards do not address situations where such distribution requires interworking across technological and administrative domains.

2. Support for Mobile Services

Developing a new model that supports seamlessness. As IN fundamentally assumes service access from fixed terminals, a new model that supports seamless service delivery to mobile users across networks has to be developed.

The main guidelines that were followed towards the solution of the problem are:

1. Development of an intuitive model for service developers.

Over the past 5 years there has been an increase in a the number of independently managed network entities. This along with the introduction of mobility has introduced significant ambiguity in traditional methods for understanding service behavior. A coherent model that integrates existing approaches has to be developed.

2. Reuse existing concepts and terminology

This objective is exceedingly important. A minimum number of new terms, acronyms and terminology is to be introduced. Intellectual manageability is exceedingly important. We believe also that our solutions allow reuse of telephone legacy software.

1.3 Organization

This thesis consists of 7 chapters organized as follows:

Chapter 1: Introduction.
Introduces the motivation, context and objectives of this work.

Chapter 2: Intelligent Network Conceptual Model

Chapter 2 is an introduction to IN. Fundamental IN concepts are introduced from the evolutionary perspective. The benefits of IN are explained. Trends driving telecommunications and the impact they have on IN are illustrated. Two dominant trends are examined: co-operative corporate telecommunications and user mobility.

Chapter 3: Introduction to Reference Model for Open Distributed Processing.

Chapter 3 seeks to illustrate the scope of RM-ODP standards. The goal of ODP - developing a uniform interaction model is explained. Reuse of interworking RM-ODP traders in IN service development is fundamental to the context of this work, hence RM-ODP trading is discussed in finer detail.

Chapter 4:Service Portability in Mobile Telecommunications Systems

This chapter examines the impact that co-operation and mobile services have on the traditional perspective of service behavior. A new method of understanding and modeling service behavior is introduced. As an extension to modeling service behavior, this chapter introduces service portability and service profile modeling.

Chapter 5: Modeling Mobile Global Virtual Network Services in the RM-ODP Framework.

This chapter examines the issues faced when an attempt is made at combining two different conceptual models: IN and RM-ODP. Key differences are highlighted, and an integrated modeling methodology is introduced.

Chapter 6: The GVNS Model in the RM-ODP Framework

The GVNS example is an illustration of integrated modeling of IN and ODP. This non-trivial example was chosen to adequately illustrate the validity of the contributions of this work. The ODP enterprise and information viewpoints are developed. The ODP enterprise viewpoint highlights important details of the service level. In this information viewpoint, appropriate structures are developed along with a state model and related information flows. The ODP trading graph is used to model distributed services.

Chapter 7: Conclusion and Further work. This chapter presents conclusions and future work.

Chapter 2
Intelligent Network Conceptual Model

2.0 Introduction

The objective of *Intelligent Networks (IN)* is the introduction of new services in telecommunications network in a quick, cost-effective and vendor independent manner. This chapter is an explanation of the characteristics of IN, their benefits and limitations with respect to trends in telecommunications.

The requirement by telephone operating companies for increased responsiveness to customer demands resulted in the following: increase in service velocity, increase in breadth of service offers, and multi-vendor implementation, through evolution from existing networks. To meet these objectives IN standards are developed by the ITU. The ITU series of standards are the Q.12xx series. The first release of the standard is called *Capability Set 1 (CS1)*. Subsequent nth releases will be numbered Capability Set n. In North America IN is called AIN and refers to the BellCore AIN 0.x series of standards.

The primary characteristics of IN is its service-independent capability. This is achieved by four fundamental steps:

- *Separation of switching and control* by placing the service control outside of the switch in a computer dedicated to service control.
- *Defining service development building blocks* for the service control node.
- *Providing a independent signalling network* with standard protocols between the service control node and the switch.
- *Standardizing the call model* for the switch.

As explained below, in IN service providers can rapidly develop services on service control nodes. The service is introduced on the network by associating a specific switch to a designated service control node. Using standardized switch call models and communication protocols, the service control nodes can communicate with switches from numerous vendors, thereby achieving multi-vendor inter-operability. This approach is in sharp contrast with the traditional telephony model, where services were loaded into each switch in accordance with vendor specific designs.

The following section illustrates IN concepts from an evolutionary viewpoint. Subsequent sections illustrate IN conceptual details and terminology as discussed in the ITU Q.12xx series of standards.

Conceptually, IN can be applied to any communication network: fixed telephone, broadband, or multimedia. In this study IN refers to the telephone network perspective. Further, ITU [Q.12xx] rather than BellCore [AIN 0.x] terminology is used.

2.1 Network Evolution

2.1.1 Plain Old Telephone Service (POTS)

Prior to the mid 1980s, the service logic was hard-wired in switching systems. Typically, network operators met with switch vendors, discussed the types of services customers required, negotiated the switching features that provided the services, and finally agreed upon a generic release date for feature availability. After this, the network operator planned for the deployment of the generic services in the telephone network. The example the Figure 2.1 illustrates the POTS implementation of the 1-800 or free-phone service.

1. A customer, physically located in San Diego with telephone number 1-415-234-5679, needs a freephone number

2. This number must be accessible from 4 cities: Ottawa, New York, Montreal and Vancouver.

3. The local service provider co-ordinates with service providers in the four cities and determines an available 1-800 number.

4. Switches in all 4 cities are programmed to translate the 1-800 number to the physical number 1-415-234-5679.

5. The local service provider issues 1-800-456-2340 to the customer.

This implementation does not scale well if a large number of switches are to be programmed.

If the number of switches are N,
and the number of services are S,
then the number of modifications to be performed to introduce S services is: NxS.

Maintaining consistency between *N* switches is tedious, resulting in slow service delivery time-frames.

The demand on the switch intelligence capability increases with complexity of service logic. Hence, provisioning of uniform service coverage requires all switches to support the most complex service.

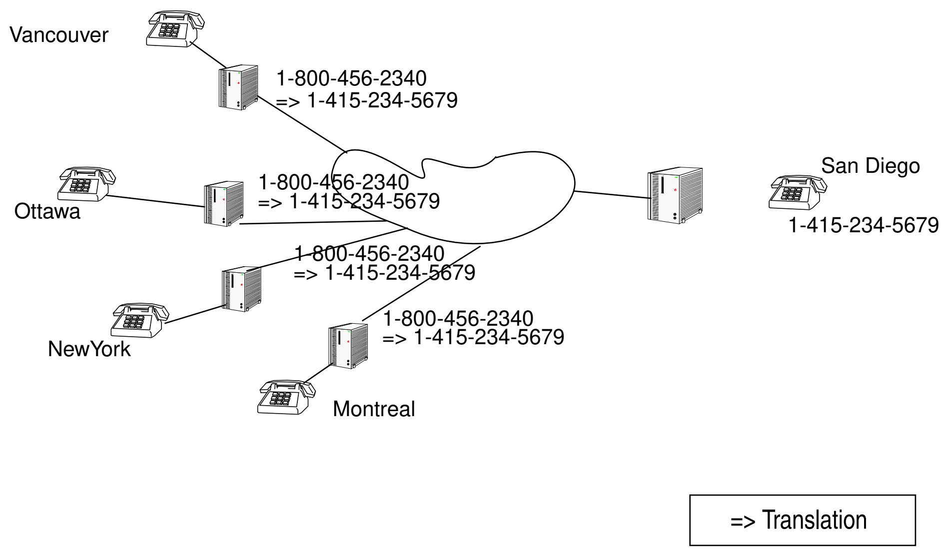


Figure 2.1 POTS Implementation of the 1-800 or Freephone Service

This process was complex for network operators with switching systems from multiple vendors. As a result, services were not offered ubiquitously across a operator's serving area. So, a customer in one end of a city, county, or state may not have had the same service offerings as a person in another area.

Also, once services were implemented, they were not easily modified to meet individual customer's requirements. Often the network operator negotiated the change with the switch vendor. As a result of this process, it took years to plan and implement services.

2.1.2 Development of IN

During the mid-1980s, the Regional Bell Operating Companies (RBOCs) began requesting development of network architectures that met the following objectives:

- rapid deployment of services in the network
- vendor independence and standard interfaces
- opportunities for non-RBOCs to offer services for increased network usage

Bell Communications Research (Bellcore) responded to this request and developed the concept of Intelligent Networks. The fundamental change was the physical separation of switching and control by placing service control in a dedicated computer. The IN based 1-800 service implementation is shown in the following Figure 2.2

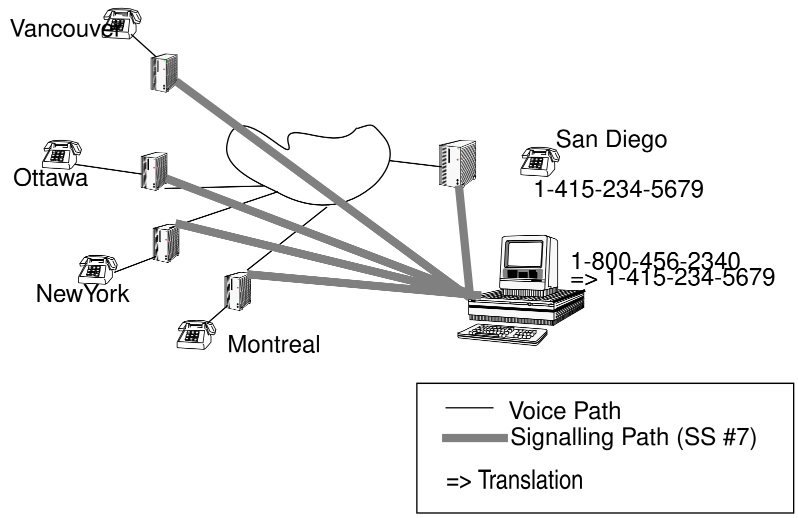


Figure 2.2 **Intelligent Network: Illustration of Basic Principle. IN Implementation of the 1-800 or Freephone Service**

1. A customer, physically located in San Diego with telephone number 1-415-234-5679, needs a freephone number
2. This number must be accessible from the 4 cities of Ottawa, NewYork, Montreal and Vancouver.
3. The local service provider co-ordinates with other service providers to determine an available 1-800 number.
4. The local service provider issues 1-800-456-2340 to the customer.
5. The local service provider co-ordinates with service providers in the four cities and switches in all 4 cities are programmed to query the service control node when a 1-800 number is dialled.
6. When a 1-800 number is dialled, the switch queries the remote service control node to provide the actual number.
7. The service control node returns the physical number to be dialled.

This implementation scales well with a large number of switches. The switches do not need to be re-programmed for every number added, but only need to know which service control node to query for number translation.

With the introduction of the IN, for the first time that service logic has become external to switching systems and located in service control nodes. These service control nodes contain databases and other intelligence and are called *service control points (SCPs)*.

2.1.3 Evolution of IN and Common Channel Signalling Network.

Evolution of IN was greatly aided by the deployment of Common Channel Signalling Network (CCSN). This ultra-reliable switching network allowed signalling between a switch and a SCP. In previous architectures, when a call was set up, a signal and talk path used the same physical path from the originating switching system to the terminating switching system. Often there were multiple switches involved in the routing of a call. This process seized the route in all of the switching systems involved. Hence, if the terminating end was busy, the entire path was set up unnecessarily. The CCSN separates the signalling from the voice path. This is called out-of-band signalling. CCSN technology frees up voice circuits between switching systems for the actual calls. The type of information transferred on the signalling network includes permission for the call setup or whether the called party is busy. IN uses Common Channel Signalling (CCSN) and Signalling System Number 7 (SS7). SS7 is the protocol that runs over the CCSN.

It is important to note that from a conceptual view any reliable out of band signalling network is acceptable for the deployment of IN. SS7 is currently used as it has very high reliability levels and is widely implemented.

2.2Intelligent Network Conceptual Model: The Four Plane Perspective

To plan and design the Intelligent Network , the Intelligent Network Conceptual Model (INCM) was defined by the ITU [Q.12xx]. The next section illustrates IN according to the Q.12xx series of Standards.

The Intelligent Network Conceptual Model (INCM) is an integrated and cohesive framework for service implementation: from high-level service description to low-level protocols between network entities.

This model involves a four plane representation as shown in figure 2.3:

- The **Service Plane (SP)** represents an exclusively service oriented view which contains no information regarding the implementation of services.
- The **Global Functional Plane (GFP)** Models an IN structured network as a single entity. This plane contains the call model and service building blocks chained together with service logic.
- The **Distributed Functional Plane (DFP)** models the distributed view of the IN structured network. Functional Entities (FEs) represent different functionalities of the network.
- The **Physical Plane (PP)** models the physical aspects of the IN network, the different types of physical entities and the protocols used to communicate.

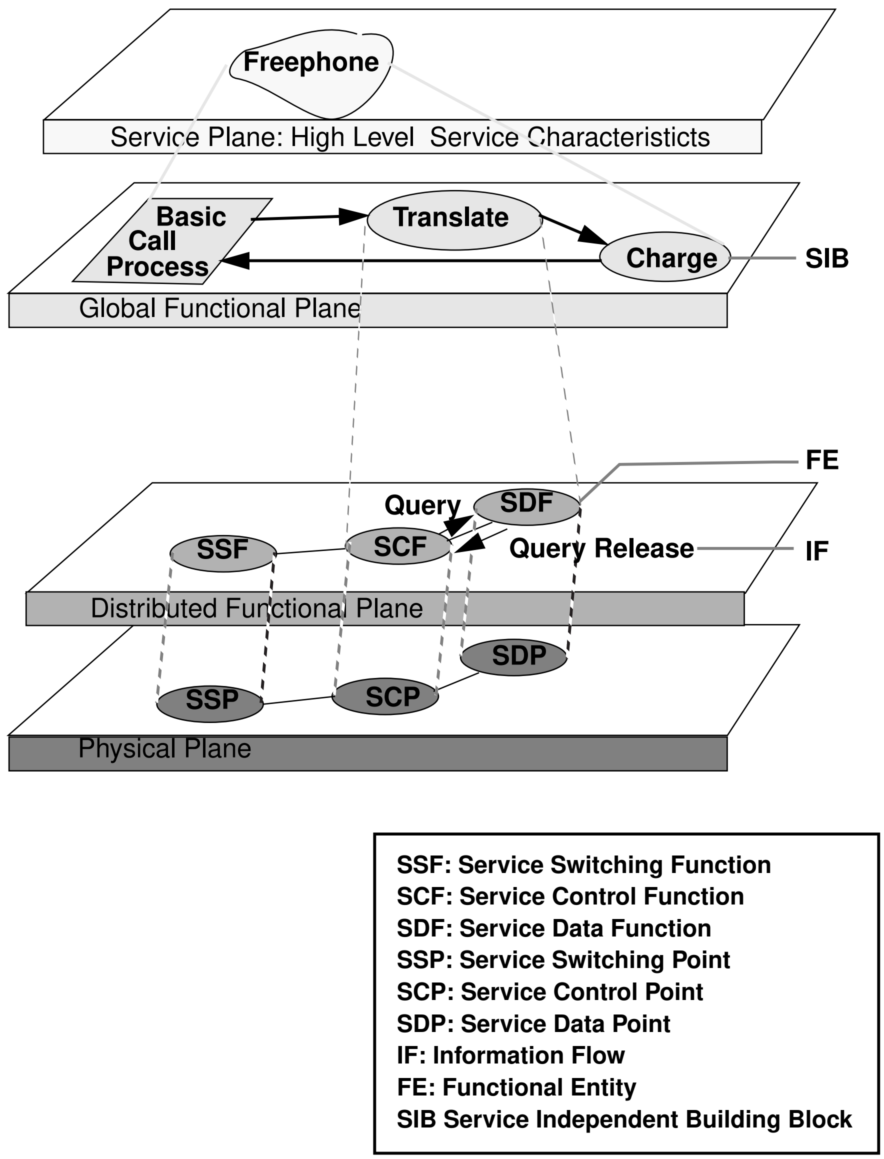


Figure 2.3 Intelligent Network Conceptual Model

2.2.1 Service Plane

The service plane is a high level description of the service. This level is appropriate for presenting service characteristics to potential customers. No implementation details are visible. It is not known where and how the service is implemented: in the terminal, in the switch or in the SCP. The example shown in Figure 2.3 is the 1-800 or freephone service.

2.2.2 Global Functional Plane

In the Global Functional Plane (GFP) the network is regarded as a single entity. The conceptual building blocks are used by the service logic for creation of services. These building blocks are called *Service Independent building Blocks (SIBs).*

A SIB

- is a standard reusable network wide capability used to create services
- is independent of functional or physical architecture.
- represents a complete activity
- has one start point and one or more end points.

A specialized SIB is the Basic Call Process (BCP) represents a process resident in a switch from which IN services are launched. The SIBs are chained together using Global Service Logic (GSL) to provide the services. By definition the SIBs including the BCP are service independent and cannot contain knowledge of subsequent SIBs.

Consider the example in the following Figure 2.4 It shows the GFP view of 1-800 service.

1. The BCP launches the 1-800 service after analyzing the address (dialled number).

2. A SIB, called the translate SIB, translates the 1-800 number (a logical number) into the actual number.

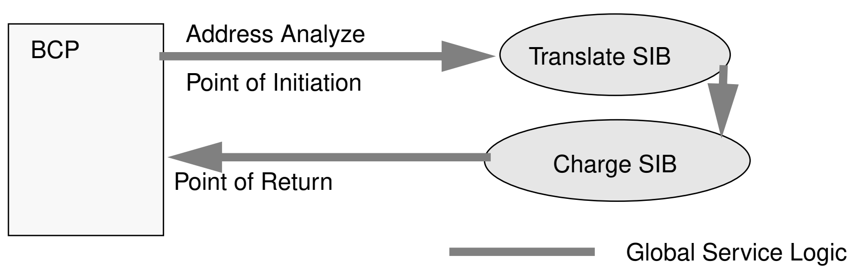


Figure 2.4 Service Development Using SIBs: An Example of 1-800 Number Service

3. The Translate SIB is chained to the Charge SIB which charges the called party for the duration of call.

In order to describe services with generic SIBS, some elements of service dependency are needed. Service dependency can be described using data parameters which enable a SIB to be tailored to perform the desired functionality.

SIBs enable significant flexibility and capability for rearranging defined network functionality. The CS1 specification defines 13 SIBs.

2.2.3 Distributed Functional Plane

The monolithic view of a SIB from the GFP can be decomposed in the Distributed Functional Plane (DFP) into a set of interacting functional entities (FEs). These capabilities can be modeled as client server relationships within the SIB. This means that part of the SIB could be realized in the SCF and part of the SIB realized in the SDF. The functional entities (FEs), the functional entity actions (FEAs) and the information flows (IFs) are detailed in the DFP. The functional entities specified on the Distributed Functional Plane (DFP), shown in Figure 2.5, are the following:

- *The Call Control Function (CCF)*: provides the means for establishing and controlling bearer services on behalf of network users. It refers to call and connection handling
- *The Service Switching Function (SSF)*: provides the means to recognize calls requiring IN service processing, and to interact with call processing and service logic on behalf of those calls.
- *The Service Control Function (SCF):* contains the IN service logic. It handles activities like analysis, translation, screening, routing.
- *The Specialized Resource Function (SRF):* provides all end user interaction with the IN network through DTMF, voice recognition, speech processing and announcements.
- *The Service Data Function (SDF):* handles access to service related and network data.
- *The Call Control Agent Function (CCAF):* provides service access to the users and represents them in call processing.
- *The Service Management Function (SMF):* provides service provisioning, deployment, and management control.

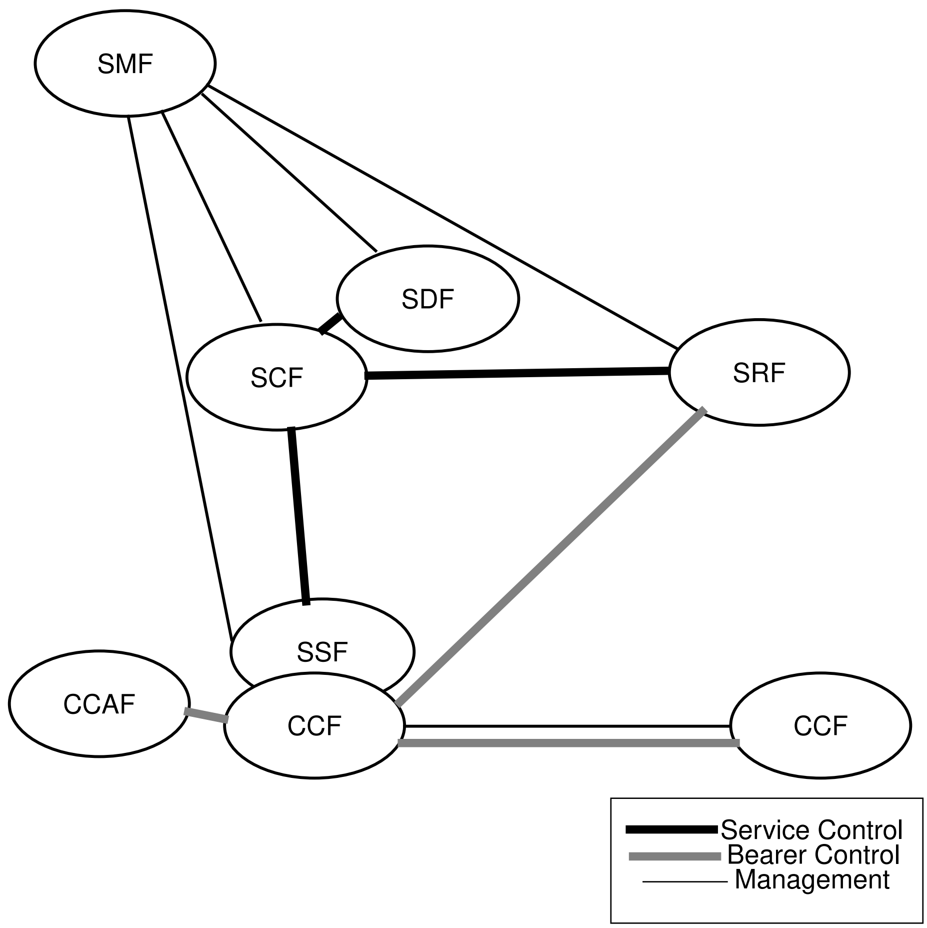


Figure 2.5 Functional Entities on the Distributed Functional Plane (DFP)

This decomposition can be understood by examining the translate SIB. One of the functions of the translate SIB is to translate a 1-800 number into the actual number. This SIB is mapped to a distributed view (Figure 2.6). A section of the SIB is implemented in the service data function (SDF) and another section in the Service Control Function (SCF). The interactions between these functional entities are called information flows. The SDF-

SCF Information Flow (IF) is Query, and the SDF-SCF IF is Query-Result.

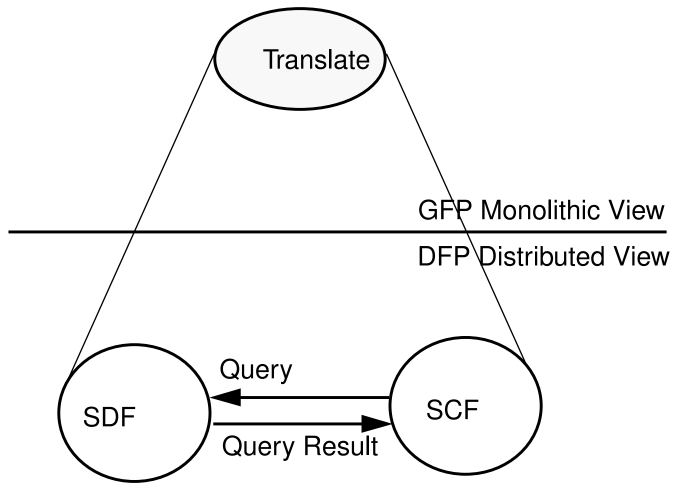


Figure 2.6 SIB Decomposition

The IN Call Model

The call model (Figure 2.7) is a generic representation of Service Switching Function (SSF) call processing activities required to establish, maintain, and clear a basic call. The call model consists of Points In Call (or PICs) and Trigger Detection Points (TDPs), and triggers. PICs represent the normal switching system activities or states that a call goes through from origination to termination. For example, the SSF is in null state or the idle state when it is actually monitoring the customer’s line. Other examples of PICs, are off-hook (or origination attempt), collecting information, analyzing information and routing.

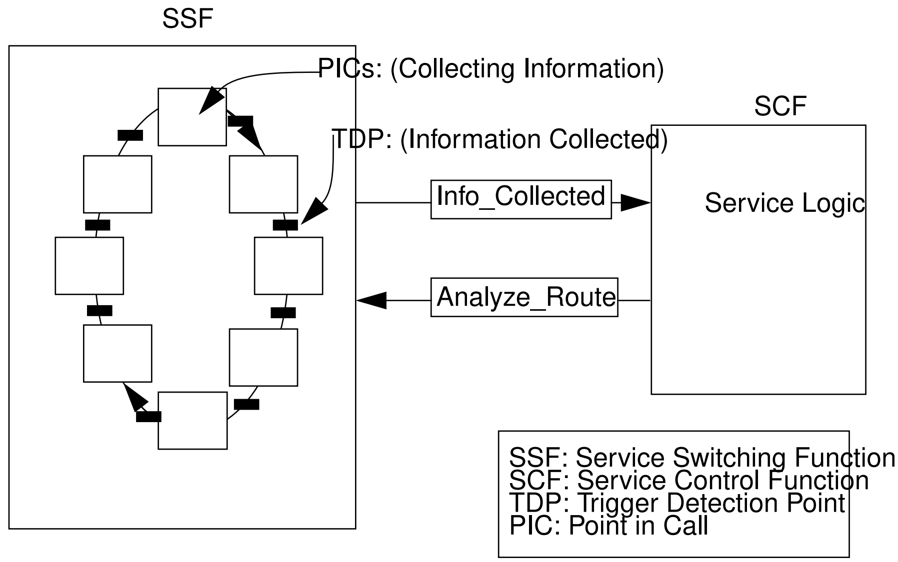


Figure 2.7 IN Call Model Conceptual View

Switching systems had similar state models before IN was developed. However, the advent of IN introduced a formal call model that all switching systems must adhere to. In this new call model is called the *Basic Call State Model (BCSM)*.

The SSF and the SCF communicate at specific points in the BCSM. For example suppose an IN call has progressed through the null state or PIC, the off-hook PIC, and is currently at the collecting information PIC. Normal call processing is suspended at the "*information collected*"detection point. Before progressing, the SSF assembles the *"information collected"* message and sends it to the SCF over the SS7 network. After SCF service logic acts on the message, the SCF sends a routing message that tells the SSF how to handle the call.

Essentially, when the SSF recognizes that a call has an associated IN trigger, it suspends the call processing while querying the SCP for call routing instructions. Once the SCF provides the instruction, the SSF continues the call model flow until completion of the call. This is basically how a call model works, and it is a very important part of IN. Figure 2.7 is illustrative of the call model concept, the details of the IN formal Call Model can be found in [Q.1219].

Each of the functional entities described above can be mapped onto a physical component of the IN structured network. This mapping can be viewed on the Physical Plane.

2.2.4 Physical Plane

The functional entities on the Distributed Functional Plane are mapped to physical entities on the Physical Plane.

A typical IN network consists of all or some of the following components:

- *Service Switching Point (SSP):* The SSP provides the SSF and CCF. It provides central switching resources, and signalling and bearer interfaces to other network components. When the SSP has subscribers directly connected to it, it also provides the call control agent functionality (CCAF).
- *Service Control Point (SCP):* The SCP contains the SCF. It is a non-switching network element for the implementation of service logic.
- *Service Management System (SMS):* The SMS provides the SMF. It interfaces with virtually all components of the network.
- *Service Data Point (SDP):* The SDP contains the SDF. Most commonly, the SDF and SCF are implemented together in the SCP.
- *Service Creation Environment (SCE):* The SCE supports the SCEF. It provides capabilities for service creation, verification and testing.
- *Intelligent Peripheral (IP):* The IP is the physical entity on which the SRF is implemented. The SRF may be implemented in the SSP.
- *Service Node (SN):* The SCF, SRF and SDF functional entities can be mapped onto a service node. The SN can control IN service and interact with users.

Two possible Physical Plane (PP) implementations of the Distributed Functional Plane (DFP) are shown in Figure 2.8. Boxes represent physical entities, hence the figure shows that functional entities can be mapped to physical entities in different ways. The physical plane is not considered in this work and will not be discussed in any further detail.

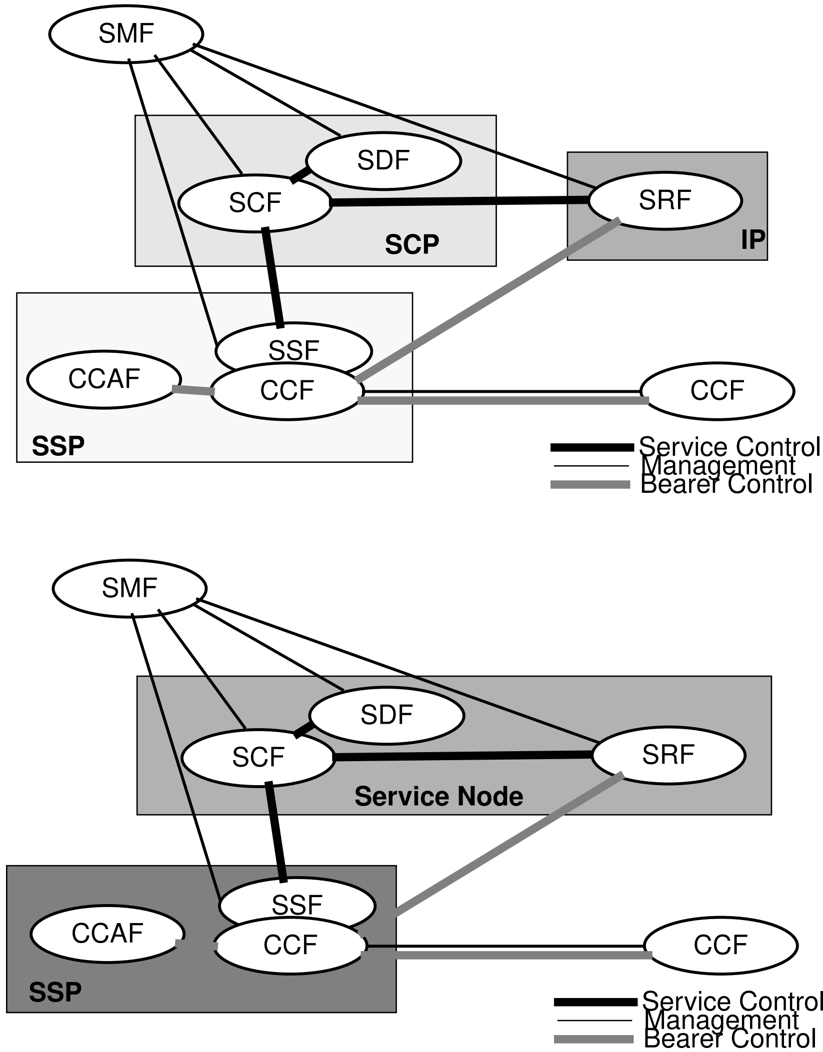


Figure 2.8 Two Possible PP realization of the DFP

2.3 Benefits of Intelligent Networks

The main benefit of intelligent networks is the ability to improve existing services and develop new sources of revenue. To meet these objectives, providers require the ability to:

Introduce new services rapidly.

IN provides the capability to provision new services or modify existing services throughout the network with physical intervention.

Provide Service Customization.

Service Providers require the ability to change the service logic rapidly and efficiently. Customers are also demanding control of their own services to meet their individual needs.

Establish Vendor Independence.

A major criterion for service providers is that the software must be developed quickly and inexpensively. To accomplish this, suppliers have to integrate commercially available software to create the applications required by service providers.

Create Open Interfaces.

Open interfaces allow service providers to introduce network elements quickly for individualized customer services. The software must interface with other vendors' products while still maintaining stringent network operations standards. Service Providers are no longer relying on one or two vendors to provide equipment and software to meet customer requirements.

The SCP contains programmable service-independent capabilities (or service logic) that are under the control of service providers. The SCP also contains service-specific data that allows service providers and their customers to customize services. With the IN, there is no such thing as one size fits all: services are customized to meet individual needs.

Network providers can offer market-focused service trials by loading service logic in an SCP and triggering capabilities in one or more switching systems. Since service logic is under the service provider's control, it is easier to create services in a cost-effective manner.

2.4 Trends Driving Intelligent Networks

Telecommunications are no longer just services, economists refer to them as enablers. That is, they are more than just their cost input. Their true value vastly exceeds their monetary cost and their absence would lead to a shut down of operations. The presence of telecommunications is felt in diverse fields from business, politics, security, education, to the environment. The potential of telecommunications is still developing and their full value is yet to be realized. The past 15 years have witnessed momentous changes in telecommunications: the deregulation of the industry, introduction of high speed networks, and rapid penetration of mobile communications. Previously unheard of entities like independent service providers and re-sellers are major players. The INCM is a service development model that has helped fulfill the potential of telecommunications. In this section the impact of two trends - corporate telecommunications and mobile services - on the IN conceptual Model are examined.

2.4.1 Business Economic Drivers: Corporate Telecommunications

Methods of conducting business have changed world wide. Government regulation and control has lessened, allowing fast and responsive organizations. A regulated monopoly limits competition, as a result innovation is discouraged. This prevents offering customers efficient and customized advantages from telecommunications.

The IN service architecture provides a platform for rapid introduction of custom-made services, and has met with considerable success. The needs of the corporate customer, however, are different and are yet to be met. This is the first of the two trends that has greatly influenced the path of the telecommunications industry.

Corporate telecommunications (Figure 2.9) can been seen in two forms, external telecommunications and internal telecommunications, each having distinct characteristics. External telecommunications allow the corporation to provide services to their customer, clients and providers in the most efficient manner. An example of external telecommunication services is direct banking over telephone. A customer can transfer funds, check balances, and pay bills. The main objective of internal telecommunications, on the other hand, is the support of all activities internal to the business organization. The globalization of markets and businesses has resulted in an increased use of telecommunications in order to overcome the distance and time zone restrictions. Most large and many medium size business are geographically distributed, leading to a substantial amount of long distance traffic.

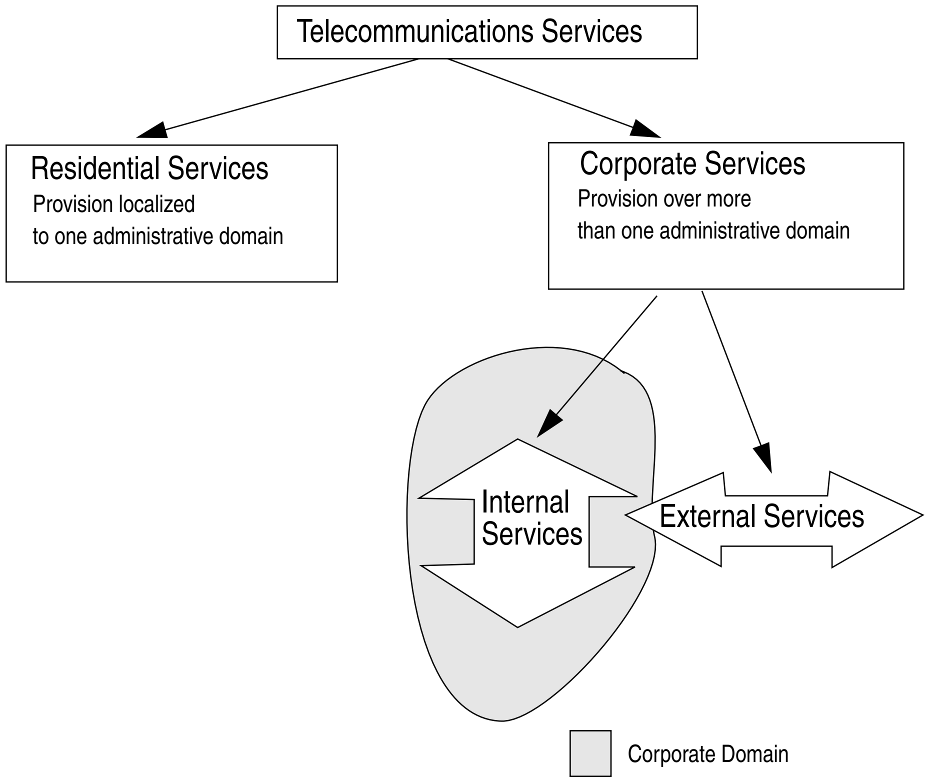


Figure 2.9 Telecommunication Services

For reasons such as cost, security and control, these organizations normally establish private telecommunications networks. A private network telecommunications service is provided by leasing telephone lines from the local telephone company. The use of such services, due to expense, is limited to large companies, with the disadvantage that the solution is vendor specific.

In a global, trans-national scenario internal corporate telecommunications services would have to be provided over domain boundaries. With the profusion of service providers in each geographical area, corporate telecommunications services need architectures for provision of joint services.

Currently an IN service cannot run beyond the geographical area covered by the service provider [MPG94a][CFO93].The non-uniform deregulation standards from country to country and region to region mitigate against development of co-operative services. There is a pressing need for the IN service platform to evolve architectures supporting modeling and development of co-operative services.

2.4.2 Technological Drivers: Mobility

Since the early 1980s, mobile telecommunications services have had an increasing impact of the development of the telecommunications sector. Numerous factors are responsible for the increase in mobility; among them: radio frequency allocations during WARC'92, reduction of weight and power consumptions through the use of VLSI, development of efficient transmission techniques.

Three concepts fundamental to mobile communications are personal mobility, terminal mobility, and service portability. Terminal Mobility refers to the ability of a mobile terminal to access a telecommunications services from any location while in motion, and the ability of the network to locate and identify the terminal as it moves. The terminal may move across administrative domains without losing connection. Personal mobility is the ability of end users to originate and receive calls and access subscribed telecommunication from any terminal in any location. Service Portability is the ability to transfer service and subscriber related data and logic across domains. Figure 2.10 shows terminal and personal mobility. Examples of currently available terminal mobility and personal mobility services are given in Figure 2.11 .

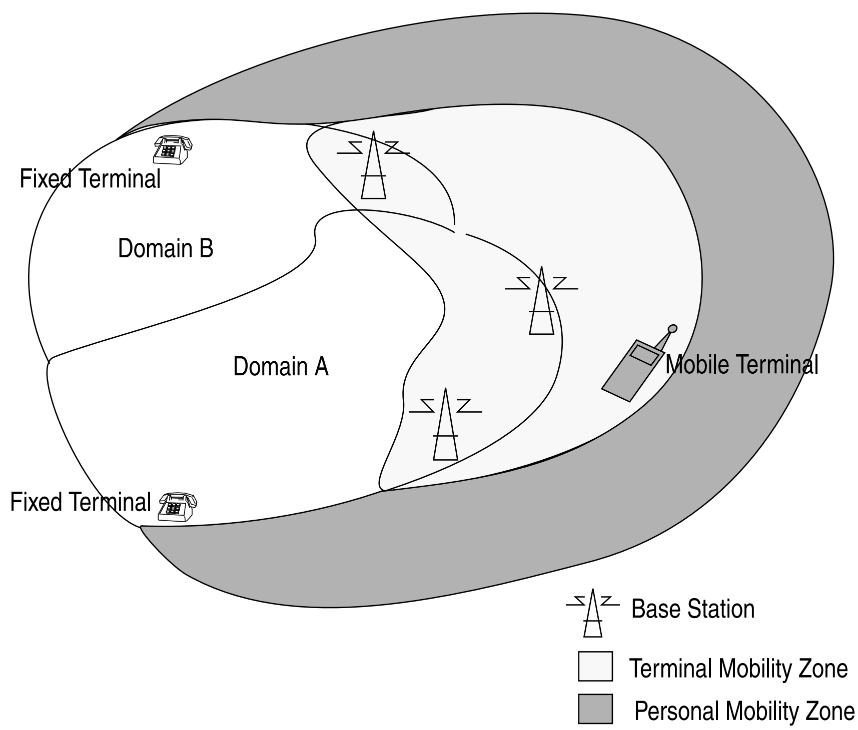


Figure 2.10 Terminal and Personal Mobility

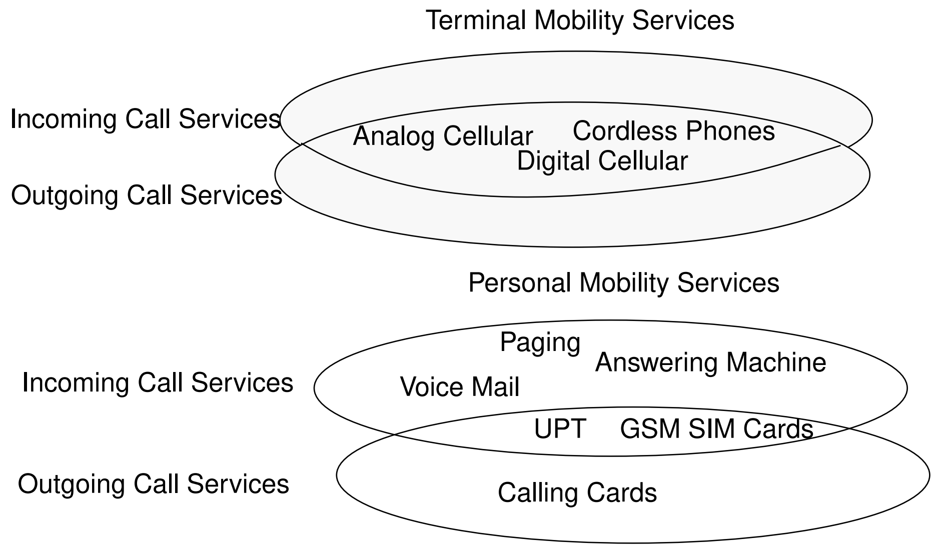


Figure 2.11 Examples of Terminal Mobility and Personal Mobility Services

In contrast to terminal mobility, the user of a personal mobility service does not need to carry a dedicated handset. Instead the user is required to register on a terminal by providing a personal identification number. The process can be manually carried out through a DTMF keypad. Once registration is complete, all facilities to which the user had subscribed become available on that terminal. Personal mobility is an important complement to terminal mobility. Terminal mobility has the added expense of purchase of a dedicated terminal, so personal mobility services will occupy the price gap that exists between fixed network services and terminal mobility services.

These two trends - internal corporate telecommunications and mobility services are driving factors in telecommunications. At the risk of becoming obsolete the INCM must evolve to support these demands.

2.5 Impact of Co-operative and Mobile Services on IN

The current state of IN standardization does not support the dynamics of technological change, marketplace transformation and customer demands as discussed in the previous section. A preliminary study in the feasibility of concepts for modeling and implementation of co-operative and mobile services, identified several areas requiring evolution. Two fundamental aspects of IN must evolve for support of co-operative services. The first aspect is distribution of service data and control. By definition a co-operative service would require co-ordination between more than one point of service control.

In the current IN CS-1 standard however, the service is invoked at the originating half or at the terminating half of the call. This invocation is controlled by only one SCF, which has access to all service data and service logic. The second aspect is inclusion of non call related functionality. IN CS-1 has focussed on call related functions only. The customers view of the service on the service plane is mapped onto SIBs on the GFP. The SIBs model describes the call setup as related to the service behavior. The service management procedures unrelated to the BCP are not modeled. However with the introduction of mobility, non call related functions like user registration and service profile transfer will have to be supported. As more complex services are created, service management aspects become more important.

2.5.1 Distributed Service Control and Data

Future development of IN is toward wider service coverage with interconnection between different network domains. Such interconnections require decentralized control. For inter domain services the use of SDF-SDF/ SCF-SCF/SCF-SDF relationships can be very beneficial since service data, subscriber information and service logic programs can be shared. Until today the SCF-SCF relationship had very low priority because it did not add any significantly new functionality to the IN capability sets. Some examples of modeling co-operative services exchange required information between domains through an unrestricted SCF-SCF/SCF-SDF link. This arrangement is clearly unsuitable, as security and autonomy are lost [Q.85].

An examination of interworking reveals that there is adequate uniformity at the three lower levels: physical, transport and routing. The challenge of IN is to interconnect networks at the service level. At the service designers perspective, it means co-operation with other domains for complementary service logic and data, without losing autonomy, control and security. From a customer perspective the expectation is that differences between network domains are not visible. This property is called seamlessness. Examples of seamlessness can be found in universal personal telecommunications (UPT) and the virtual private network (VPN). When a VPN service is implemented across domains it is known as the Global Virtual Network Service (GVNS). In the case of UPT, roaming requires prior agreements between carriers and a registration process. A UPT call setup would verify if the user is valid and route the call to the UPT users visited network. Thus a UPT user could roam from one location to another, and over multiple networks.

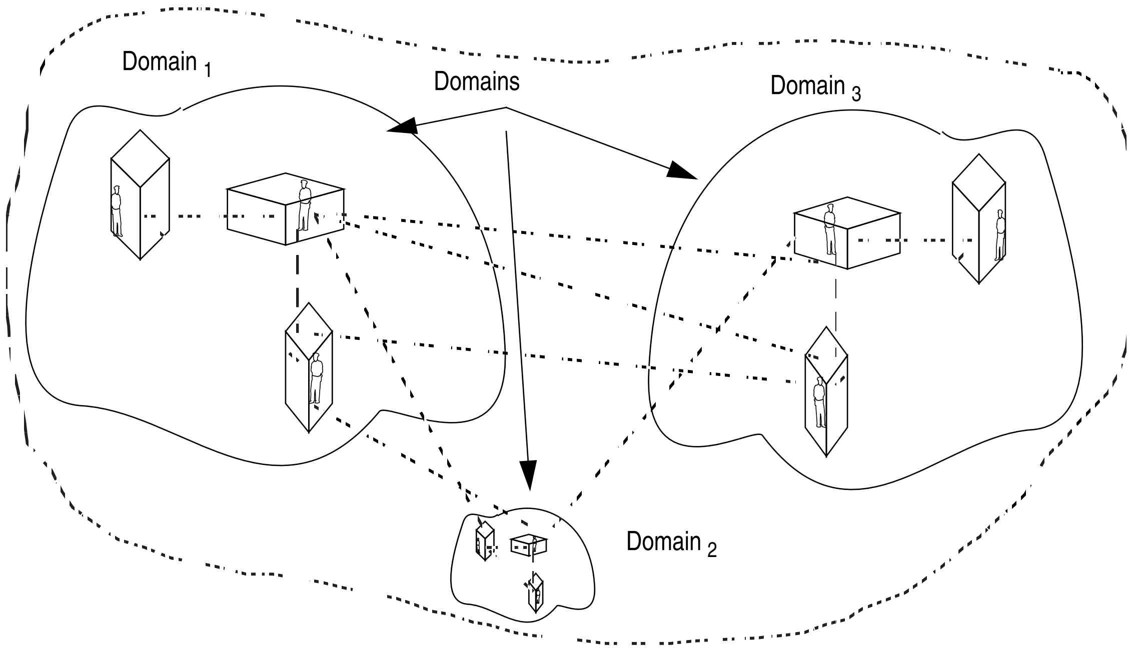


Figure 2.12 Global Virtual Network Service

If a GVNS is considered (Figure 2.12), similar actions would be required. Here a GVNS user in Domain 1 would like to access other locations in on the GVNS located in other networks (domains 2 & 3) This user desires a common signalling plan, common account information, and the same "look, touch, and feel" worldwide.

2.5.2 Distributed Services and the Problem of Control

The approach of IN, as with many other distributed systems, is to mask out all problems of the distributed system [RB91]. The Global Function Plane hides the network and all related distribution aspects. As a result, information regarding distribution, for example, the location of a user or differences in naming schemes, is not visible while modeling the service. This approach does not accomodate complex patterns of management of distribution in co-operative services. (An example of such patterns would be security policies which involve a siginificant variation with user context) In addition, the presumed control decisions are embedded into the system and hence cannot be tailored to specific services. Because of the dynamic requirements of GVNS and co-operative services, it will be unlikely that such prescribed solutions of full transparency will be suitable.

Another suggested alternative approach is that of non-transparency [CH94]. In non-transparent systems, all features of distribution are visible to the service designers. They must deal with all issues such as failure, naming and access. This allows much more flexibility since individual services can deal with all distribution aspects. Services can exploit information on the nature of cooperation supported by making all management decisions. The biggest limitation with the non-transparency view is that the handling of the distribution can become an intolerable burden on the service engineer.

2.5.3 Non Call Related Functions

Up to now the IN platform has addressed services which are applied to calls that are in the process of setting up, already established or clearing down. Hence there is emphasis on the basic call state model in the switch. However mobility features such as user registration, location updating and service profile transfer can occur when there is no active call. IN CS-1 covers call related functions only. In this case the SSF receives triggers from the CCF which represent certain points in call. For non call related functions, the SMF will have to communicate with the SCF. When available a whole new range of capabilities will be within reach. This would enable not only location updating, but also allow users to update their personal profiles.

When a new user enters a new network, extra functionality is needed unrelated to bearer control. The old and new network have to negotiate whether the mobile user has rights (roaming agreements, valid subscription) to access the new network and whether the new network can offer the capabilities required by the mobile user. This and other related procedures are under the purview of service management and administration.

While CS-1 focussed exclusively on invocation of services, subsequent capability sets attempt at defining and linking of all phases within the scope of service management. . In CS -2 management and creation relationships are discussed to the functional level only, i.e., entities responsible for management and creation - the SMF, SMAF, SCEF - and related functionality are defined., but the protocol model is not discussed. The following aspects are postponed until the CS-3 time frame: IN service creation, IN specific management, Internetworking between IN structured networks.[Q.12xx]

The area of service administration and management is a key area that will impact the future evolution of IN. So far, this area has not been discussed in IN. Administration and management must follow the same rules of flexibility as IN services. This means that the ability to add rapidly a new IN based service depends on fast and easy updating on an entire chain of systems involved, including technical and administrative systems.

On the harmonization of Intelligent Networks and Telecommunication Management Network

Under the aegis of ITU-T there is some work under progress on the management of IN[TM93]. “The Baseline Document on Integration of Intelligent Networks (IN) and Telecommunication Management Network (TMN)” proposes a mapping of IN functional entities onto TMN function blocks and the integration of IN and TMN modeling techniques. This document suggests basing IN management on TMN. IN defined functional entities can be mapped on to TMN functional blocks; and management activities originally ascribed to the IN functional entities can in fact be replaced by TMN functionality. Therefore the document suggests replacing IN FEs with more general TMN functional blocks, so as to clarify and promote the reusability of the TMN defined functions for IN. However the TMN concept merely aims at identifying interfaces between systems and lacks the concept of service creation and application building blocks[KJ95]. This makes it difficult to rapidly develop new IN services. The same document recommends that TMN be expanded to incorporate application oriented concepts. The discussion of IN and TMN harmonization is not within the scope of this work. The focus instead is on general purpose models for service creation and management.

2.5.4 Impact of Mobility on IN Service Design

Increasing mobility for both the calling and called party will be the major re-shaper of IN services. New services will be added, some old ones will be updated, some split into one or more services and some will be made redundant. In fixed network telephony, dialing a number refers to one location, one terminal and one person. When mobility is introduced in fixed networks, the calling line and called party will no longer connect with the physical network access points. Instead they must be re-structured into more than one role.

Consider the example of the call forwarding service. Call forwarding can be used since it provides two functions. The first is to redirect important calls to another number where the user has temporarily relocated. The second is used in the unavailability of the called party, where another user receives or attends to the calls. The first function will become redundant as the user will be able to receive calls irrespective of current physical location.

A number of mobile service features have been offered by providers have offered recently. These are however ad-hoc attempts to provide mobility. Feature interactions are a severe problem in telecommunications systems and receive a great deal of research attention. A principle that can be used to direct the decision about how a feature should behave is: a feature should fit and address a recognizable human situation as described in terms of its roles. The service designer must find a way to describe a desired feature behavior that is complete, consistent, unambiguous and above all, maintainable as new features are introduced. Consider the characteristics that a successful model must have [PZ96]: The right abstractions must be chosen so that they eliminate rather than obscure important issues. Each abstraction must reduce description complexity or the entire specification may become unreadable.

This is a critical modeling aspect absent from IN models. The standardization document Q.1290 on IN terminology does not include contexts for mobile services. Instead, the Q.12xx series lists 63 imprecisely defined or understood sets of services. The first step to reduce complexity in IN service development is to define entities, roles and related contexts. Mobile services should then be developed using these well understood abstractions. This thesis is a development of such a framework for mobile services.

2.6 Conclusion

This chapter was an introduction to IN concepts. It presented a number of challenges facing IN. IN must evolve on all planes to support telecommunication trends. This section has discussed two of the trends - cooperative services and mobile services - that will impact the global function plane and the distributed function plane. As will be discussed later in this thesis, three aspects must be considered for further alignment:

- In the absence of service management and creation concepts, current approaches to IN service creation lead to imprecision and complexity. The set of target services are not well defined and understood. Service management and creation must incorporate roles and contexts for a more manageable development platform. Chapter 4, Service Portability in Mobile Telecommunications Systems examines these issues and proposes alternative models. Section 4.2 proposes a new perspective to service modelling and service profiles.
- IN modeling must evolve to support inter-domain information configuration through the SCF-SCF relationship. This relationship is developed by use of ODP trading, especially interworking traders. ODP trading is introduced in Section 3.2.1 Trading . The application of ODP is clarified by use of an example in Section 6.1.2 BetterMouseTraps GVNS Trading Graph. Additional requirements for interworking are met by a new proposal : the Interworking State Model. The Interworking State Model is acontribution of this thesis and is explained in Section 6.1.3.
- The limitations of masking all distribution aspects on the GFP must be recognized, and approaches other than full transparency and non-transparency must be considered.

Chapter 3
Introduction to Reference Model for Open Distributed Processing

3.0 Introduction

Advances in computer networking have allowed computers all over the world to be interconnected. Despite this, heterogeneity in interaction models prevents complete interworking of systems. The *Reference Model for Open Distributed Processing (RM-ODP)* aims to enable interworking of heterogeneous systems by providing a common interaction reference model.

The ISO and ITU-T are developing RM-ODP to allow interoperability between heterogeneous systems through a reference architecture [X.900].

The focus of RM-ODP is on the development of a reusable set of functionalities with well defined interfaces. These well understood functionailities can be combined in various configurations to build required application structures (RM-ODP provides reference abstractions only and makes no attempt to actually implement these functions). Though the current (1996) standards are incomplete, significant work has been done in developing conceptual frameworks and identifying reusable functionality. The reference model organizes the open system into a coherent unit. The components are carefully described without prescribing the implementation or unnecessarily influencing technology.

This chapter is an introduction to the ODP, with emphasis on concepts reused for IN co-operative service development in Chapter 5 and Chapter 6.

3.0.1 Scope of ODP Standards

Given the pace of technological development, heterogeneous systems are inevitable. Legacy systems, though implemented with older technology and providing limited functionality, work effectively. As the performance and functionality demands increase, users of these systems wish to evolve these systems by adding new functionality as it becomes technologically available and economically feasible. Therefore, old and new components will *always* have to interwork. Thus interworking and evolution are of fundamental importance.

The scope of Open Distributed Processing can be summarized as:

- interworking between ODP systems, i.e., meaningful exchange of information and convenient use of functionality throughout the distributed system.
- graceful evolution.
- scalability.
- distribution transparency, i.e. hide the consequences of distribution from both the application programmer and the user.

• portability of applications across heterogeneous platform.

RM-ODP provides conceptual frameworks, called *viewpoints* to *understand* a system. It identifies *generic functionality* that can be *reused* for applications development. The section 3.1 "RM-ODP Viewpoints" introduces the concept of viewpoints and section 3.2 "ODP Functions" discusses functionality reused in the context of this work.

3.1 RM-ODP Viewpoints

In order to reason, understand, design and implement ODP systems, various perspectives of the system need to be used. Conceptual frameworks have been developed over the years, but since they usually provide only one perspective of a system they are incomplete. ODP provides the notion of *viewpoints*. A viewpoint is *a form of abstraction achieved using a selected set of architectural concepts and structuring rules, in order to focus on particular concerns within a system* [X.900]. Five viewpoints are defined in [X.900] and are:

- *Enterprise Viewpoint*: An enterprise viewpoint defines the purpose, scope and policies of an ODP system.
- *Information Viewpoint:* An information viewpoint defines the semantics of information processing in an ODP system.
- *Computational Viewpoint:* A computational viewpoint defines the functional decomposition of an ODP system into objects which interact at interfaces. In the computational viewpoint, applications and ODP functions consist of configurations of interacting computational objects.
- *Engineering Viewpoint:* An engineering specification defines the mechanisms and functions required to support distributed interaction between computational objects in an ODP system.
- *Technology Viewpoint:* A technology specification defines the choice of technology for an ODP system

A viewpoint can be seen as perspective that reflects the concerns of a set of users. Each viewpoint is modeled by a set of concepts, structures and rules. It focuses on related issues, and is, in isolation, incomplete. A coherent system specification emerges when all five viewpoints are brought together. Thus, complete ODP system must specify all five viewpoints.The relationship of the RM-ODP viewpoints to software engineering processes, as illustrated in [KR95] are shown in Figure 3.1 .

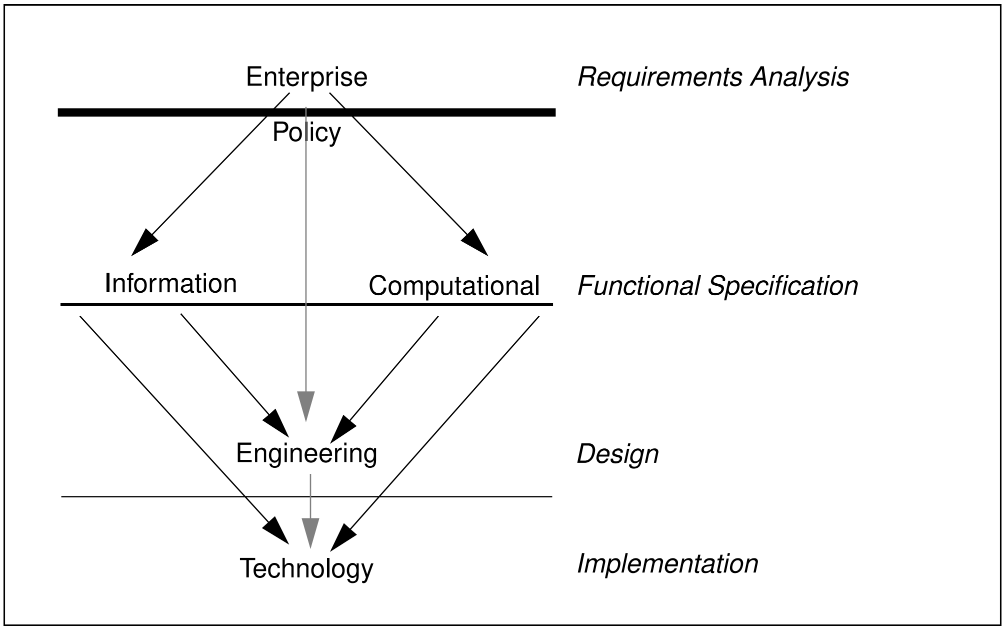


Figure 3.1 RM-ODP Viewpoints and Software Engineering

3.1.1 The Enterprise Viewpoint

The non technical issues of a system, are placed in a system specification called the *Enterprise Viewpoint*. The aim of the Enterprise Viewpoint is to discuss the purpose, scope and objectives of the system from the organizational perspective. The enterprise viewpoint is a basic framework from which organizational policies can be defined. Its scope is vast and covers common business practices and laws.The enterprise viewpoint deals with the definitions of concepts common to many businesses. The semantics are generic enough to be applicable for the description of business events, policies and processes across many types of enterprises. A purpose of the Enterprise Viewpoint is to facilitate inter organizational contract activities.

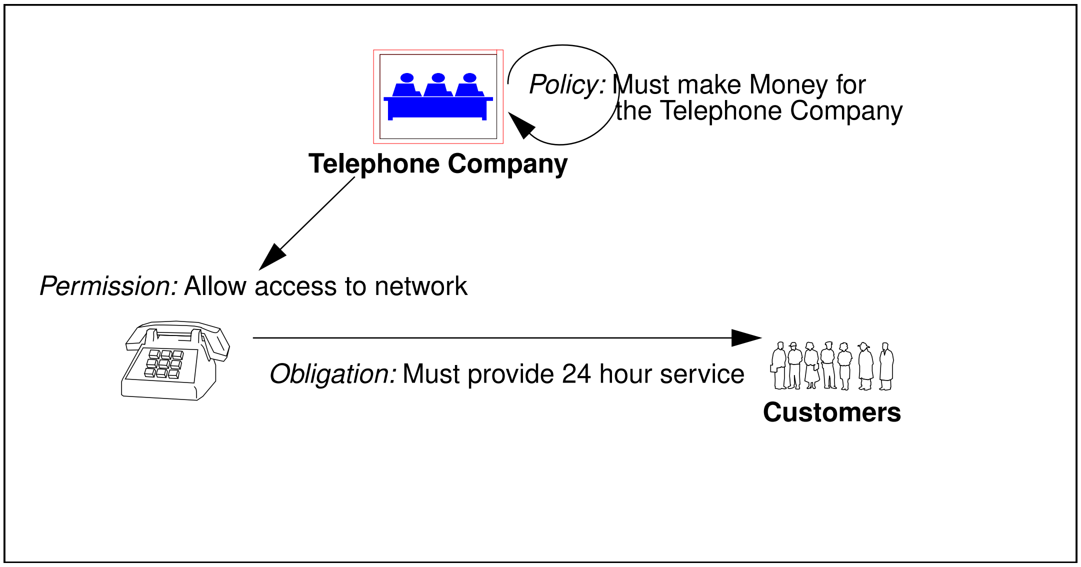


Figure 3.2 An example of an ODP Enterprise Viewpoint

The current ODP enterprise viewpoint language is not mature enough to formally specify ODP systems. Therefore an enterprise specification is normally described using textual descriptions and diagrams.

An example of the enterprise viewpoint of an ODP system is shown in Figure 3.2 . To illustrate the enterprise viewpoint, consider an example of a telephone company that offers fixed terminal telephone subscriptions to the public (Figure 3.2). The enterprise viewpoint shows high level view of the system: the telephone company, the terminals and the customers. Also shown are related policies, obligations and permissions.

- *Policy:* The service must make money for the telephone company.
- *Obligation:* Customer must have 24 hour access to the service.
- *Permission:* Allow terminal to access the network.

In the enterprise viewpoint, specifications are restricted to a basic set of concepts and rules addressing the scope of the enterprise activities. It is important to put the activities in the right context, to identify the on-line (real time) and off-line (non-real time, contractual) relationships and relate activities to domain interfaces.

The enterprise viewpoint is specifically concerned with *performative* actions that change policy [KR95]. Creating an obligation or revoking a permission is a performative action. In a telephone service provider environment, a subscription is seen as a performative action, since this creates an obligation for the telephone company to provide the service. A call is not seen as a performative action, as obligations, permissions and prohibitions are not affected. An enterprise viewpoint should not include the setup of calls; such functionality is identified in the Information or computational viewpoint.

In the enterprise specification, policies are determined by the organization rather than imposed by technology choices.

The basic enterprise language concepts and structuring rules as introduced in the

RM-ODP standard [X.900] are:

- *Community* - a configuration of objects formed to meet an enterprise objective.
- The *Objective* is expressed as a contract which specifies how the objective can be met.
- *Federation* - a particular kind of community. This is a coming-together of a number of groups administered by different authorities in order that they may jointly cooperate to achieve some objective.

Scope of Enterprise Viewpoint as applied to IN service modeling.

The IN service plane specifications are prose description hence potentially vague. The IN standards [CS1] does not provide any high-level modeling concepts. The service is seen from the user's point of view *only*. In contrast, the ODP enterprise model offers additional concepts and functionalities to highlight relevant issues in the enterprise models. The ODP enterprise viewpoint recommendations on *inter-domain contractual specification and inter-domain management policy [X.900]* can be used in IN service development. These are especially valid in a mobile and co-operative environment where the key issues, as discussed in section 2.5 , are the interworking of administrative domains.

3.1.2 The Information Viewpoint

The Information Viewpoint focuses on the information content and information processing activities of the system. The information modeling activities consists of identifying:

- information structures of the system, i.e. *static schema*,
- constraints on these information structures, i.e. *invariant schema*
- information flows between information structures,.e. *dynamic schema*.

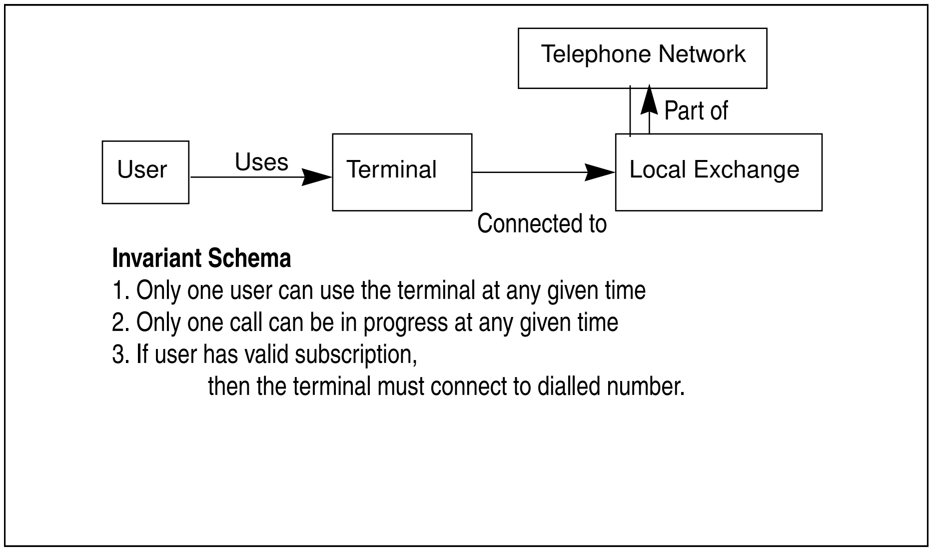


Figure 3.3 ODP Information Viewpoint

Figure 3.3 shows the information viewpoint of the telephone environment. The information objects identified are the user, terminal, the local exchange and telephone network. Three Invariant Schema are specified:

- 1. Only one user can use the Terminal at any given time
- 2. Only one call can be in progress at any given time
- 3. If user has valid subscription, then the terminal must connect to the dialled number.

The information specification of an ODP system can be represented by a variety of methods: entity-relationship models, Z, or other techniques.

Scope of Information Viewpoint application in IN service modeling

The global function plane, (section 2.2.2), uses SIBs and GSL (Global Service Logic). A SIB is an information structure and the global service logic represents information flows. Therefore a GFP could be considered an information model. However, a fundamental limitation with SIBs, BCP and GSL is that only service at originating or terminating end of the call are modeled. The network is considered global, and all distribution details are hidden. As a result, co-operative services between interworked domains, where representation of distribution dimensions is essential, cannot be modeled. This point is discussed in greater detail in section 5.1 “Introduction to Integrated Modeling of IN and ODP”. This thesis, unlike the GFP, develops an information model that reflects the distribution patterns of co-operative services (Section 6.1 “The GVNS ODP Information Model Viewpoint”).

In this thesis we use Object Modeling Technique to show object structure and Event Diagrams to show information flows. It is assumed that the reader is familiar with

OMT and Event diagrams.

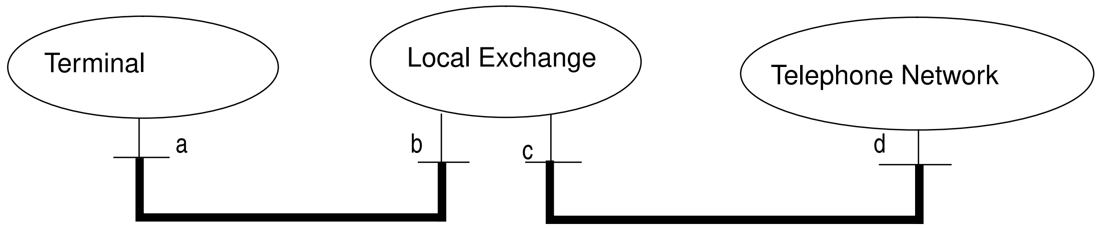
3.1.3 The Computational Viewpoint

The *Computational Viewpoint* represents the distributed system as seen by application designers and programmers. In the computational viewpoint, applications and ODP functions consist of configurations of interacting computational objects.

The ODP model introduces the concept object as a unit of encapsulation and distribution. The purpose of an object is to provide a service to its clients. The set of services provided by an object is called the behavior of the object. Procedures provided by the server define a set of operations providing a service. The point of access to those operations is called an interface. Thus the separation of the service provided by a server object from the technology implementing the service is the primary benefit of this viewpoint. Because of potential separation between components of a distributed application all interactions must be of the request-response type. In any interaction, there must be a sender of a request (the client) and the recipient of the request (the server).

This Computational Viewpoint is a high level view for applications. It a hides the actual degree of distribution. This characteristic is called *distribution transparency*. To ensure distribution transparency, all access to objects should be made by references to the interface of the object. Such indirect addressing allows appropriate communications to be inserted transparently in the path between client and server. These communication channels for distribution transparency support are designed in the ODP engineering viewpoint. While the computational model is concerned with *what* an interface must provide, the engineering model concerns with *how* the interface is to be implemented.

Figure 3.4 shows the computational view of the telephone system. 3 types of objects are shown: a terminal, a local exchange and a telephone network. The interfaces are arbitrarily numbered a,b,c & d. Further, each object has a role, a signature which corresponds to the syntax of the interactions at respective interfaces and a behavior. Figure 3.4 is an example only and the objects and attributes have no direct relation to subsequent material.



Object	Interface	Role	Signature	Behavior
Terminal	a	client	DTMF Tone	Terminal Client
Local Exchange	b	server	DTMF Tone	Switch Server
	c	client	CORBA IDL	Connection Client
Tele-phone Network	d	Server	CORBA IDL	Network Server

Figure 3.4 ODP Computational Viewpoint

ODP Distribution Transparencies

The principle requirement of distributed systems is that masking from applications the differences in mechanisms used to overcome problems and details of distribution. This is called *distribution transparency*. Transparency is different from security. Security implies that the user must not know underlying details, while transparency implies that the user may choose not know the underlying details.

Distribution cannot be denied, programmers may have to deal with variations in distribution dependent factors like concurrent access and latency. However, the ODP architectural aim is to minimize the extent to which applications are cluttered by details of underlying platforms. Since some applications may need complete control over distribution to participate directly in its provision, transparency must be declarative and selective. Transparency becomes possible since in ODP interfaces between applications and between applications and their supporting infrastructure are simple and uniform.

In ODP, transparencies are defined as constraints on the mapping from a computational specification containing a transparency schema to a specification that uses specific ODP functions and engineering structures to provide the required form of masking.

Scope of Computational Viewpoint application in IN service modeling

This work does not use computational concepts. Some recommendation regarding the application of ODP computational viewpoint to IN described in chapter 7.

3.1.4 Engineering Viewpoint

With the computational model, there needs to be an engineering model to organize a system for transparency. From the computational viewpoint, transparency requirements are expressed as environment constraints within interface specifications. The environment constraints detail the properties required of the engineering infrastructure. The engineering model defines basic system functionality that may be used to support the computational viewpoint. The issue of the computational viewpoint is *what* interfaces need to be provided, and the concerns of the engineering viewpoint are *how* this will be accomplished.

Scope of Engineering Viewpoint application in IN service modeling

Although ODP engineering ideas can make significant contributions to IN, they are omitted here for the sake of manageability. Recommendations about applications in the IN context are discussed in chapter 7.

3.2 ODP Functions

The Reference Model specifies the functions required to achieve Open Distributed Processing. These functions are essential to support distribution transparency. The standard [X.900] provides an extensive set of functions, such as *management functions, co-ordination functions, repository functions and security functions*.

Each ODP function description contains:

- an explanation of the use of the function for open distributed processing.
- a statement of other ODP functions upon which it depends.
- prescriptive statements, about the structure and behavior of the function, sufficient to ensure the overall integrity of the Reference Model.

This thesis makes use of the *ODP trading function* for IN co-operative service development.

3.2.1 Trading

RM-ODP aims to provide development of distributed applications over heterogeneous software and network architectures. In order to utilize services in an open distributed system, potential clients must be aware of potential servers and be capable of accessing them. Also since sites and applications are frequently changing in large distributed systems, it is necessary to provide late binding between providers and consumers. With the support of late binding, a software component can find service providers dynamically.

Mechanisms that assist in locating resources are of critical importance. The ODP *trading function* enables this dynamic selection of appropriate services at run time.

A trader is a third party object that enables clients to find servers.

Figure 3.5 shows principal interactions for late binding between clients and servers using a trader. A server that offers services through a trader is called an *exporter*, while a client that requests a service through a trader is called an *importer*.

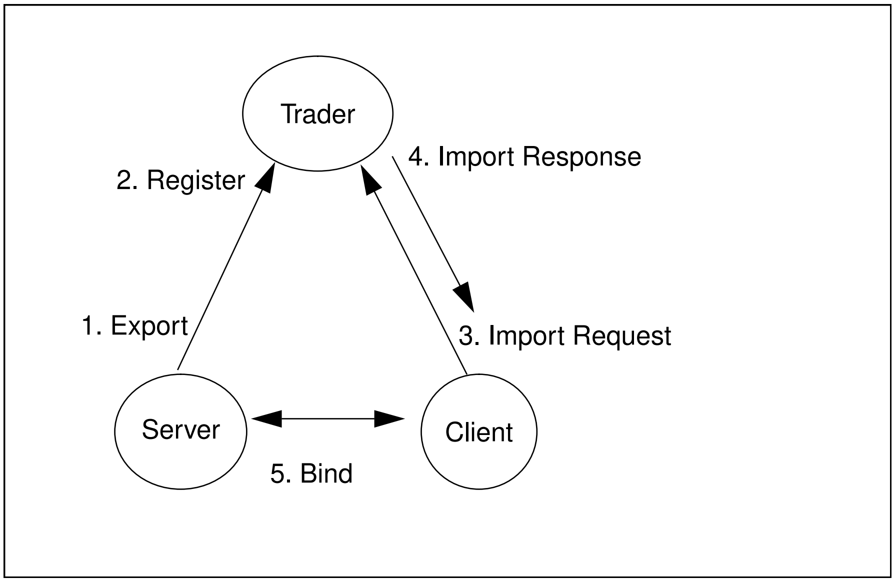


Figure 3.5 Late Binding Using ODP Trading Concept

Service discovery at run time using the ODP trading function is performed in the following steps:

1. *Export:* A trader accepts service offers from exporters of services when exporters wish to advertise service offers. A service offer contains the characteristics of a service that a service provider is willing to offer.
2. *Register:* Service offers are stored by the trader in a database.
3. *Import Request:* The trader accepts service requests from importers when importers require knowledge about appropriate servers. A service request is an expression of service requirements made by the importer.
4. *Import Response:* The trader searches its database to match the importers service request. If required, the trader can select the most appropriate service offer that satisfies the importers request.
5. *Bind:* After a successful match, the client directly interacts with the server.

This matching of appropriate services allows objects to be configured at run-time. The same object can be an exporter and an importer at different instances of time. The set of services that are registered with a trader are called the traders *service offer space.*

Uses of Trading: An Example in Telecommunication Services

In a telecommunication services environment, like IN, a high degree of customization is a key aim. In the case of compile time or link time binding, every time the subscriber customizes the service configuration, a re-compilation will have to be done. Therefore for management and efficiency, run time discovery and binding is necessary. A subscriber may buy the subscription with one configuration, and at a later time, need a different set of services. This need is fulfilled by asking the trader for services with certain characteristics.

Trading Criteria and Constraints

All activities within a domain are governed by *trader properties*, which are used in specifying trading criteria and trading policies [MYB 95] concerning the use of service offers. The set of exporters from whom services are accepted are governed by the *service offer acceptance policy*, and the set of importers from whom import requests are accepted are governed by the *import request acceptance policy*. Associated with each domain is a trader-administrator that enforces this policy [PM95].

Interworking Traders

Traders are said to *interwork* when they are linked to *share the service offer spaces without loss of autonomy*. Traders form an interworking group to enlarge the service offer space available to users in one or all of the inter-worked domains. Interworking of traders is performed at two levels

- *Between administrators.* This level of interworking is performative (Section 3.1.1). It is done only when interworking is configured for the first time or when changes are made. This level of interworking normally has no real time constraints.
- *Between traders.* Service discovery is propagated between traders at this level of interworking.

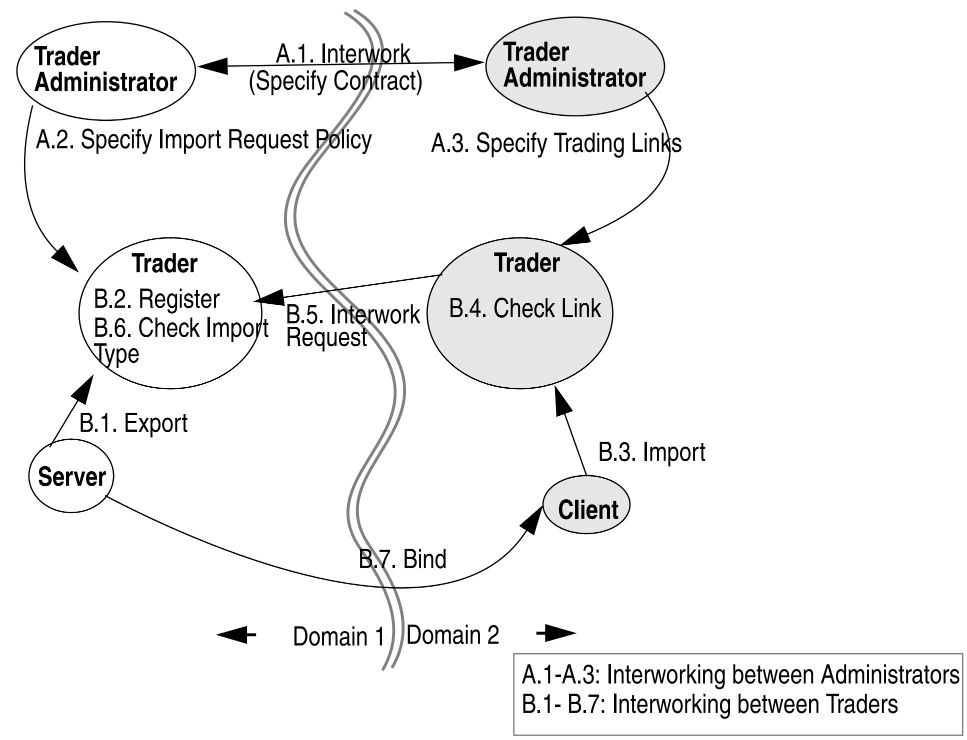


Figure 3.6 Interworking Trader

The following sequence, shown in Figure 3.6 , is used to interwork traders.

Steps A.1-A.3 are between administrators. These steps are executed only at initialization or at re-configuration. The steps B.1-B.7 are between traders, and are executed, every time a client in one domain binds with a server in an interworked domain.

A.1. *Interwork between Administrators.*

The administrators develop and specify enterprise level policies and contracts. Domain 1 is the exporting domain, Domain 2 is the importing domain.

A.2. *Specify Import Request Policy.* The trader administrator of domain 1 specifies the Import Request Policy and informs the trader. This policy specifies types of interworking requests to be accepted.

A. 3. *Specify Trading Links.* The trader administrator in domain 2 informs the trader of links to the interworked domain.

B.1. *Export.* A server in domain 1 exports a service.

B.2. *Register.* This service is registered in the trader database

B.3. *Import.* The client in domain 2 requests for a service.

B.4. *Check Trading Link.* The trader checks its links and initiates an interworking request to the trader in domain 1.

B.5. *Interwork Request.*

B.6. *Check Import Request Policy.* On a receiving an import request the trader in domain 1 checks is Import Request Policy. Since the request is from an acceptable domain (Step A.1 & A.2), the server identifier is returned to domain 2.

B.7. *Bind.* The client and server bind.

The ODP interworking model is decentralized and non-hierarchical. Each trader manages its own offer space. Interworking between domains use contracts. Interworking contracts are asymmetric and between an exporting trader and an importing trader. The exporting trader offers access to offer space, possibly with restrictions. The importing trader uses exporting traders trading space, also possibly with restrictions. Contracts permit incremental change in any trader without risk to other traders.

The group of interworking traders is linked to form a trading graph, which provides information on the next trader to be accessed in the search for an import operation. It is not necessary for a user to know the distribution details of the service. The user of a trader places requests on one trader only, and transparently accesses other traders in the established interworking group via established links.

Application of Interworking Traders to IN Service Development

In a mobile environment, users will access their services from various domains. Each time a user registers in a visited domain, authentication will have to be performed. One solution is to store authentication servers for all users in all domains interworking to provide global mobility. This does not scale well:

If N *domains are interworked,*

supporting a total of m *users*

then total number of authentication information entries are Nxm.

And

Every time a subscriber is added or removed, N *changes are made.*

An elegant solution can be provided using interworking traders.Each domain stores its local subscriber authentication information in an authentication server. When a user visits a domain, the visited domain checks interworking links, places an import request for the home domain authentication server. If authentication is successful then registration is successful. The absence of an interworking link to the home domain would indicate to the visited domain trader that interworking contracts are not in place and authentication would fail. On invocation of an interworking request, the home domain trader would check the import request acceptance policy. If this policy fails to recognize the origin of the request, it would indicate that the request if fraudulent.

In this scenario:
If N *interworked domains are interworked*
supporting a total of m *subscribers,*
then a total authentication entries are stored is m.

Every time a subscriber is added or dropped, only 1 *change is made.*

This example which illustrates a limited example of an authentication server, can be expanded to included all Service Logic Programs (SPLs) available in a IN domain. Each exported SLP would be represented as an entry in the traders service offer space.

As seen above, ODP traders offer a non-hierarchical interworking model. Each domain retains control of its internal space, but can still freely interwork. Contracts and policies can be specified incrementally without affecting other traders. These characteristics meet the requirements identified in section 2.5.1 "Distributed Service Control and Data" and in section 2.5.2 "Distributed Services and the Problem of Control".

Enterprise Viewpoint of Traders

A trader is a member of a community established for the purpose of trading. The enterprise specification identifies a set of policies that limit the trader to a certain type of behavior. The focus of this study is on interworking IN domains, hence the ODP recommendation on interworking policies [X.900] are of interest. The details of reuse of ODP Trader interworking are shown in Chapter 5.

A group of interworking traders is a community with their respective administrators, importers and exporters.

For traders to be a member of an interworking group, the following ODP recommendations are used:

- one trader is not obliged to perform an activity for another trader
- each trader must have complete control over its trader policies.
- when a trader joins a group of interworking traders, all members are obliged to be a part of the group.

Information Viewpoint of Traders.

In the information specification, the information structures and information handling of the trading function are specified.

The trader accepts exported service offers as part of its trading offer space. A trader with its service offer space is represented by a node. An edge between two nodes specifies a relationship between the nodes. It is unidirectional and connects a source node to a target node. The nodes and edges from a *trading graph*, which governs the propagation of a request through connected nodes. The trading graph represents the distributed knowledge of trader objects of other trader objects in a group of interworking traders. The nodes of the graph represent traders. The state of a trader is described by:

- a set of service offers (dots in Figure 3.7), offers from its local community and offers from remote traders
- a set of trader properties (crosses inFigure 3.7), rules that guide its behavior.
- a set of links, (bars in Figure 3.7), pointers to adjacent traders.

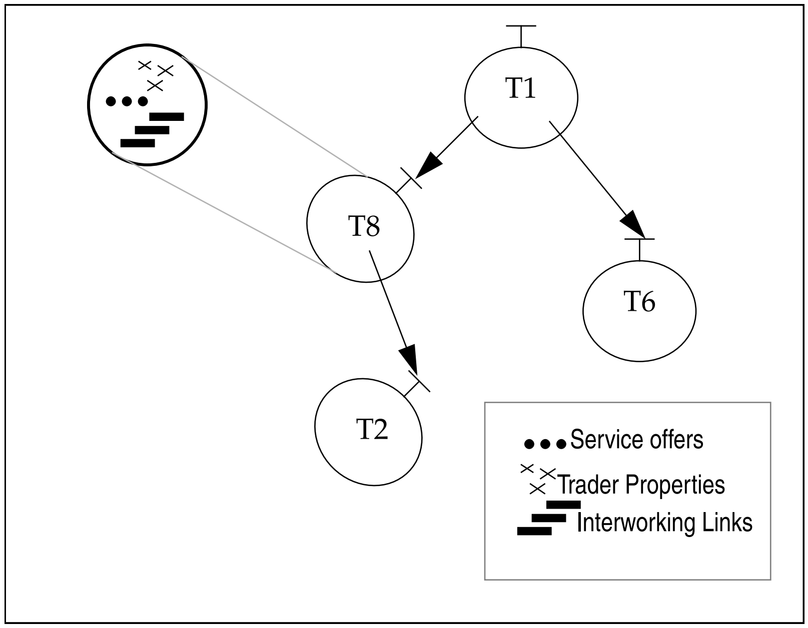


Figure 3.7 ODP Information Viewpoint of Interworking Traders: A trading graph.

The edges in the graph represent knowledge of adjacent traders. The links have properties such as Import Request Policy.

In the information viewpoint, the design of the trading graph is important. If each node represents an IN domain the trading graph illustrates the propagation of requests for service functionality. Graphs can be traversed using various algorithms such as, sequential breadth or depth first, or parallel. The traversal algorithm should avoid:

- *duplication:* a same request reaching a trader by more than one path
- *loops:* a request returning to a trader that initiated it.

Nodes are mapped into traders in the computational viewpoint, where one node corresponds to a trader object. Edges are mapped into links in the computational viewpoint, where a link in a trading graph corresponds to an arc between two interworking traders.

Computational Viewpoint of Trader

In the computational Viewpoint the interfaces of a trader with its environment are visible. The computational specification defines an object template for a trader. To facilitate interoperability of traders the trading function has identified a set of standard properties. These properties have a standardized name type and semantics [MYB95]. In the computational specification, operation signatures are specified using the CORBA IDL and the operation behaviors are described using policies.

A link with a property, such as import request policy are implemented using computational abstractions.

Engineering Viewpoint of Traders

A simplistic implementation of an abstract interface in the computational model would be very inefficient. Several optimizations would be made in the engineering viewpoint. A number of ODP trader engineering implementations have been shown [WB95][RH95] using the X.500 OSI directory service standard. Current IN standardization activity (CS2) is considering X.500 for interworking requirements. Therefore ODP application to IN using X.500 has significant potential as further work.

3.2.2 Conclusion

This chapter introduced the basic ODP framework. ODP trading will be used subsequently in Chapter 6 for the development of distributed services.

Chapter 4
Service Portability in Mobile Telecommunications Systems

4.0 Introduction

One of the ultimate goals of mobile telephony is to allow subscribers access a set of subscribed services from all locations, without interruption of services during location changes. Ideally, the look and feel of these services should be preserved as the subscriber roams across administrative and technological boundaries. Since telecommunication services require custom service logic programs and data, mobility support infrastructure must provide the ability to access these logic and data. Such support is called service portability [RP95]. Service portability requirements become more demanding when both personal and terminal mobility are provisioned.

This chapter discusses service profiles in a mobile environment. Section 4.1 discusses mobility in the IN context; reviews current approaches and highlights associated problems. Section 4.2 and proposes enhanced concepts from the ODP perspective. The conclusions are presented in Section 4.3.

4.1 Service Portability in IN

The ODP perspective of services is vast: any functionality offered to users is a service. In IN the understanding of a service is more precise: an IN service is additional functionality offered in addition to the basic voice call. The service is implemented using IN capabilities and meets IN constraints and restrictions. An IN domain is an authority that provides IN capabilities so as to support a set of IN services. Every IN is identified via a unique IN operator identifier.

Two roles identified for IN domains are *home domain* and *visited domain*. A home domain is the domain with which a user has a subscription. It provides user profile related information to other IN domains and collects charging information from other operators for its roaming users. A visited domain is any domain except the home domain in which users are allowed to roam and use IN services. A user roaming in a visited domain may use available resources in a visited domain. Charging information concerning use of resources in a visited operators domain is transferred to a home domain

Three aspects - terminal mobility, personal mobility and service portability - are key in understanding mobile telecommunications. *Terminal mobility* is the ability of a terminal to move during a call. *Personal mobility* refers to the ability of a user to register, access services and call from different fixed and mobile terminals. *Service portability* refers to uniform service behavior and user-network interfaces across domains. This requires transfer of verification, authentication and subscription related information from a home network to a visited network. The verification, authentication and subscription related information is known as the *Service Profile*. The transfer of the service profile between networks is known as *Internetwork Service Profile Transfer* (ISPT) [RP95].

ISPT is performed to minimize control signalling between networks. In the absence of service profile transfer, the home network would have to be queried every time a call is established or a service invoked. Figure 4.8 compares signalling between network domains in the absence and presence of ISPT. In the absence of ISPT (fig 4.1.a), every time a mobile user or terminal makes a call, the visited domain has to query the home domain for authentication and verification. In the presence of ISPT (fig 4.1.b), the visited domain queries the home domain the first time a call is requested. After this, the subscription and authentication information is transferred to the visited domain. Interdomain signalling is minimized as the service profile is now locally available.

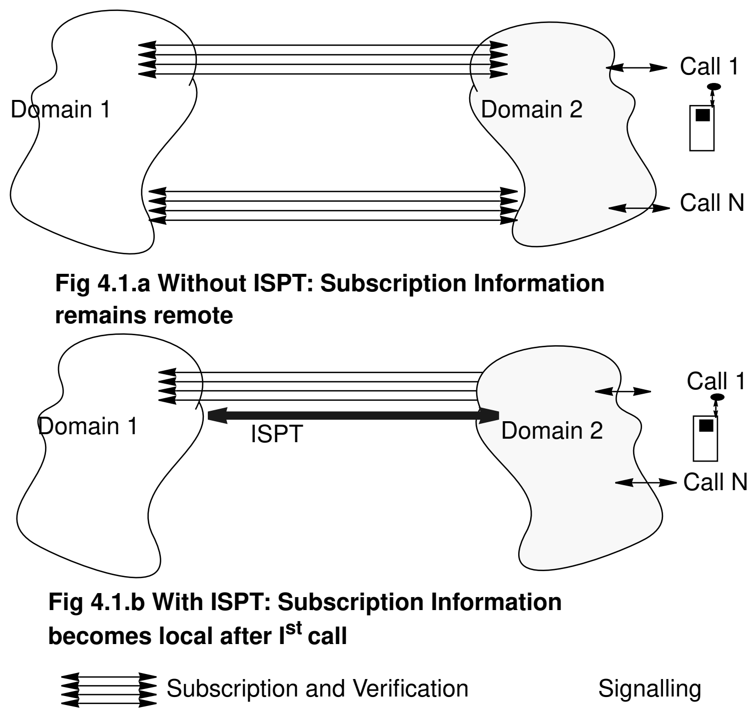


Figure 4.8 Signalling between Interworking Domains: with and without ISPT

ISPT is further illustrated in Figure 4.2, which shows an example in the following steps:

1. A mobile terminal moves from the home domain to the visited domain.
2. The terminal attempts registration at the visited domain.
3. The visited domain queries the home domain for authentication of terminal.
4. The home domain validates query.
5. The home domain and visited domain co-ordinate service profile transfer from

the home to visited domain.

6. The terminal receives registration approval.

7. The subscribed service can be accessed from the terminal.

The user view comprises steps 1,2, 6 & 7; the management view comprises of steps 2,3,4,5& 6. All related networks - the home network, the visited network, and intermediate visited networks - are coordinated and consistency is maintained.

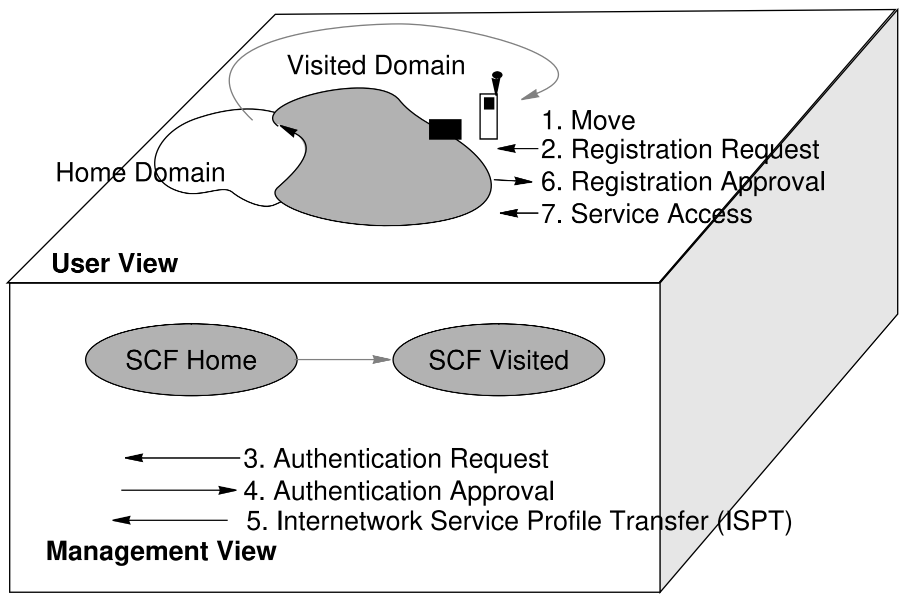


Figure 4.9 Interwork Service Profile Transfer

Current distributed computing technology can make service profile transfer feasible. A number of approaches have been proposed: moving text, moving byte code and moving contexts. Research on migratory applications [LC97] introduces a network computational model, where applications can migrate from one machine to another, take their contexts with them and continue where they left off. There are, however, two important factors specific to the domain of telecommunication services, that impact the understanding of a service profile and the mechanisms required to move it.

The first factor is the *distribution of service related information.* Compared to fixed telephone networks, mobile networks are composed of more numerous entities. Examples of such entities are user identification cards, mobile terminals, base station subsystems and mobile switching centers. These entities are administered by independent organizations. As a result, information for service provisioning, previously available under a single administrative domain is no longer accessible.

The second factor is *high-level design perspective of IN services*:

- Who uses a service? Is it one unique individual, or any one from a group; such as a family or business organization.
- To which points of the network should a service be associated? Should the service be associated with a fixed network location, a mobile terminal or a personal number.

Understanding these issues allows appropriate abstraction levels to be chosen for structuring the service. Such an investigation into these service concepts would have a two fold impact:

- a clarification of *what information* should be transferred in a service profile.
- *simplification* of service configuration and management.

The following sections discuss these aspects and summarize potential problems.

4.1.1 Distribution of Service Related Information

As mentioned, compared to the fixed telephone network, a mobile network has additional entities. Examples are user identification cards, mobile terminals, base station subsystems and mobile switching centers. These entities can be provided or managed by a number of independent organizations.

To exemplify distribution of service related data consider the personal mobility service. A personal mobility service could be purchased from one service provider. The access to the personal mobility service could be from a network access point belonging to another independent service provider. For successful execution of the personal mobility service, the two independent entities will have to co-operate. With such services, subscriber and service related information gets distributed network-wide.

A scenario with two mobile entities visiting a domain is illustrated in figure 4.3. Shown are two mobile networks and one fixed network. A *mobile terminal* T_1^1 from *mobile network* N_1 attempts registration at *mobile network* N_2. A *user* U_3^1 with a personal mobility subscription from the *fixed network* N_3, attempts to register on the *mobile terminal* T_1^1.

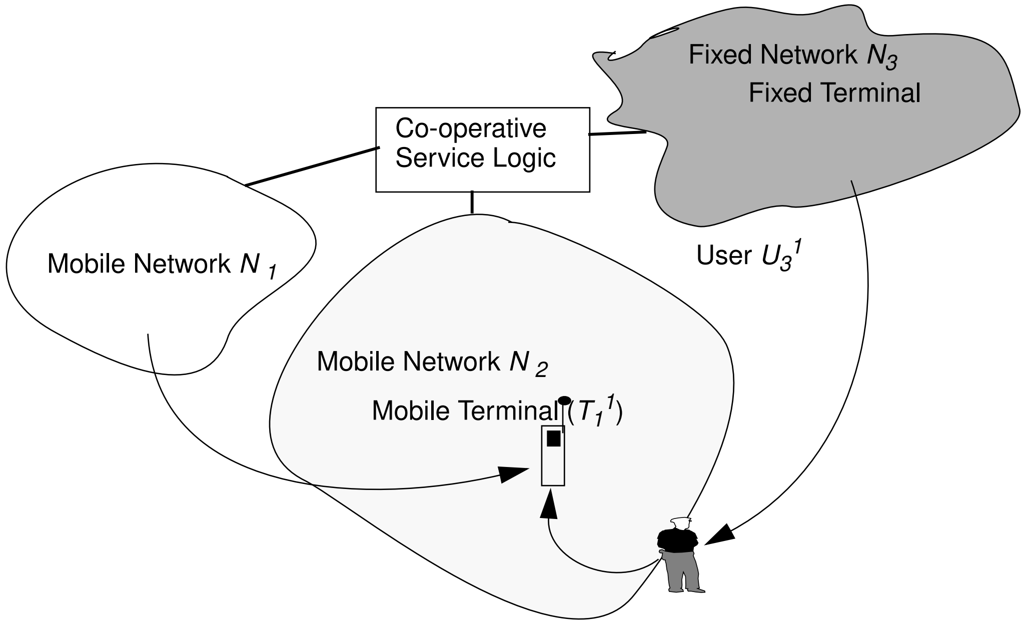


Figure 4.10 Mobile User and Mobile Terminal in a Multi-Domain Environment

Service provision would be through co-operative service logic developed between the 3 domains.

Evolution of mobile communications systems requires big investments and only a partial set of target capabilities can be provided in a short period of time. In order to provide global mobility and other services, it will be necessary for domains and service providers to co-operate. Management of the distribution of service related information is the key to successful delivery of mobile and co-operative IN services.

4.1.2 High Level Design Perspective of IN Services

Current View

An IN service has a purpose and value: to increase the functionality available above a basic call, where the basic call is viewed as a voice connection between two directory numbers. Nowadays, on a conventional fixed network in North America, this number is ten (10) digits long and is represented as *npa-nnn-xxxx* called North American Numbering Plan (NANP). Each number identifies a location on the physical network. This location is assumed connected to a terminal. The operating telephone company provides such a network connection to the subscriber's establishment. Hence the location, the terminal, the user and subscriber are associated with the same network location (Figure 4.11). There is a 1-1 (one is to one) association between the location, terminal and user.

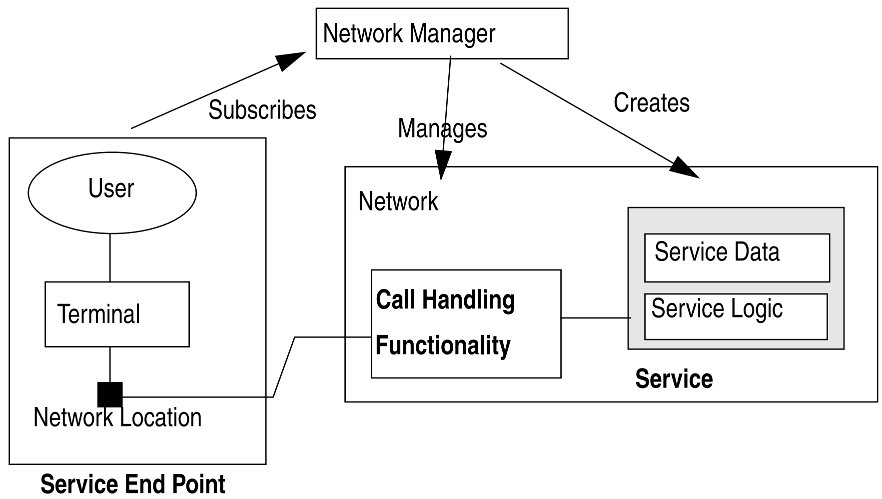


Figure 4.11 IN Service Existing Conceptual View. One Service End Point associated with location, terminal and user.

The approach of identifying the user, terminal and location with the same number works well on a fixed network. Implementing and managing services is uncomplicated since the location is the only point of service delivery. The services and data associated with this location is available at the local SCF and SDF.

With mobility however, this design perspective is inadequate and service behavior becomes ambiguous.

To illustrate such ambiguity, consider examples of the commonly used fixed network services: Terminating Call Screening and Call Forwarding.

In *Terminating Call Screening*, a user can block calls from a specified call screening number. On a fixed network, this number refers to specific location. In a mobile environment, what does this *specified call screening number* refer to: a personal mobility number, a mobile terminal number or a location? Should all calls from a specific user be blocked, or should all calls from a specific mobile terminal be blocked, or should calls from a specific network location be blocked.

In *Call Forwarding* the user can program the network to forward all calls to another *call forwarding number.* At the fixed network level the number refers to a location, and all calls to one number are forwarded to another number. In a mobile network, what does the *call forwarding number* refer to: a user, a terminal or a location?

As an example of ambiguity in call forwarding consider the following scenario. Mr. Manager manages Better Mousetraps, and has subscribed to call forwarding on his mobile terminal. Mr. Manager decides to leave his town and wishes to forward all calls to his secretary. Should all the calls be forwarded to the location that the secretary normally is at, or to the personal mobility number of the secretary. If the calls are forwarded to the personal mobility number, and the secretary takes a holiday, the calls will still be forwarded to the same number - clearly an undesirable situation.

Though mobile systems show a great increase in the complexity of service provision, this presents a unique opportunity to better understand service behavior in a mobile environment.

4.2 RM-ODP Models for Service Profile and Service Portability

In contrast to the approach described above, this thesis proposes that services are better understood by considering a user, a terminal and a network location as three types of *Service Delivery Points* (SDP).

A Service Delivery Point (SDP) is an entity on a network where a service may be delivered by a service provider and accessed by a user. A subscriber designates the SDPs at which specific services are to be provisioned. Service behavior is dependent on the SDP type. Service behavior and its relation to SDP type is further discussed in examples shown in Figure 4.12, Figure 4.13 and Figure 4.14. A SDP can be associated with more than one subscription. Relevant subscription information for a instance of a SDP is stored in its service profile.

Traditional IN service development is viewed as provision of additional functionality to the basic call. Management aspects such as location management and service profile modification though important have not been investigated. In contrast, we include service management and non-call related functionality as integral to the service. A service is a privilege offered by co-operation of all managerial entities of co-operating networks.

Consider the following three examples which justify this approach to modeling a service. This approach addresses recognizable human situations. The human situation acts as a guideline for choosing useful and predictable behavior. It is not necessary to involve the subscribers in the technical details of the service features, and yet services behave as subscribers expect, because both designers and subscribers are thinking in terms of the same human situations[PZ95].

Terminal as Service Delivery Point

Example 1

A car rental company wishes to equip all its cars with a mobile terminal. To provide additional value to the customers, the company offers local calls from the car-terminal. All long distance calls are blocked. However, to provide customer support, the long distance number to the company support line is free. This functionality of Call Screening, and free calls to the customer service is associated with the car-terminal. The terminal is associated with a service profile where this information is stored.

Example 2

A certain bank has a number of employee posts, two of which are an accounts manager and the loan manager. The bank provides the accounts and loan managers with mobile terminals. The duties of the accounts manager and loan manager are fulfilled by individuals from a pool of employees, and on any particular day, any two of pool registers on these terminals. The Call Forwarding on Busy service is used to efficiently distribute incoming calls. When a call is made to the accounts manager terminal, and a busy signal is encountered, the call is forwarded to the loan manager terminal. Similarly, when a call made to the loan manager terminal, and a busy signal is encountered, the call is forwarded to the accounts manager terminal. This scenario is shown in figure 4.5. The Call Forwarding on Busy service is associated with the terminal, and this information is stored in the terminal service profile. Minimal management and re-configuration needs to be done, as service behavior is structured around terminal behavior and is independent of the employee registered on the terminal.

A terminal SDP is uniquely identified by an International Mobile Equipment Identifier (IMEI).

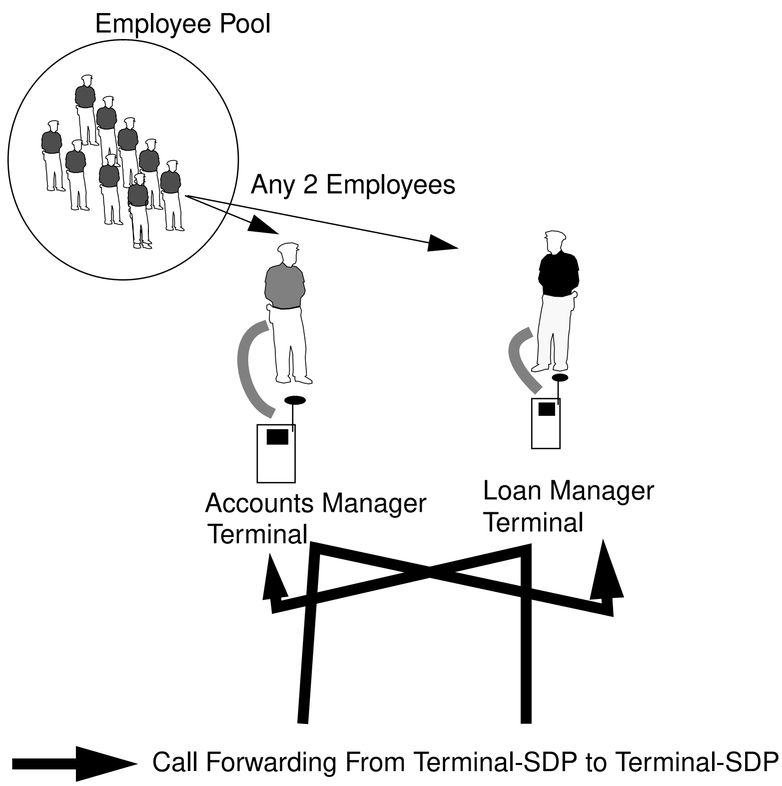


Figure 4.12 Terminal as Service Delivery Point

User as Service Delivery Point.

Consider an enhanced situation at the bank as described above. This scenario is shown in figure 4.6. Mr. A is an employee of the bank, and has services associated with his personal number. Mr. A subscribes to call forwarding on busy service. When he is busy in mid-call, the call should be transferred to his voice mail box.

Mr. A is currently registered on the accounts manager terminal. He is busy in mid-call, and an incoming call is made to his personal number. Since the call is to his personal number, it should not transfer to the accounts manager, but to his voice mail box.

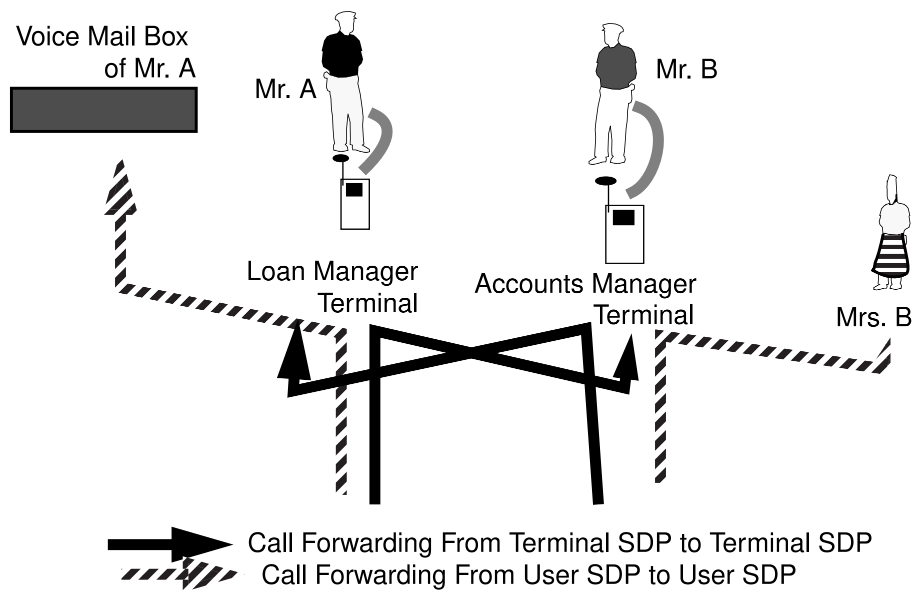


Figure 4.13 Terminals and Users as Service Delivery Points

Similarly, a call to Mr. B's personal number should be forwarded to Mrs. B and not to the loan manager terminal. Associated with the personal number is a service profile which contains this Call Forwarding information. When Mr. A or Mr. B register at a new domain or terminal this service profile is transferred accordingly.

This is an illustration of the clarity introduced with SDPs. In the above example, in spite of four entities, two calls and two services, the service behavior is clear and unambiguous. A User-SDP is uniquely identified by a International Mobile User Identity (IMUI).

Location as Service Delivery Point

Consider a high security defense establishment as shown in Figure 4.14. The telecommunications network supports full mobility, and has a pico cell C_1.

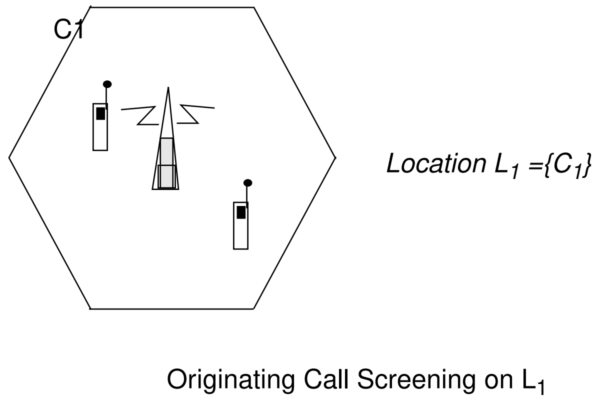


Figure 4.14 Location as Service Delivery Point

To enforce security, the defence establishment administrators wish to screen all outgoing calls. The Originating Call Screening service is associated with this location, i.e.: the defense establishment. This service behavior is consistent and understandable irrespective of terminal and user identities using the cell.

Associated with a location is service profile. The location could be a fixed network access point (npa-nnn-xxxx), a mobile system base transceivers station, or a logical location comprising of a set of fixed network access points and base stations. The generic term *location identifier* is used in this work to refer to a network entities associated to location. The use of the term location identifier will become clear in the ODP Enterprise Viewpoint model presented in Section 6.0.1.

The above scenarios clearly illustrate that service behavior is simplified when the location, terminal and user are each considered as types of SDPs. Such service provision presents flexible, hence adaptive mechanisms for enforcing differing ranges of service policies.

Complete call setup and service access can be seen in 2 sequences

1. Registration
   - 1.a Access to service profiles.
   - 1.b Resolution of constraints as specified by the obligations and permissions in the three types of service profiles.
2. Call Setup and Service Invocation

Regarding the point 1.b, it should be noted that significant interactions between service profiles may result in highly mobile environment. The services invoked will depend on the priority assigned to service profiles. The mechanisms of resolution of service profile conflicts and interactions is beyond the scope of this thesis.

In a mobile environment, the adoption of the IN model causes an increase in the number of entities, both manager and managed. Service creation and management mechanisms are critical for reliable and efficient service provisioning. However, these mechanisms have not yet been examined and developed in current IN standards. Efficient development of IN services needs additional concepts to aid understanding and simplify development. This thesis makes contributions in the area of service profiles, service delivery and their management. This discussion leads to the following preliminary conclusions:

- Three types of SDPs should exist: user, terminal and location.
- Service behavior is unambiguous and management and configuration easier when services are associated with SDPs.

The following section proposes an ODP specification model for service profiles

4.2.1 Specification of Service Profiles

This thesis proposes that from a high level user perspective, a call is placed to a role rather than to a type of SDP. The user has no concern of the properties or identity of the SDP. She is concerned only about the functionality the SDPs offer and the role they play. Depending on required service behavior a user, terminal or location can play the required role. A SDP is a network level abstractions as it has identifiers (IMUI, IMEI or Location Identifier) on the network level. The role these SDPs play is a set of properties which are important for the SDP to be able to behave in a certain way as expected by a set of other objects, i.e. the users of the service. [KO95].

Service Delivery Point Properties Vs. Service Role Properties.

SDP properties are *intrinsic and independent* of the service perspective. A user is identified by an IMUI, a terminal is identified by a IMEI, and a location is identified by a location identifier. Service role properties are *extrinsic and dependent* on service behavior. A role property cannot exist independent of the SDP; and is imposed from the outside by the service designer. Role properties may be temporary and are given for the length of service activity. To be useful, role properties must be accepted and recognized by the clients of the SDP, i.e. the users of the service.

To illustrate the above consider the following example. A service designer wishes to design call forwarding for a bank. The call forwarding is from one role to another: an accounts manager's calls are forwarded to the associate account manager. The service designer imposes these roles on the SDPs. At the network level the two entities are of different types: the accounts manager uses a mobile terminal, while the associate accounts manager uses a fixed terminal. To the users of the service the network level entities are abstracted.

In fixed and mobile networks the following SDPs and their corresponding intrinsic properties are identified.

Table 1: Intrinsic Properties of SDPs

Fixed Network

Table 1: Intrinsic Properties of SDPs

Service Delivery Point (SDP)	*Intrinsic Property*
Location (Physical Network)	Directory Number.
Mobile Network	
Service Delivery Point (SDP)	*Intrinsic Property*
Location (Physical Network)	Base transceivers (BTS) Address.
Mobile Terminal	IMEI
Person	IMUI

The above discussion explains the crucial difference in network level abstraction, i.e. the SDP; and service design abstraction, i.e. the role the SDP plays in the service behavior. This leads to a *functional* definition of the service profile.

A Service Profile relates SDPs to Service Roles.

A service profile is an object that integrates SDPs and the service roles.

The service profile is a mobile object. It can be moved to a new domain (ISPT). An example of ISPT is illustrated in Figure 4.9.

Since SDP can play more than one role, i.e, it is possible to have more than one subscription associated with it. An example of such a situation would be a manager at a bank who has office and home service subscriptions associates with her mobile terminal.

Therefore:, this can be more formally expressed as:

If *SDPID* is a *SDP Identifier*,
and *SS* is a *Subscription Set*,
then the *Service Profile* SP_i of SDP_i is:

$SP_i=<SDPIDi,SSi>$

The *Subscription Set (SS)* is composed of independent subscriptions SUB_i. Each subscription SUB_i is composed of a *role name (RN)* and an associated *service privilege list (SPL).*

$SS_i=<RN,SPL>$

An alternative tabular representation of a service profile is:

Table 2: Service Profile Tabular Representation

<table>
<tr><td rowspan="5">SDPID</td><td colspan="6">SS</td></tr>
<tr><td colspan="2">SUB $_1$</td><td colspan="2">SUB $_2$</td><td colspan="2">SUB $_n$</td></tr>
<tr><td rowspan="3">Role</td><td>Service $_1$</td><td rowspan="3">Role</td><td>Service $_3$</td><td rowspan="3">Role</td><td>Service $_5$</td></tr>
<tr><td>Service $_2$</td><td>Service $_8$</td><td>Service $_4$</td></tr>
<tr><td>Service$_6$</td><td></td><td></td></tr>
</table>

This above table and description is a template from which a service profile object is generated.

Access and Usage Rules for Service Profiles

The objects representing the service profiles are stored in the trader. A service component in a local or remote domain would place a request to the trader asking for a service of type “service profile”, with specific service properties. The service properties used can be any combination of the elements of the tabular representation of the service profile shown in table 2. In this manner a flexible mechanism for accessing service profiles can be used.

4.3 Conclusion

Service portability is critical in mobile systems. If a user is to interact differently with visited networks, the need of adapting will inconvenience users and the penetration of global mobility services will be limited. The requirements for similar “look and feel” are especially valid in telephone systems as the user-network interface is a numeric keypad.

This chapter illustrated the ambiguity introduced in service behavior in a mobile environment. The problems associated with distribution of service related information and existing design perspectives were shown. The first contribution was to enhance Service portability perspectives by introducing the Service Delivery Point concept. Using SDPs to structure services, behavior is clear and unambiguous. The SDP approach is supplemented with a definition and explanation of service profiles. This clarified *what information* should be transferred during ISPT.

A second contribution was in IN service management. Due to focus on lower level protocol development SMF functionality has largely been ignored in IN standardization. The template provided for service profiles should enable interworking of management entities. Two primary functionalities of the SMF should be

- identification of SDPs relevant to service behavior.
- creation of associated service profiles.

These concepts are elaborated in chapter 5 and 6 to illustrate terminal mobility, personal mobility and service portability in a mobile virtual network service.

Chapter 5
Modeling Mobile Global Virtual Network Services in the RM-ODP Framework

5.0 Introduction

Evolutions from existing telecommunication systems towards third generation mobile systems - IMT 2000 and UMTS - requires a big investment and only a partial set of third generation capabilities will be able to be provided in the short to medium term. In order to minimize investment and provide services as soon as possible, it will be attractive for operators of different comprehensive systems to form alliances.To meet co-operative and mobile service requirements, interworking aspects between different network operators are to be addressed. Interworking will allow exchange of service control and service management information between independent IN networks for the provision of required services.

In this chapter, interworking issues between IN domains are addressed. The Global Virtual Network Service (GVNS) is used as a case study. The GVNS allows service providers to offer subscribers a service with features and functionality similar to that of a private network, minimizing the need for dedicated network resources. This service is provided across domains and administrative boundaries which are transparent to the user. A GVNS customer is assigned a unique identifier which identifies the GVNS globally. The GVNS configuration is defined by the customer.The GVNS provides a customer with global services as a result of interworking among GVNS participating service providers.

This service is chosen as a case study for two reasons:

- modeling the GVNS has traditionally been difficult as the IN capability sets have not focussed on interworking [FG95].
- GVNS is a generic service, and illustration of successful modeling using ODP concepts can be extended to other services like UPT, PCS interoperability[HM96] and FPLMTS/UMTS [LC96][TM96]

5.1 Introduction to Integrated Modeling of IN and ODP

In this thesis, the coordinating framework used to develop the Global Virtual Network Service (GVNS) service is RM-ODP. The RM-ODP framework is used to create an integrated architecture for the distribution, interworking and interoperability of IN services.

The real problem to be faced is not the number of requirements, as they are often very scarce and imprecise, but the fact that they belong to extremely different categories.

This study will attempt to show that the ODP viewpoint approach is useful in handling such diversity in telecommunication service requirements and specifications.

In order to be useful in the IN context, RM-ODP must account for conceptual differences in the IN model. The primary conceptual differences are abstraction mechanisms. The INCM defines four planes, where the lower planes show greater distribution and implementation detail. These planes are viewed as design process refinement steps [CH94] as the INCM prescribes a mapping from one plane to the next. To minimize complexity, RM-ODP uses two abstraction mechanisms: viewpoints and distribution transparency. Multiple viewpoints are a cornerstone of the ODP model, they enable a different perspective of a system to be presented to different observers.Viewpoints are developed through a selected set of structuring rules and architectural concepts [X.901]. Viewpoints are not design process refinement steps [AH91] but may be viewed as such. Each viewpoint is a partial view of a complete system specification. It is through this separation of concerns that the inherent complexity of a complete distributed system is decomposed.

However the subdivision of the system raises the issue of consistency. The descriptions of the same or related entities will appear in different viewpoints and it must be shown that multiple specifications are not in conflict with one another. What is an appropriate definition for consistency? The RM-ODP is ambiguous on this aspect [BDS95]. During the development process there must be some way to combine specifications from different viewpoints into a single implementation specification. Three specification to specification transformations, i.e, translation, refinement and unification have been investigated in [BDS95]. Translation maps specifications to new languages, refinement has the usual meaning and unification is a transformation which enables specifications of the same language to be combined. Significant work is being carried out on the process of viewpoint consistency. However these approaches currently are exploratory and do not prescribe a formal methodology. Whilst it is accepted that the viewpoint model greatly simplifies the development of systems specifications, the practicalities of how to make this approach work are only beginning to be explored [BdS95].

Hence this work uses viewpoints to develop the GVNS through viewpoints, but does not investigate the formal unification and consistency process. Instead this study uses general heuristics as prescribed in [X.900] [KF93] to guide development.

Prior to the actual modeling of the GVNS service the next sections 5.1.1 and 5.1.2 discusses the integration of ODP and IN in modeling abstraction and co-operation.

5.1.1 Modeling Abstraction

To make service development manageable and understandable, abstraction must be carefully applied to the GVNS. Not every aspect of the service has to be agreed on, since the need for understanding is more important than the need for representing all details.

In the search for precision, service analysis can take place at the wrong abstraction level - at the level of the network implementation rather than at the level of telecommunication service requirements. As a result, often a telecommunications service is ambiguous and service behavior does not lead to agreement between the customer and the service designer. Hence understanding the service requires the right level of abstraction and correct implementation demands a precise and detailed specification. An important aspect of abstraction is encapsulation, which means that information is understood not by knowing its detailed structure but by what it can do. Once the "what" of a particular abstraction level is specified, it may be implemented. "How" it is implemented depends on the capabilities of the next available level. A service specification can be refined into more detailed specifications. At each level a specification is constructed using building blocks appropriate for that level and glue appropriate for that level.

In the IN model the global functional plane (GFP) models an IN structured network as a single entity (Figure 5.0). Contained in this view is the global Basic Call Process (BCP) and Service Independent Building Blocks (SIBs). The immediate lower abstraction level is the distributed functional plane (DFP) which models the distributed view of the IN structured network. SIBs are realized in the DFP as actions performed by functional entities (FEs). The informations flows (IFs) consist of messages which enhance information flows between FEs. The physical plane represents the actual implementation of the functional entities specified in the GFP.

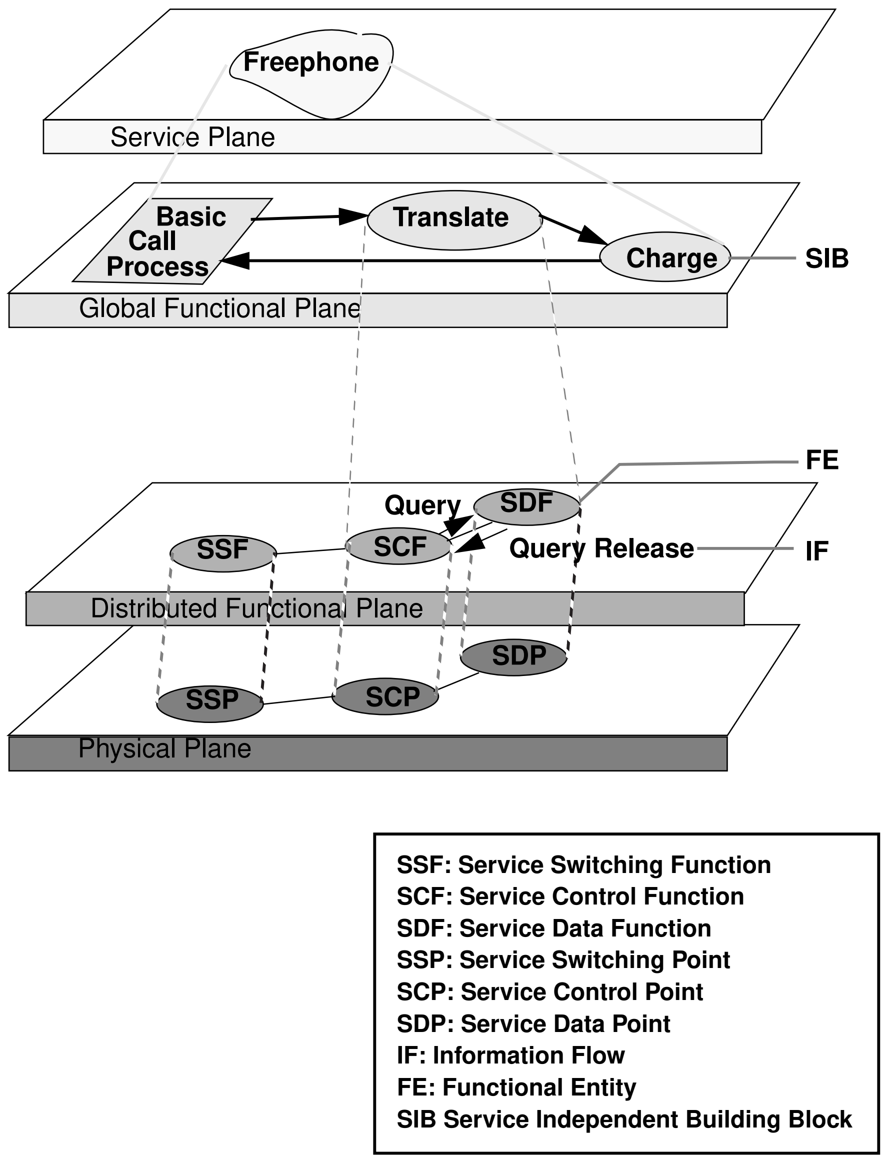


Figure 5.0 The Intelligent Network Four Plane Model

Two deficiencies are noticed in the IN service development approach. The service plane specifications are normally written in prose form. This may be adequate for simple service, but falls short in describing complex services that require interworking and co-operation between business entities. A suitable abstraction mechanism for specification of enterprise level concerns would be of considerable help. The second deficiency is the gap between the GFP and the DFP. The jump in abstraction levels from GFP to DFP is significant. At the GFP the network is considered global, and at the DFP the logical entities, information flows and protocols are visible.

To address the first deficiency concerning the limitations of Service Plane descriptions, the ODP enterprise model abstractions are used.The ODP enterprise modeling recommendations help supplement the service plane specifications using well understood enterprise level abstractions and representations. Since the ODP enterprise view point reflects co-operation and distribution of the enterprise, it contains more implementation behavior specific information than the service plane. Thus the ODP enterprise viewpoint is introduced below the service plane.

The second deficiency, i.e, the gap between the GFP and the DFP, is addressed by using the ODP information viewpoint. This viewpoint allows a smoother transition between the GFP and the DFP: appropriate information is retained and visibility of key details is maintained. The information model is proposed below the Global Function Plane. The reason for this is that the GFP treats the network as a uniform single entity with unified access to global resources. In a distributed environment this is not the case. The information model represents the details of information flows and structure the illustrate distribution. As it contains more implementation specific information, it is modeled below the GFP. These plane are shown in Figure 5.1.

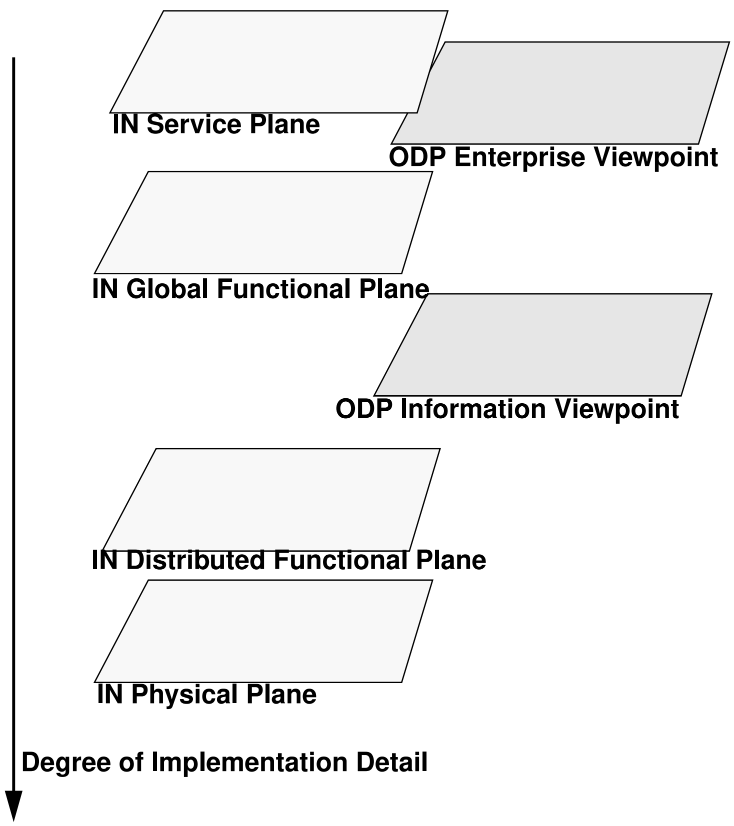


Figure 5.1 Comparing IN and ODP Abstractions -I

5.1.2 Modeling Co-operation

A closer examination of IN shows that the IN model prescribes a hierarchy for service management and provisioning. The current IN CS-1 standard is based on the half call concept where a service is invoked at the originating or terminating end of the call. An implicit assumption in CS-1 is that the SCF and SMF work independently and do not need to interwork with other SCFs and SMFs. Hence, there are no defined information flows that enable exchange of information at the management (SMF-SMF) and control (SCF-SCF) level [Q.1219].

The consequence of this approach is that services cannot span domains. This is a limitation for the development of interworked and distributed services. For such services IN domain administrators should be able to exchange data and share functionality and simultaneously must be able to apply their own management policies and make design decisions for the systems they own. It is unreasonable to assume sufficiency of a hierarchical service provisioning - as in IN - in which some domains gives management authority of themselves to others.

To overcome this glaring limitation, there must be gateways between IN domains that will enforce the security and accounting policies of each domain and oversee interactions in between. This work uses the RM-ODP trader, a commonly accepted standard, as such a gateway. This is in contrast to proposals for Interworking Service Control Points (ISCPs) introduced in [TE96] [GHY96] An interworking SCP is a special SCP that controls two or more domains. In an ISCP model, both domains lose service management and administrative control to the ISCP. This is an undesirable configuration, as domain operators may not agree to hand-over of control, administration and security.

This thesis proposes a solution by which the ODP trading functionality elegantly supplements existing IN models. Compared to interworking proposals [TE96][GHJ 96], this thesis model is more complete, and allows building of extensible and dynamic services and mechanisms for configuration of information flows through policy specification. This study uses ODP trading as:

- a mechanism to interwork IN domains through the concepts of trader interworking
- configuration of information flows between domains through specifying trading policies.

The trader approach allows building the IN -network as an extensible platform. Instead of building everything into a monolithic telecommunications service, a core system - the ODP trader- with certain built in potential for extension is used rather than an arbitrary rich service. ODP prescribes separation of policy, “what” from mechanism “how”. The advantages of such an architecture is the fact that once the mechanism is implemented, new service configurations can be introduced by specifying a new policy.

The modeling proceeds as follows:

Customer requirements are specified in the ODP Enterprise Viewpoint. The enterprise viewpoint reflects the objectives more precisely and extensively than the IN service plane. Hence the service plane model is not presented. As the target service is distributed, a few distribution details are introduced at the enterprise viewpoint. The enterprise specification defines the purpose, scope and policies of the GVNS. A similar specification has been successfully used in the modeling of personal communication in [BH94]. Since the extent of service provision is dependent on technology, some technology aspects are discussed.

The GFP in the INCM treats the IN structured network as a global entity, thereby masking all distribution aspects. Since the Global Virtual Network Service (GVNS) is inherently distributed, modeling the GFP would serve no purpose. For this reason the GFP model of GVNS is not developed.

Subsequent to the enterprise specification, the ODP information viewpoint is developed. The information language captures the common understanding shared by the components which make up the distributed system. The information viewpoint shows the invariant static and dynamic aspects of the system. ODP prescribes [X.900] a declarative and contractual approach to model information viewpoint concerns. The data part of LOTOS or Z are ideally suited for declarative specifications. In order to maintain manageable scope, rather than formal development of information concerns we use less rigorous, but more intuitive and understandable object modeling techniques. The GVNS information viewpoint developed:

- Shows the structure of the system using object diagrams
- Shows the information exchange between domains through a trading graph. A node on the trading graph represents a trader, and the edge represents a link to another trader. The nodes and edges form the trader interworking graph, which govern the propagation of a request for service logic or service data between interworked IN domains (import operation) through the offer spaces of the connected nodes. Associated with an edge are properties that describe the edge and the target node.
- Proposes an interworking model called the "Interworking State Model". This model allows distributed services to be developed.
- Shows the information exchange between objects through event diagrams

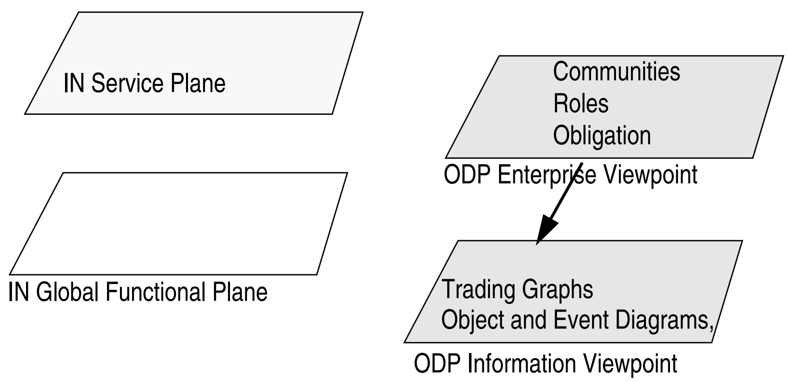


Figure 5.2 Comparing IN and ODP Abstractions -II

5.2 GVNS Service Procedures and information flows based on Current IN Procedures

This section is a discussion of GVNS service procedures and information flows based on existing CS-1 standards. Call setup procedures using existing standards and assumptions are examined. The procedures, functional entities and information flows described in this section relate to service provision across multiple networks, by allowing the data to move across a network boundary. The section concludes by highlighting problems with the current approach.

Call processing Mechanisms

A SIB entity on the GFP represents a standard network wide reusable capability residing on the GFP. The SIBs are mapped onto the DFP and certain processes or techniques are prescribed in order to achieve their functionalities. Such techniques are called call processing mechanisms and existing IN standards [Q.85] discuss three types of call processing mechanisms called Types A, B & C. The choice of processing mechanism implementation is not specified in the standards [Q.12xx] and depends on the network operator.

Type A: Customer specific information is stored in the originating network.

In an implementation of a GVNS, this implies that *all* domains must have all information about all on-net SDPs. The databases are duplicated, and no real-time co-operation is needed between networks after the databases are initialized. This approach has the following deficiencies:

Scalability: In case the number of participating service providers and the number of on-net SDPs is large, the storage requirements become unacceptably high. Assume that a GVNS is to be provided over 3 domains with 100 on-net SDPs. The database of each domain would store information about 100 on-net SDPs, with total number of allocation for 300 entries. If the GVNS is to be provided over 5 domains with 1000 on-net SDPs, each domain would store 1000 locations and total allocation would be for 5000 entries.

Management: Consider a GVNS over 5 domains with 1000 on-net SDPs. If the service profile of one of these locations is changed, all 5 domain would have to be updated. Keeping all 5 domains consistent and updated, in an efficient and effective manner, is difficult.

Type B: Customer specific information is stored in both originating and terminating networks but without direct interaction.

In implementation of a GVNS service this implies that each domain maintains information of is local on-net SDPs. When required SDP information is not found in a domain, that domain forwards the call to another domain. This approach has the following disadvantages:

unstructured: When a specific domain does not possess required information of an on-net SDP it forwards the call to another domain. This approach contradicts the principles of IN. Interworking remains at a routing level and many of the advantages of IN concepts are lost. There is no negotiation between the IN SCFs, but rather one SCF relinquishes control to another SCF.

Type C: Customer specific information is stored in both originating and terminating networks with direct interaction.

In the implementation of the GVNS service each domain would have to maintain data of local on-net SDPs. When a domain needs information of other SDPs, it directly accesses the SDF of other interworking domains through a direct SDF-SDF link. This approach has the following disadvantages

Security: type C call mechanisms allow unrestricted data access. A co-operating domain should have access to subscriber and service data only of those on-net SDPs that are served under the GVNS. It should not access data of other SDPs in collaborating domains. One of the aims of this work is to provide an adequate access control mechanism.

This call setup procedure is quite unsuitable as it allows unrestricted access to a collaborating service provider's data. Management and control data influence service processing and billing, the messages to exchange these data are very sensitive and security is a critical aspect in this domain. In addition, some national regulatory bodies might require protection of privacy of service subscribers.

Excessive Transparency.

The type C interworking configuration is through a SDF-SDF relationship. This interworking is location transparent to the SCF. This transparency is a limitation since SCF may need to invoke service logic on the initiation of SDF-SDF interworking.

It is evident then, that current interworking mechanisms do not satisfy minimum requirements. Of the three call processing mechanism types A,B &C, interworking requirements are not met. Types A and B are difficulty to manage and are not in alignment with two aspects of intelligent network philosophy: separate switching from data; and minimization of data duplication. Type C has security problems.

This section discussed the integration of IN and ODP abstraction mechanisms. The limitation of current IN call processing mechanisms were shown. The next section presents the ODP model for a GVNS.

Chapter 6
The GVNS Model in the RM-ODP Framework

This chapter presents the application of RM-ODP enterprise and information viewpoints to the Intelligent Network (IN) Global Virtual Network Service (GVNS).

Service provisioning across heterogeneous architectures is a combination of interworking at two levels:

- Switching and routing level
- Service management and control level.

In this thesis the emphasis is interworking between service control and management entities. It is assumed that switching environments are uniform and do not mitigate against support for interworking.

A complete ODP specification would be beyond the scope of this work, therefore the focus is restricted to the Enterprise and Information Viewpoints (See sections 3.1.1 & 3.1.2). The ODP enterprise model, with its high level abstractions, presents the goals and the objectives of the service. The enterprise model is presented in section 6.0. The ODP information model is developed subsequently in section 6.1. This corresponds to requirements analysis and functional specification in software engineering process as shown in Figure 3.1.

6.0 The GVNS ODP Enterprise Model

According to RM-ODP an enterprise specification defines the purpose, scope and policies of the ODP system in terms of the each of the following items:

- one or more enterprise objects and the roles in which these objects are involved.
- identification of communities, domains and federations
- activities undertaken by the system
- policy statements about the system

In the enterprise viewpoint, specifications are restricted to a basic set of concepts and rules addressing the scope of the GVNS. It is important to put the service features in the right context, i.e., to identify the on-line (real time) and off-line (non-real time, contractual) relationships and relate services to domain interfaces. The enterprise viewpoint relates entities to their roles. The same entity may act in different roles, simultaneously or at different times. This distinction is often referred to as a distinction between actors and roles, whereby a certain role can be played by one or several actors and a certain actor can play one or several roles simultaneously.

The enterprise specification is presented in two sections. The first, section 6.0.1, comprises of generic ODP enterprise definitions and descriptions. The definitions and descriptions of this section are generic and can be reused for other services. Inorder to further clarify, a specific case study is taken up in section 6.0.2.

6.0.1 GVNS Generic ODP Enterprise Definitions

Global Virtual Network Service (GVNS)

GVNS offers to end-users a service with features and functionality similar to that of a private network, minimizing the need for dedicated network resources. This service is provided over administrative and technological boundaries, in a way that these boundaries are transparent to the user.The GVNS provides a customer with global services as a result of interworking among GVNS participating service providers in various domains. A GVNS customer is assigned a unique identifier which identifies the GVNS globally. The GVNS configuration may be defined by the customer.

GVNS Participating Domain

The interworking between multiple domains creates the GVNS offering. Each GVNS participating domain provides customer access to the GVNS. Each of these domains is called a GVNS participating domain. A participating domain can play different roles depending on the relationship it has with SDPs and other domains. This is reflected in terminology used in this chapter. A home domain is a domain where a SDP is originally is identified. A visited domain is a domain where an SDP roams to from another domain. An interworked domain is a domain that has an interworking relationship with another domain. An external domain is a domain that does not have interworking relationships to domains interworking to provide a distributed services.

GVNS customer

A GVNS customer is that entity which purchases services. The GVNS customer serves as an interface between the GVNS domains and the GVNS users. The GVNS customer can define aspects of the service: for example, the on-net SDPs, their numbering plan and associated calling privileges. An example of a GVNS customer would be the telecommunications manager of a corporation.

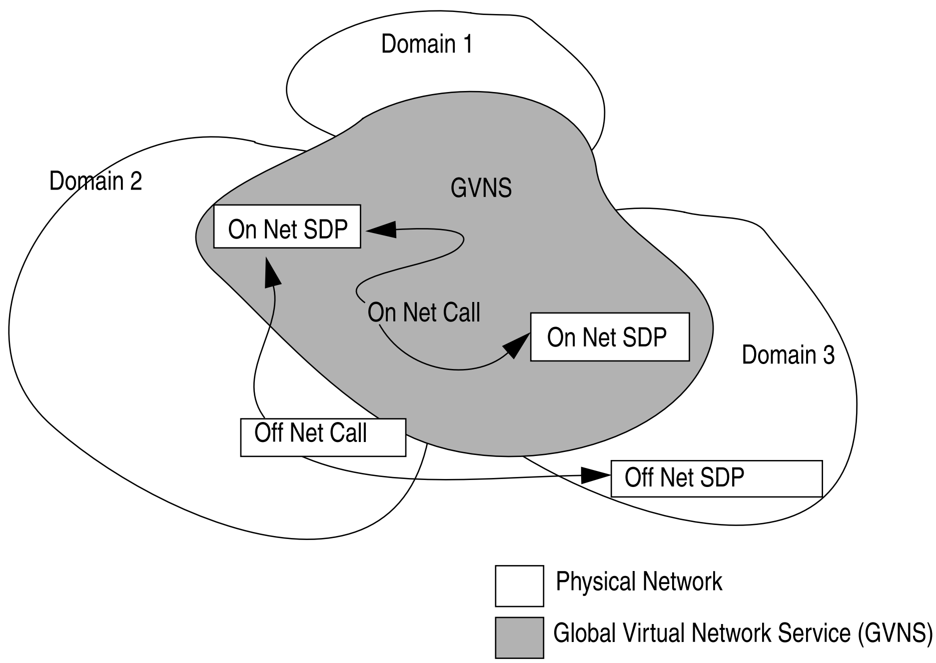


Figure 6.3 GVNS ODP Enterprise Viewpoint

On-Net Service Delivery Points (on-net SDPs)

These are entities that are logically defined by the GVNS customer to be part of its virtual network. The on-net SDP could be a person, identified by a International Mobile User Identifier (IMUI) or a terminal, identified by the International Mobile Equipment Identifier (IMEI) or a location identifier. (SDP types are discussed in Section 4.2.) GVNS on-net SDP may be mobile and move from one domain to another. A variety of routing privileges may be assigned to GVNS calls. GVNS users can make calls to other on-net or off-net SDPs.

Off-Net Service Delivery Points (off-net SDPs)

These are SDPs that are not defined by the GVNS customer to be part of GVNS.

GVNS user

A person accessing the GVNS is the GVNS user. Any person using a personal number, or a terminal or a location defined as an on-net SDP automatically accesses the GVNS, and hence is a GVNS user.

GVNS Calls

GVNS calls are from on-net SDPs. The calls may terminate to on-net or off-net SDPs. A call is a temporary relationship between two users, one which assumes the roles of caller, the other of callee. A call is not an enterprise object, because it does not play a role and is not a resource at this level of abstraction.

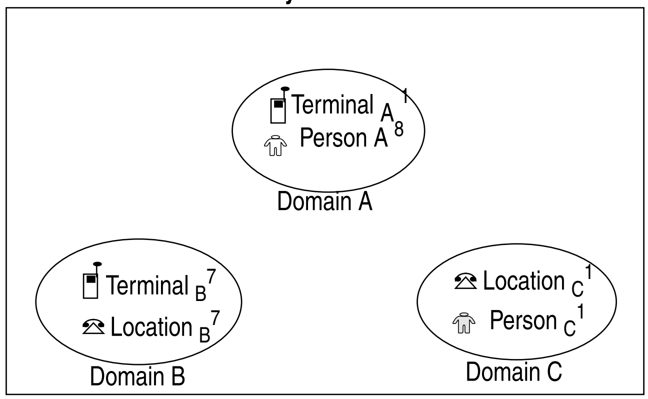
GVNS Customer Defined Numbering Plan (CDNP)

A GVNS customer has a numbering plan which associates numbers with on-net

SDPs. Such a numbering plan is called a Customer Defined Numbering Plan (CDNP). The number of digits sent and received will be at the customer's discretion, within a given range, specified by each GVNS participating service provider. The customer defined numbering plan and its relation to the physical and logical view of the network is shown in Figure 6.4.

Figure 6.4(a) shows the physical view of the network. This view consists of domains A,B & C, and SDPs (Terminal_A^1, Person_A^8, Terminal_B^7, Location_B^7, Location_C^1, Person_C^1). The CDNP, Figure 6.4(b), shows the addresses of the SDPs. (The CDNP addresses (201...206) refer to this figure only, and bear no relation to later sections) Figure 6.4 (c), represents the logical view of the GVNS. It is to be noted in Figure 6.4(c) that the SDP type and SDP location are transparent to the user.

GVNS: Physical Network View



(a)

Customer Defined Numbering Plan	
On-Net SDP	Address
Terminal $_A^1$	201
Person A ^{8}	205
Terminal $_B^7$	202
Location $_B^7$	206
Location $_C^1$	203
Person $_C^1$	204

(b)

(c)

GVNS Logical View = {201,202,203,204,205,206,206,}

GVNS: Logical View

Figure 6.4 (a,b,c) The Customer Defined Numbering Plan: mapping GVNS Physical and Logical Views (Numbering Plan refers to this figure only and is used as an example).

GVNS User Access

Dependent upon the GVNS participating service provider capabilities, the customer can choose various access procedures.

Direct access

This type of access allows the user to reach the GVNS without providing any authorization code. If the call is made from an on-net SDP, the network automatically verifies against the users service profile, to determine if GVNS and associated service access is allowed. On successful verification of the SDP, the user may access required GVNS services.

Indirect access

This type of access allows the GVNS user to reach its GVNS by providing a service access code. To use the GVNS, an authorization code must be manually entered irrespective of the SDP type. The network analyses this code and verifies if GVNS privileges are to be offered.

GVNS features description

The GVNS customer can define call screening to determine what type of calls are allowed from on-net SDPs. This may include, but is not limited to, restrictions such as, allow on-net calls only, or allow both on-net and off-net calls. Call screening is also a mechanism which may be used by the GVNS customer to restrict calling privileges.

GVNS Standard announcements

General and network announcements are to be provided with appropriate for GVNS calls. Standard announcements are to be provided in all participating domains.

GVNS Activation and Deactivation

Activation, deactivation and registration of the GVNS may be performed by the GVNS participating providers by arrangement with the customer. The GVNS participating service providers and customer can change the information regarding the GVNS, by following appropriate procedures.

This section has defined fundamental enterprise level entities and procedures and operations. The following section is dedicated to enterprise description of a GVNS example.

6.0.2 Enterprise Descriptions for the BetterMouseTrap GVNS Case study.

The concepts in this thesis are illustrated through the use of a specific service example. In this way, the proposed model can be closely examined and presented with greater clarity. This example of the GVNS is called the BetterMouseTraps GVNS.

The BeterMouseTraps GVNS is provided by 3 domains. Also, the GVNS supports mobility: some users are mobile and access the GVNS from all three types of SDPs spanning 3 domains.

GVNS Participating Domains

The 3 domains, shown as sectors of the circle in Figure 6.5, have similar conceptual models which can be mapped to IN functionalities, but have different technological implementations. They possess the following characteristics:

- Domain 1 (D_1): is a IN-structured PSTN. Access to the network is through fixed terminals.

- Domain 2 (D_2): is 2nd generation analog mobile network. Access to the network is through mobile terminals.
- Domain 3 (D_2): is a 3rd generation digital mobile network. Access to the network is through mobile terminals.

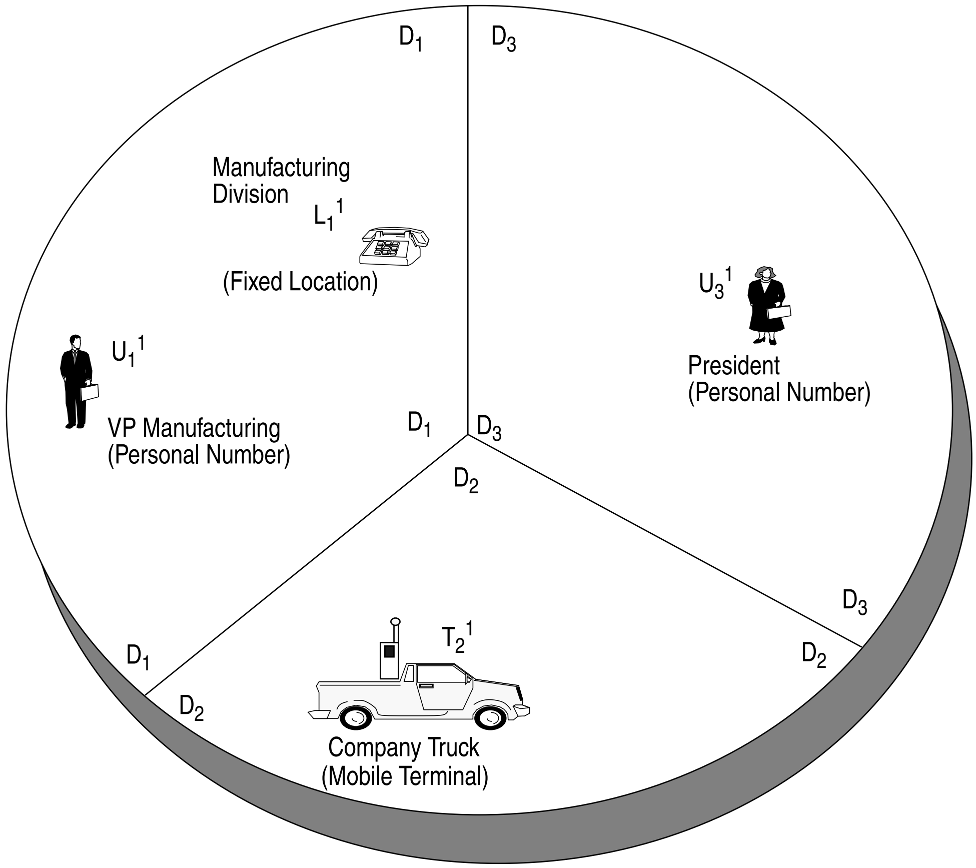


Figure 6.5 The BetterMouseTraps GVNS: Enterprise Model.

GVNS Customer

The customer of the GVNS is the president of BetterMouseTraps. The home domain of the subscriber is D_3. In addition to being GVNS customer, the president is also defined as an on-net SDP.

GVNS On-net SDPs

The 3 domains co-operate to provide a GVNS service.

SDPs are referred by $\text{SDPType}_{\text{Domain Number}}{}^{\text{SDP Number}}$.

SDP types are User (U) Terminal (T) and Location (L).

Hence in Domain 2 the first SDP of type terminal is referred to as T_2^1. The Better

Mouse Traps Inc. GVNS consists of the following SDPs:

- *A company president.* The president's home domain is D_3, and president has a personal number: U_3^1. She should be able to register and access services from any terminal from any of the 3 domains spanning the GVNS. This is an example of personal mobility.
- *A manufacturing division.* It is located in D_1. The manufacturing division has only one phone terminal, a fixed terminal. This access is identified by a unique location identifier: L_1^1. This is equivalent to a fixed (traditional wireline) telephone system.
- *A delivery truck* that moves between domains D_2 and D_3. A terminal, issued by domain 2, is fixed on the truck. This mobile terminal is identified by a unique terminal number: T_2^1. Network access for this terminal is wireless. This is an example of terminal mobility.
- *A vice-president of manufacturing.* The home domain is D_1. The vice-president has personal number U_1^1 and should be able to register and access services from any terminal on any domain part of the GVNS. This is a case of personal mobility

GVNS Auxiliary Services

In addition to basic call functionality, Better Mouse Traps Inc. needs

- a customer defined numbering plan for all on-net locations.
- a voice messaging system provided by D_2 accessible from anywhere on the GVNS.

GVNS Customer Defined Numbering Plan (CDNP)

The GVNS subscriber, i.e. the President, defines a numbering plan for on-net SDPs. The CDNP associates on-net SDPs with numbers.:

Table 3: Better Mouse Traps CDNP

On-net SDP	Number
President. U_3^1	11
VP Manufacturing. U_1^1	14
Mobile terminal on Company truck. T_2^1	16
Location in Manufacturing Division. L_1^1	18
Voice Message System VMS_2^1	20

GVNS Access

The BetterMouseTraps GVNS allows direct access to its users. They do not need to enter an authorization code.

Call Setup

To establish a call (from an on-net SDP), a user dials the number of the required on-net SDP as defined in the customer defined numbering plan (CDNP) shown in Table 3. The user does not need to know the physical location of the called SDP. The GVNS establishes the call independent of the current location of the called SDP.

The BetterMouseTraps GVNS: The technological perspective

Numerous architectures have been implemented over the recent past: traditional wireline telephone systems, 2^{nd} Generation Cellular Systems, Personal Communication Systems (PCS), Cellular Digital Packet Data (CDPD) systems. To demonstrate the strength of this work in the interworking of existing architectures, a technological perspective is added. This scenario is very likely with current technology implementations. Such a shown in Figure 6.6., the BetterMouseTraps GVNS has three co-operating domains:

- D_1: an IN structured PSTN. Access and interworking of the PSTN is through an enhanced SCF. This enhanced SCF is discussed in subsequent sections of this chapter.
- D_2: an AMPS network. The AMPS interworking intelligence is stored in the Home Locator Register (HLR) and Service Management System (SMS).
- D_3: is a PCS 1900 mobile network. Access and Interworking of the PCS-1900 is through a Operations and management centre (OMC) and Home Locator Register (HLR).

Detailed discussion of these networks is not relevant here. Each architecture defines functional entities for management and interworking. These domains are considered IN structured as functionalities can be mapped onto IN functionalities. Again, it is important to highlight that service management and control mechanisms to exchange information between such networks is unclear. Current IN standards do not discuss this issue [Q.12xx].

Provision of interworking mechanisms would find application in interworking of mobile communications networks systems, the target of much current research [HM96] [LC96]. With the multiplication of mobile terminals, interworking is seen to be a significant problem; as current interworking is limited to primitives provided by lower level network protocols such as IS-41. This study will show how a high level interworking can be achieved through the RM-ODP framework.

.

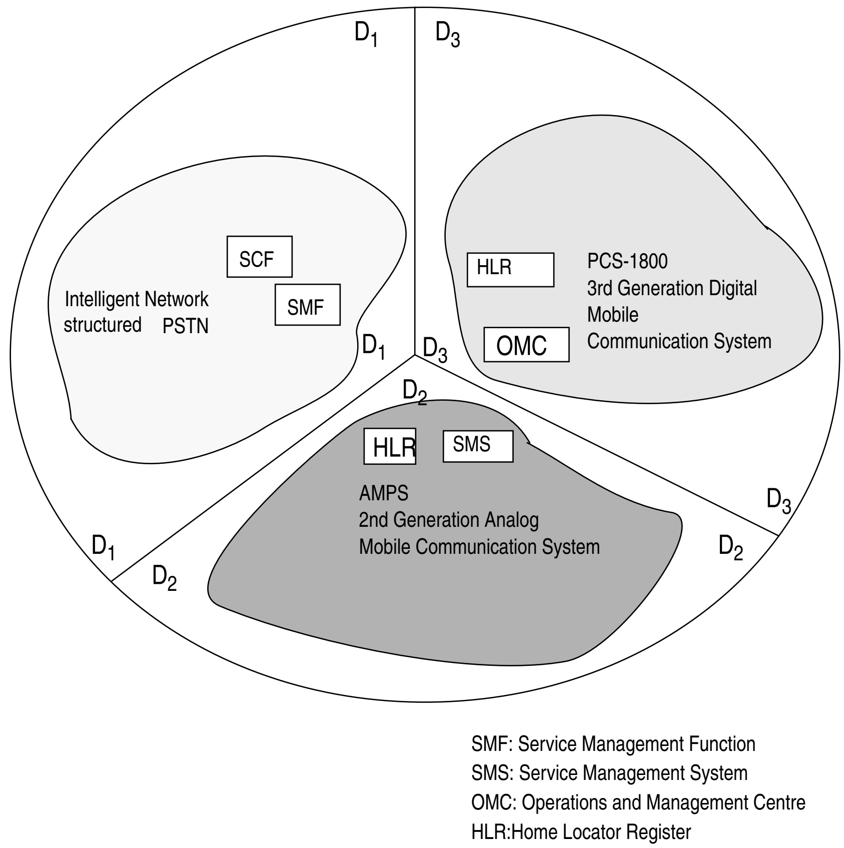


Figure 6.6 Technological Scenario of a GVNS

ODP Enterprise Recommendations on GVNS Management Policy

For the purpose of non-hierarchical management - a key aspect of IN interworking - RM-ODP prescribes the following rules:

- one domain is not obliged to perform an activity on behalf of another domain.
- each domain must have complete control of its interworking policies
- when a domain joins a group of interworking domains comprising the GVNS, all existing members of the GVNS community are obliged to be part of the GVNS.

These recommendations of management policy are in alignment with IN operator demand for independent management of control and data. (Operator co-operation issues were discussed in Section 5.1.2.) As a result, application of ODP enterprise level management concepts allows non-hierarchical control structures with a high level of interoperability.

6.1 The GVNS ODP Information Model Viewpoint

The information viewpoint describes the information processing requirements of the GVNS. The information model is divided into two sub-sections:

- *information viewpoint structure:* which shows the static structure of the information processing entities
- *information viewpoint messaging:* which shows the exchange of information between information processing entities.

Information Viewpoint Structure

The information processing entities and their static relationships are shown in the information viewpoint structure. This includes 3 parts

- **Object Diagrams**. The object diagrams show integrated IN and ODP network abstractions. Objects diagrams show static relationships and basic composition using the "part-of" relationship.
- **Trading Graph.** The trading graph, discussed in chapter 3 (see figure 3.7), shows the propagation of interworking requests between the interworked domains. In this example three domains interwork - D_1, D_2 and D_3. The requests are for user data (service profiles) or service logic programs. The trading graph represents distributed knowledge in the mobile GVNS. This trading graph for the BetterMouseTraps GVNS is illustrated in Figure 6.11.
- **Interworking State Model.** This thesis proposes an *Interworking State Model*. The interworking model is a finite state machine representation of a process that identifies points at which domains may interact with each other. Only events used for interworking of service logic are identified in the Interworking model.

The above points are elaborated in subsequent sections 6.1.1, 6.1.2 and 6.1.3.

Information Viewpoint Messaging

The information viewpoint messaging diagrams describe fundamental interworking procedures between domains interworking to provide the GVNS. Authentication, Call-Setup and Auxiliary Service Access are the important procedures. These procedures are shown below in table 4 and are further explained in section 6.1.4

Table 4:

Procedure	Entities and Activities Involved
Authentication	on-net SDP (user or terminal) attempts registration in a *home domain.*
	GVNS on-net SDP (user or terminal) attempts registration in a *visited domain*.

Table 4:

Call Setup	Between two on-net SDPs in a *home domain.*
	Between two on-net SDPs between *interworked domains*.
Auxiliary Service Access.	From an on-net SDP to a service logic program in a *home domain*.
	From an on-net SDP to a service logic program in *an interworked domain*.

6.1.1 Information Viewpoint: Structure

As explained in section 6.0.2, the BetterMouseTraps GVNS is provided by three interworking domains D_1,D_2 & D_3. Figure 6.7 shows the GVNS composition.

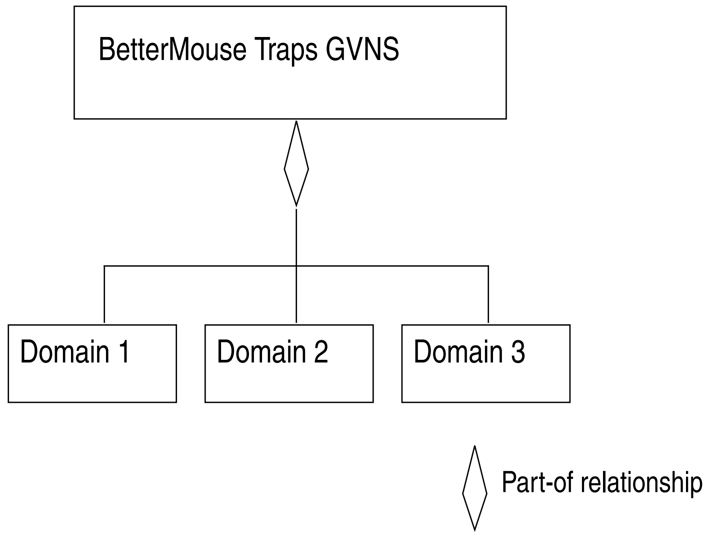


Figure 6.7 GVNS Information Viewpoint Structure

The GVNS basic information structure as shown in Figure 6.7 is refined by development of IN and ODP functionality in each domain. Important functional elements of each domain are highlighted. These elements shown in Figure 6.8, Figure 6.10 and Figure 6.9. The ODP trader and trader-administrator are combined with existing architectural models. The reuse of the ODP trader is uniform, but since the GVNS domains are technologically different, recombinations of ODP functionalities differs slightly with each case. These legacy systems influencing the GVNS are discussed briefly in the previous section (Figure 6.6), and correspond to ODP technology viewpoint constraints.

Domain 1: IN structured PSTN.

Domain 1 is a IN structured PSTN. Access to the network is through terminals fixed to locations.

As the focus in this work is on service control and management, the relevant entities are the Service Control Function (SCF) and Service Management Function (SMF). Details of these entities are discussed in sections 2.2.2 & 2.2.3. In typical IN specifications, the SCF is shown separate from (SDF)Service Data Function. However, since control of SDF is assigned to the SCF, (in this work it is assumed that the SCF accesses SDF whenever required) the SDF is considered an unnecessary detail and not shown in diagrams or discussed in the text.

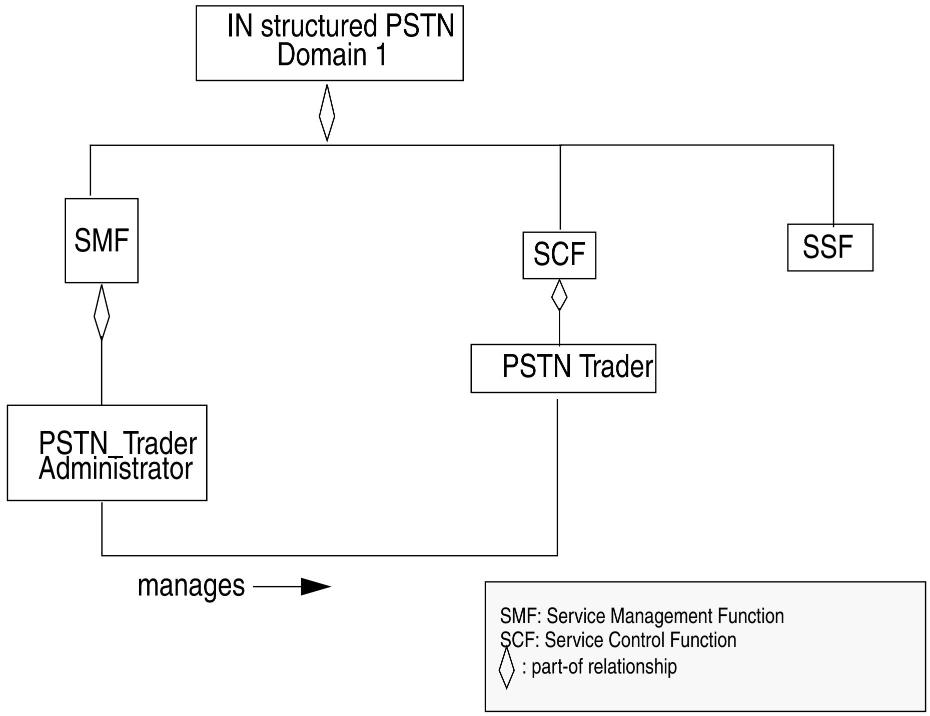


Figure 6.8 Information Structure of the PSTN Domain

The following ODP objects provide the necessary interworking functionality:

PSTN_Trader_Administrator:

The trader administrator establishes interworking contracts with trader administrators of other domains. In this case, it interworks the Operations and Management Center (OMC) in the PCS1800 domain and with the Service Management System (SMS) of the AMPS system. This interworking relationship is shown via the trading graph (Figure 6.11). The trader administrator comprises part of the functionality of the Service Management Function (SMF).

The PSTN_Trader_Administrator manages the PSTN_Trader. The PSTN_Trader_Administrator has the two primary responsibilities

- Link Management:
- Import Request Acceptance Management.

Link Management. The trader is directed to add unidirectional links to traders of other domains with which an interworking relationship is desired. In the BetterMouseTraps mobile GVNS links are to be established with the PCS_Trader and the AMPS_Trader. The PSTN_Trader uses the link to locate and connect to interworked domains in two cases:

1. When a service profile of a visitor SDP is needed for authentication and verification.
2. When a part of a distributed service is to be invoked in an interworked domain.

Import Request Acceptance Management: The trader must screen requests from external domains. Only domains that form part of a co-operative service are to be allowed to access the offer space. This activity is import request acceptance management.

In the BetterMouseTraps GVNS, import request acceptance is used in the following scenario: When the on-net SDP U_1^1 moves to D_2 or D_3 and attempts registrations, the visited domain (D_2 or D_3) will need to access the associated Service Profile of U_1^1. This service profile is located in the home domain (D_1), and the import request acceptance policy will specify that only Domains D_2 and D_3 have access to this service profile.

PSTN_Trader

The trader provides real time binding and configuration at the SCF-SCF level. Interdomain requests handled are for services and data. The PSTN_Trader is a sub-component of the SCF.

Domain 2: AMPS domain.

This domain is a 2nd generation analog mobile communication network. Access to the network is through mobile terminals.

The service management and control entities are the Service Management System (SMS) and the Home Locator Register (HLR).

Service Management System (SMS).

The Service Management System (SMS) is connected to all equipment in the service control and switching system. An important function of SMS is to provide a network overview and support the maintenance activities. In this GVNS the purpose of SMS is also to configure co-operative services developed with external domains.

Home Locator Register (HLR)

The home location register (HLR) is a database used for storage and management of subscriptions. The HLR is considered the most important database since it stores permanent data on SDPs, including the SDP service profile, location information, and activity status. When an customer buys a subscription from an operator, the SDP is registered in the HLR of that operator.

The following objects (Figure 6.9) provide necessary functionality.

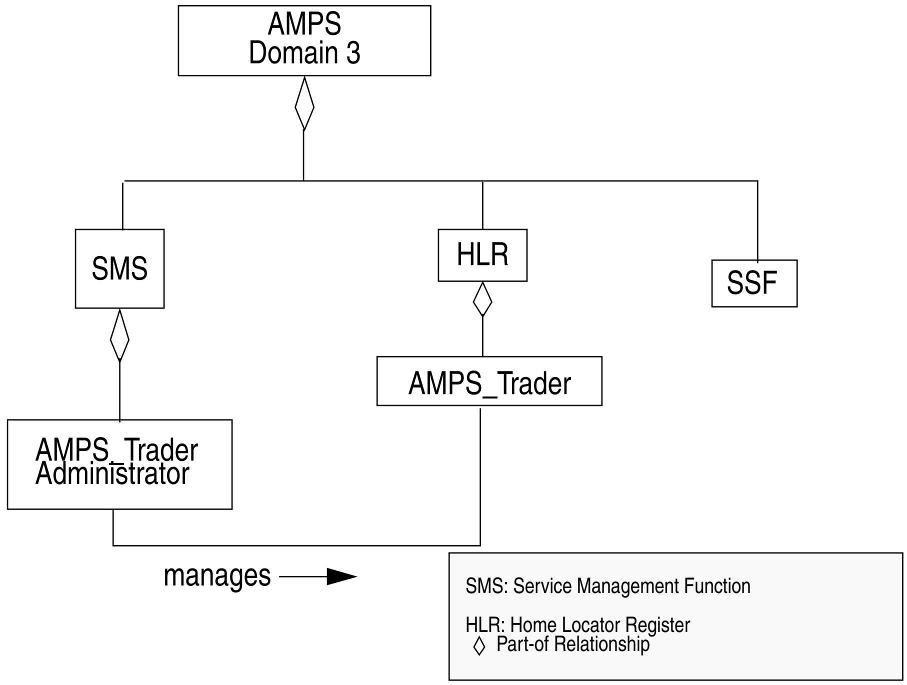


Figure 6.9 Information Structure: AMPS Domain

AMPS_Trader_Administrator:

The trader administrator has part of the functionality of the Service Management System (SMS). It establishes interworking contracts with the administrators of the other domains. Here, it interworks with the Operations and Management Center (OMC) in the PCS1800 domain and with the Service Management Function (SMF) of the PSTN system.

The AMPS_Trader_Administrator manages the AMPS_Trader. The AMPS_Trader_Administrator has the following two responsibilities.

•*Link Management.*

•*Import Request Acceptance Management.*

It is to be noted that the responsibilities of the AMPS_Trader_Administrator are identical with those of trader-administrators in other domains.

Link Management. Depending on the interworking contract, it directs its trader to add links to traders of other domains to be interworked with. In this case, links are to be established with the PSTN_Trader and the PCS_Trader. The trader uses links to locate and connect to inter-worked domains in two cases:

1. When a service profile of a visitor SDP is needed by a service logic program.
2. When a service logic program in a interworked domain is to be accessed.

Import Request Acceptance Management. The administrator programs the trader to accept requests only from recognized domains, i.e., those domains with which co-operative services are developed.

In the BetterMouseTraps when the on-net SDP T_2^1 moves to domains D_3 and attempts registration, the visited domain D_3 will need to access the Service Profile of T_2^1. The import request acceptance policy will specify that only domain D_3 has access to the service data and logic.

AMPS_Trader

The AMPS_Trader is a subcomponent and additional functionality added to the AMPS network HLR.

The relationship between the three domains is shown in the Trading Graph in Figure 6.11. The important associations between domains are between trader administrators and traders. Trader administrators interwork and decide high level enterprise policies. These relationships are developed off-line in non-real time. The trader interworks in real-time according to trading links and policies specified by the respective administrators.

Domain 3: PCS Domain

This domain is a PCS 1900 mobile network. Access to the network is through wireless terminals. With respect to the scope of this work, the management and control entities of this architecture are important; other architectural details are ignored.

The pertinent entities are Operation and Management Center (OMC) and the Home Locator Register (HLR).

Operation and Management Center (OMC).

The Operations and Maintenance Center (OMC) is connected to all equipment in the service control and switching system. The OMC is the functional entity from which the network operator monitors and controls the domain. An important function of OMC is to provide a network overview and support the maintenance activities. In this GVNS the purpose of OMC is also to configure co-operative services developed with external domains.

Home Locator Register (HLR)

The home location register (HLR) is a database used for storage and management of subscriptions. The HLR is considered the most important database since it stores permanent data on SDPs, including the SDP service profile, location information, and activity status. When an customer buys a subscription from an operator, the SDP is registered in the HLR of that operator.

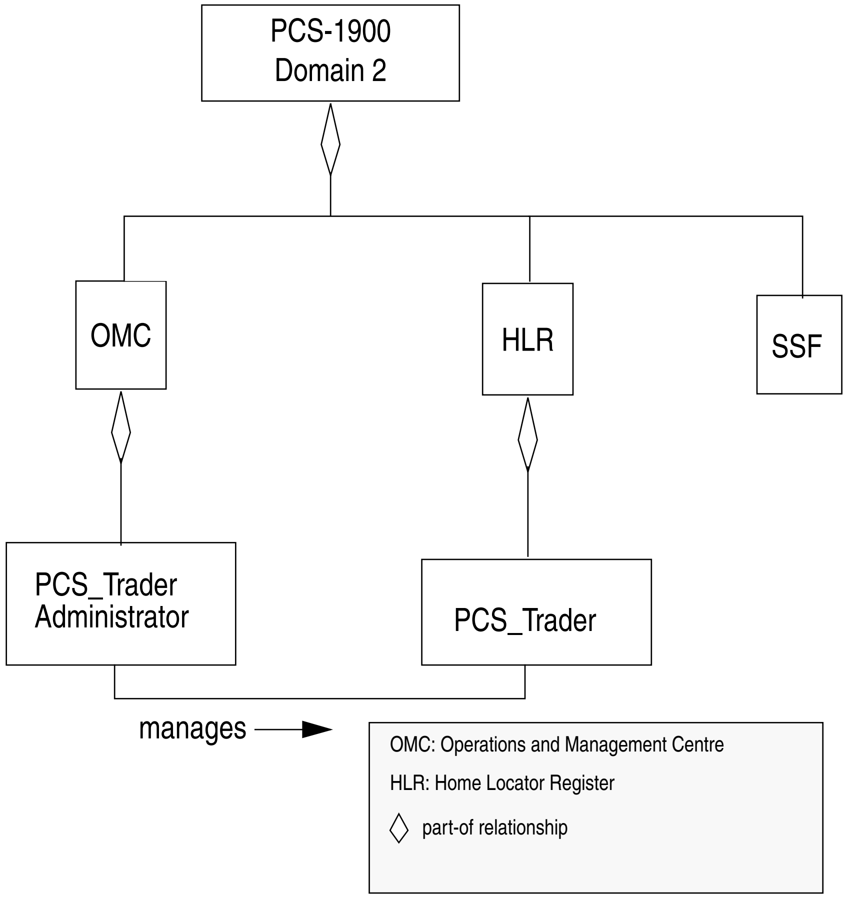


Figure 6.10 Information Structure PCS-1900 Domain

The following objects (Figure 6.10) provide the required functionality

PCS_Trader_Administrator:

The trader administrator includes part of the functionality of the Operations and Management Centre (OMC). The trader administrator establishes interworking contracts with the administrators of the other domains. In this case, it interworks with the Service Management Function (SMF) in the PSTN and with the Service Management System (SMS) of the AMPS system.

The PCS_Trader_Administrator manages the PCS_Trader. It has the following two responsibilities.

•*Link Management.*

•*Import Request Acceptance Management.*

It is to be noted that the responsibilities of the PCS_Trader_Administrator are identical with those of trader-administrators in other domains.

Link Management. Depending on the interworking contract, it directs its trader to add links to traders of other domains to be interworked with. In this case, links are to be established with the PSTN_Trader and the AMPS_Trader. The trader uses the link to locate and connect to inter-worked domains in two cases:

1. When the service profile of a visitor SDP is needed by a service logic program.
2. When a service logic program in a interworked domain is to be accessed.

Import Request Acceptance Management. The administrator programs the trader to accept requests only from recognized domains, those domains with which co-operative services are developed. Consider this scenario in the BetterMouseTraps GVNS: the on-net SDP U_3^1 moves to domains D_1 or D_2 and attempts registration, the visited domains (D_1 or D_2) will need to access the Service Profile of U_3^1. The import request acceptance policy will specify that only Domains D_1 and D_2 have access to the service data and logic.

PCS_Trader

The PCS_Trader comprises part of the functionality of the HLR. This enhanced network element allows interworking with external networks. This proposal is in alignment with other proposals for extensions to PCS 1800, where the HLR is being significantly enhanced to contain service control functionality [JR95].

6.1.2 BetterMouseTraps GVNS Trading Graph

The primary requirement in interworking telecommunications domains is the need for models to show information exchange between independent networks. Services in IN, are controlled by an SCF at the originating end or an SCF at the terminating end of a call. Mechanisms that support the exchange of information between any two SCFs do not exist. This mitigates against service provision across interworked domains. (This issue is discussed in Section 2.5)

An important example of a service where interworking between domains is required is personal and terminal mobility. In a mobile environment the location of a user and the associated SCF responsible for service invocation is non-deterministic at the time of system design (compile-time). Therefore, a representation that allows a service designer to define information structure and valid information flows between structural entities is required.

Such a representation should be simple, as potentially a large number of domains may need to co-operate and larger number of users supported. It should adequately reflect static structures but also lend understanding to the dynamic execution of the service provision. This thesis proposes the use of the ODP trading graph as such a representation.

The ODP trading graph is used to represent configuration of domains and the flow of control information across them. The trading graph illustrates the propagation of interworking requests between the 3 domains that consist of the BetterMouseTraps GVNS. The distributed knowledge of the GVNS is represented using three components of the trading graph:

- Interworking Links
- Trader Offer Space
- Trader Properties.

Interworking Links

An interworking link is a pointer to an interworked domain. Such a link is followed by the trader when there is a run-time binding requirement between a client and a server. In the context of a mobile GVNS client-server interactions are between a SCF and another SCF in an interworked domain.

Interworking links, diagrammatically, are shown as unidirectional arrows.

In the BetterMouseTraps GVNS all three SCFs need to interwork. The interworking requirements are summarized as follows:

1. SDPs from D_1 and D_3 will visit D_2, hence D_2 has trading links to the offer space of domains D_1 and D_3.
2. SDPs D_1 and D_2 will visit D_3, hence D_3 has links to D_2 and D_1.
3. SDPs from D_3 may visit D_1, hence D_1 is linked to D_3. D_1 is a fixed network, the mobile SDP (a terminal) in D_2 will not visit D_1, hence D_1 is not linked to D_2.

Trader Offer Space

In the mobile GVNS the offer space includes user data (SDP profiles) and shared service components. Interworking requests help access to a traders offer space. The restrictions of the access to a trader's service offer space are specified as trader properties. The following entities comprise the service offer space of the domains:

Table 5:

Domain	Entity	Type
D_1	$LSP(L_1^1)$	SDP-Location
	$USP\ (U_1^1)$	SDP-User
D_2	SID(VMS)	Voice Message System Service
	$TSP(T_2^1)$	SDP-Terminal
D_3	$USP(U_3^1)$	SDP-User

Trader Property.

Access to service components and SDP profiles must be well regulated. Regulation is essential for security, billing and competitive advantage. This regulation is enforced by associating each request with trader properties. Here the most important property is the Import request acceptance policy.

The detailed specification of the Import Request Policy may be developed in the computational viewpoint. (When prescribed in the Computational Viewpoint this policy governs and specifies the domains that may access relevant objects, interfaces, operation on interfaces, and also parameters passed during operation invocation.)

The BetterMouseTraps GVNS Trading Graph

Figure 6.11 is the BetterMousetraps GVNS Trading Graph. It shows the domains interworking to provide the GVNS service. The domains are shown as circles. The pointers between domains represent interworking links. Each domain is represented by a circle, contains data and service components that are to be shared (Offer Space). The shaded rectangles at the head of the arrows represent the presence of a Import Request Policy. This policy regulates access to service programs and data. Requests only from authorized domains are allowed.

An alternative understanding of the trading graph in Figure 6.11 is to view interworking domains as sets (shown in graph as circles). The elements of the set are shared components and data (shown in trading graph as squares). The pointers represent elements that may be needed in one set that are located in other sets.

.

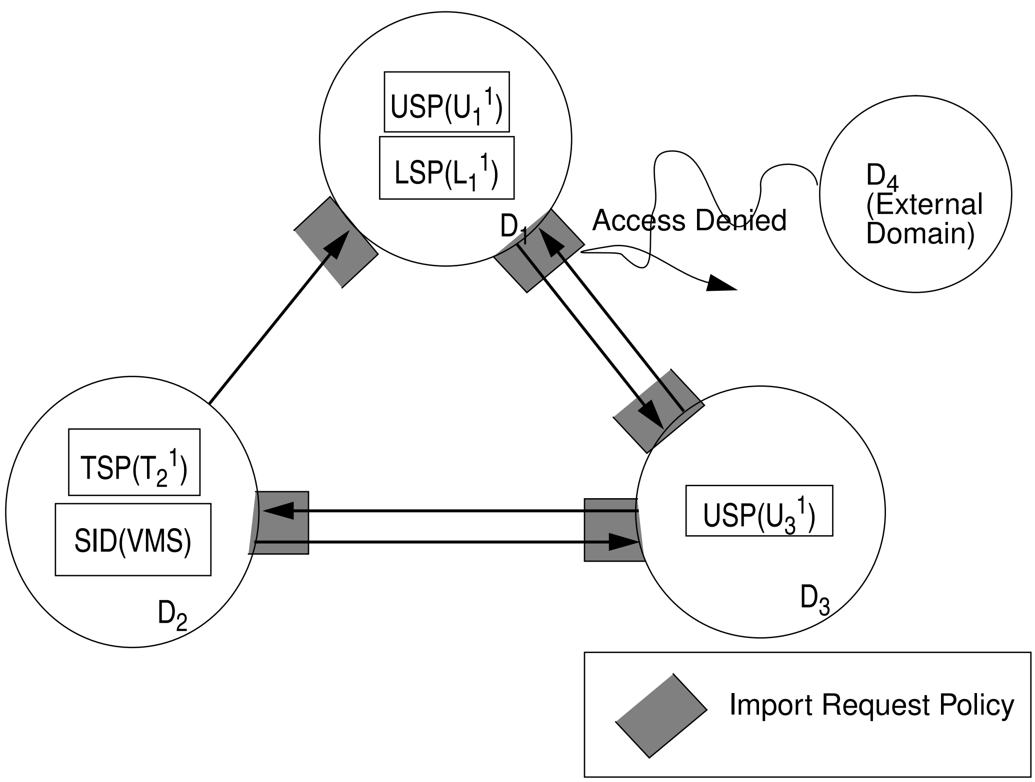


Figure 6.11 BetterMousetraps mobile GVNS Trading Graph

Advantages of Trading Graphs

Using trading graphs has numerous advantages. The graph isolates the cross-references between domains and highlights interworking aspects. The shared service logic and data are easily visible. Policies associated with co-operation and administrative constraints can also be explained

6.1.3 Interworking State Model

The previous chapters have established the context of the work; the limitation of IN in service provision in a distributed mobile service environment (Section 2.5). It is clear (2.5.1, 2.5.2 & 2.5.3) that SCF-SCF and SDF-SCF functionality and relationships need to be developed; especially representation of the process of interworking and the points at which information exchange occurs.

This work proposes an Interworking State Model that allows development of distributed services. As no suitable proposal that meets this requirement exist, this model is the second primary contribution of the this work.

The purpose and scope of the ISM

The ISM is the representation of a process that runs within the SCF but outside the trader. It identifies points in interworking where service components are permitted to interact with each other. The ISM is used by any service component that has interworking requirements. The model consists of two parts, an originating section and an terminating section. The originating section is resident in the domain that invokes the interworking request. The terminating ISM in resident in the domain that receives the interworking request. Any service that uses the ISM sees the state of the distributed service through information of the ISM. This proposal fulfills a basic requirement; that it is essential that both interworked domains maintain the coherent and coordinated perception of the state of the distributed service logic.

The Interworking State Model (ISM) allows development of co-operative services across administrative domains: it identifies points in interworking where service functions are permitted to interact with each other. When standardized and adopted by all domains, this ISM would provide a uniform view of distributed service processing between interworking domains. The basic conceptual model is shown in Figure 6.12.

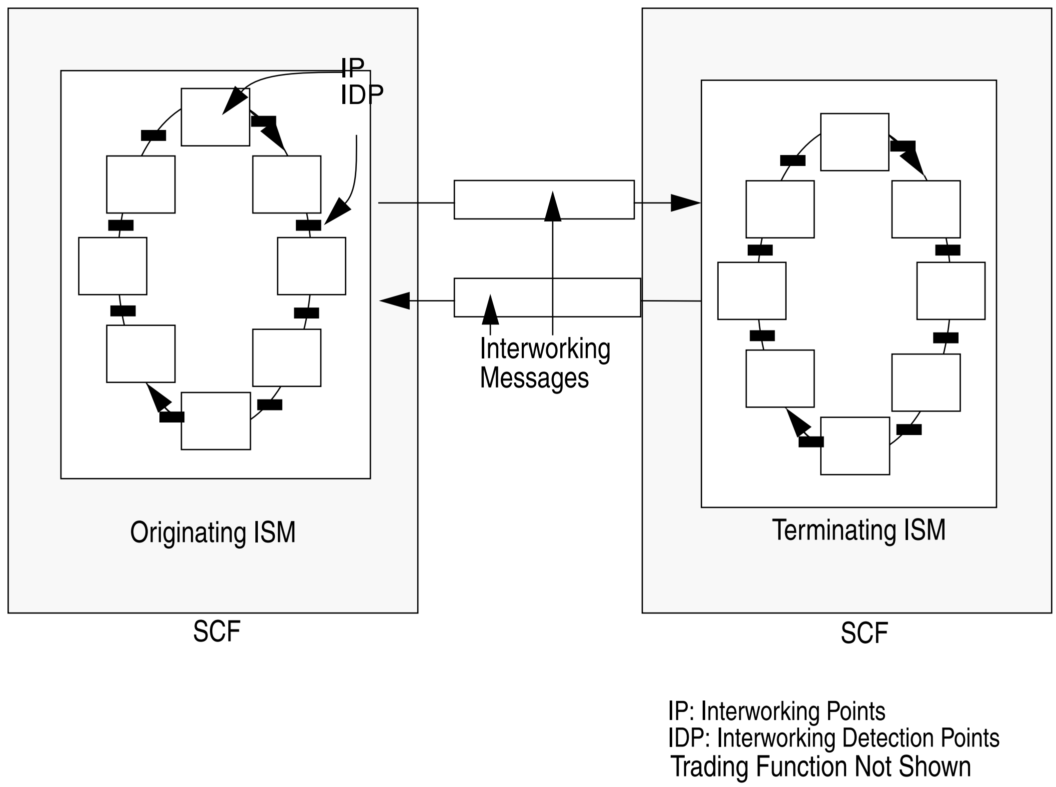


Figure 6.12 Interworking State Model: Conceptual View

The process of interworking from origination to termination can be described by a serial loop of activities. Each activity is called an Interworking Point (IP). Interworking points (IPs) are characterized by actions performed within, and are an external view of the state of interworking.

Placed between (IPs) are Interworking Detection Points (IDPs). These are points at which conditional flags are installed. If a condition at a IDP is satisfied, then interworking is required, and must take place. If conditions are not satisfied then service processing continues. Service processing may be suspended while waiting for interworking requests to complete.

The ISM is instantiated independent of any specific service logic programs. The process implementing the ISM is invoked when co-operative service are to be provided.

Advantages of the ISM

The Interworking State Model provides a uniform view of interworking activities. Such a model allows the SCFs to have vendor independent interactions with each other. The scope of all of existing IN standards has been limited to services initiated at the originating or terminating ends, i.e. at a single point of control; and two way calls only. Service with such restrictions are called Type A services. Multi-ended, multi-domain and multi party services are called Type B services. With the proposed ISM interdomain services and multi-party calls become possible. An example of such type B services is shown in Figure 6.13.

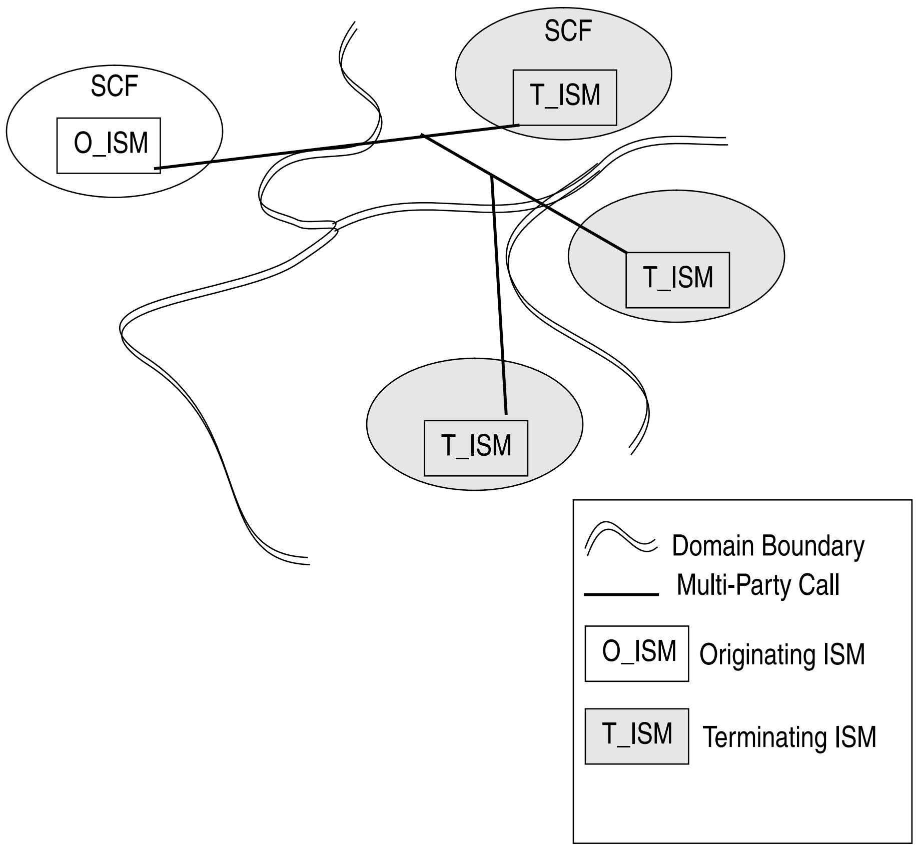


Figure 6.13 A Type-B Multi-Party Call Using the ISM

Four parties take part in the call. The Originating SCF is shown as a white circle. The terminating SCFs are shown as shaded circles. Domain boundaries are shown a double lines.

ISM Components: Originating and Terminating Sections

The ISM consists of two sections - the originating section and the terminating section. The originating section initiates the interworking. When an interworking request is received at a domain, the terminating section is instantiated. Both sections consists of IPs - activities performed at a state; and IDPs - points between IPs where conditional flags are installed.

The originating ISM (Figure 6.14) consists of the following IPs.

0. Null.

This is the null state of the ISM, no activity is performed.

1. Analyze Information.

The user, terminal and location information are analyzed.

2. Import SDP-Location Profile.

The profile of the SDP-location is imported. A location-SDP refers to an entity fixed to a geographic location, an example of which is shown in Figure 4.14. It is not mobile and it's profile is available locally, hence no interworking is required.

3. Import SDP-Terminal Profile.

The profile of the SDP-terminal is imported.

4. Import SDP-User Profile.

The profile of the SDP-User is imported.

5. Resolve Conflicts and Authorize Registration.

Any conflicts between service profiles are resolved. According to the priority assigned to services, the appropriate service triggers are loaded into the SSF.

Example:

The necessity of loading data (also known as loading triggers) into the switch is illustrated by the following example. An on-net SDP-User attempts to register on the GVNS. The user wishes to connect to the voice message system, identified by the number 20 in the CDNP (Customer Defined Dialling Plan). This user registers through a borrowed terminal. This terminal has an abbreviated dialling service: where 20 represents a 11 digit number 1-613-565-5885. Here there is conflict between service profiles. The registration procedure resolves these conflicts by nullifying the abbreviated dialling service associated with the terminal. When 20 is entered, the data loaded into the switch invokes the GVNS service associated with the user and not the abbreviated dialling service associated with the terminal.

6. Analyze Information.

After successful registration, the user enters some digits. This information is collected and analyzed. If the information corresponds to invocation of a service, the service is located and provided to the user. This service invocation is represented in step 7. Incase the user wishes to connect to an on-net SDP, step 8 is invoked.

7. Import Service.

When a user requests access to a service, the service is located, and provided to the user. If the service is located in a remote domain, and interworking request, step 9 is executed. If service is not requested, then processing continues as though step 7 did not exist.

8.Import Called SDP Profile.

When the user requests a connection to another SDP, the SDP profile is located. The import request is placed to the trader to find the called SDP. If the SDP is not available in the local domain, an interworking request is performed.

9. Interwork.

In a co-operative provisioning environment, request for a service or SDP will require interworking across domain. Step 9 is an interwork request that establishes communication with the terminating section in the interworked domain.

After steps 7 or 8, the service address or the SDP address is available. The switch is provided with this routing information and a connection is established.

At the completion of these activities the state reverts to null. On any error conditions, the state goes to disconnect, and then null.

After registration, steps 1-5 are not invoked. Subsequent to this the process uses steps 6, 7, 8 & 9 only.

10. Disconnect.

Disconnect in case of error.

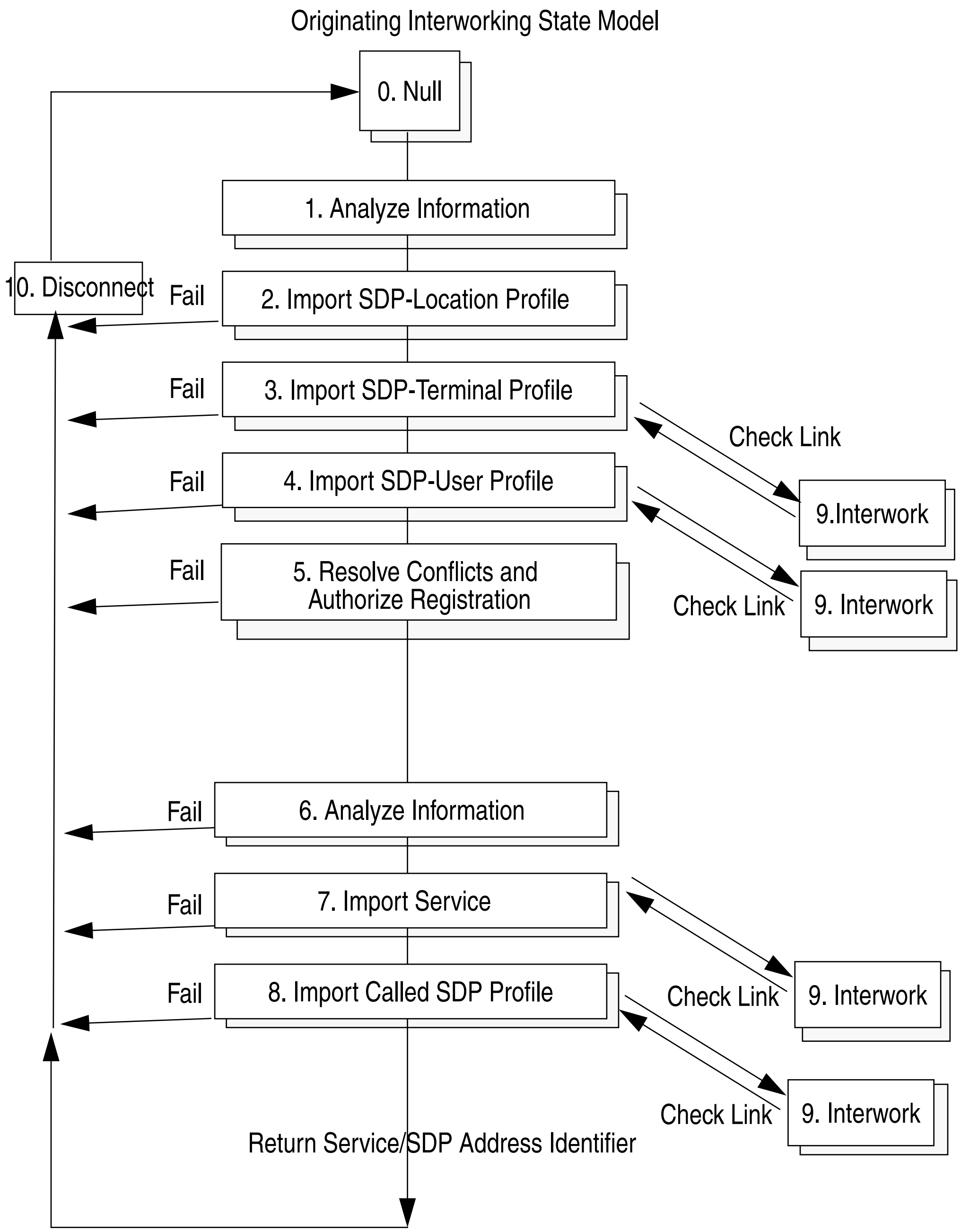


Figure 6.14 Interworking State Model: Originating Detail

The Terminating section of the ISM consists of the following IDPs.

0. Null.

This is the null state of the ISM, no activity is performed.

1. Analyze Interwork Request.

The interworking request is analyzed. The service being requested and the domain the request originates from are checked against the import request policy. In case the policy approves the request, steps 2 or 3 are executed

2.Import SDP Profile.

If the request is for a local SDP the SDP profile address is returned.

3.Import Service.

If the request is for a service, the Service address identifier is returned.

4. Interwork.

In the event that the requested SDP or service are unavailable, another interworking request is initiated.

On any error conditions, the ISM disconnects by moving to the disconnect state. At the completion of the interworking, the state reverts back to null.

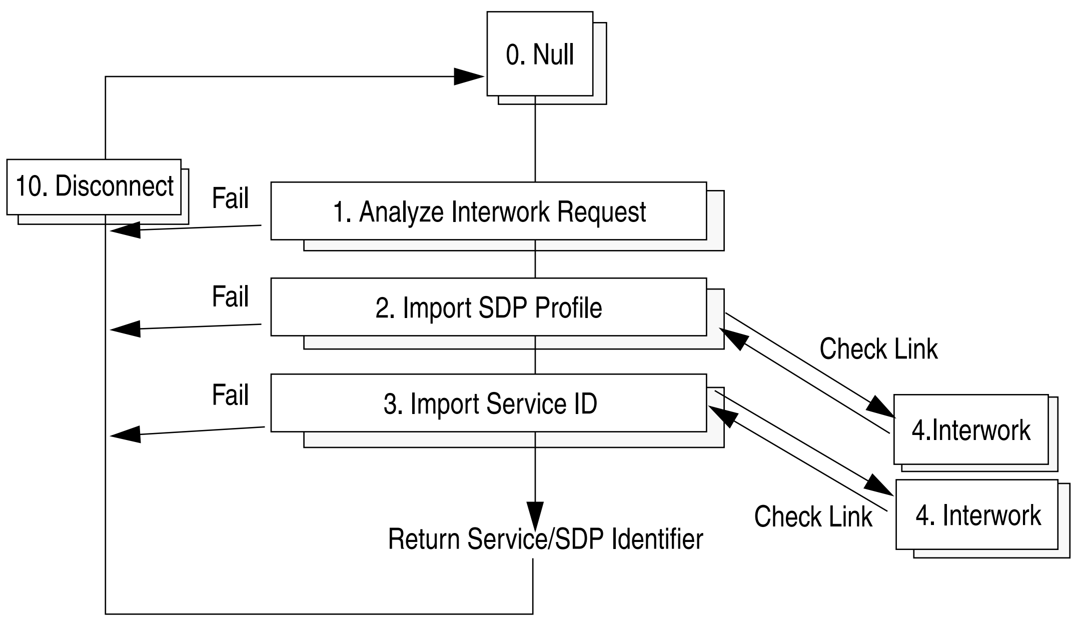


Figure 6.15 Interworking State Model: Terminating Details

6.1.4 GVNS Information Viewpoint: Messaging

In this section on GVNS messaging, the information flows between domains are explained. These flows indicate information exchanged between objects. The object interface definition, message formats, or computational technology used is not relevant in this viewpoint. Only information for successful information flows is shown, but behavior of the system under error conditions is not explained. This is in alignment with standard ITU methodology [Q.12xx].

It is important to demonstrate functional validity of the concepts introduced in this work: service profiles, ISPT, SDP-types, trading graph and ISM. From the GVNS case fundamental procedures are illustrated. These key procedures are listed in table 6.

Table 6:

Authentication	SDP registration in home domain.
	SDP registration in visited domain.
Call Setup	Between two on-net SDPs in the same domain
	Between two on-net SDPs in different domains.
Auxiliary Service Access.	From on-net SDP to a service logic program in the same domain.
	From an on-net SDP to a service logic program in a different domain.

A note on terminology used for Information Viewpoint Messaging.

Since is assumed that switching functionality is uniform, a switching function is called a Service Switching Function (SSF). The HLR, like the SCF is responsible for service control. For purposes of clarity the information flows only show a SCF. The HLR functionality is assumed to be encapsulated within the SCF.

Authentication

When a SDP identified attempts registration to access the GVNS, the network authenticates the SDP or rejects the registration attempt. Authentication is of two types: local authentication and visitor authentication.

Local Authentication: When a SDP attempts registration in it's home domain. In this case, no interworking is required as the SDP-profile is available locally.

Visitor Authentication: When a SDP attempts registration in a visited domain. Here, interworking between domains would verify authenticity of the registration attempt. Incase of a successful registration attempt, ISPT is performed.

Local Authentication.

Local authentication can further be of two types. The first type is authentication of terminal SDP, the second type is authentication of user SDPs.

The authentication procedure for terminal SDPs (Figure 6.16 is of an external behavior view and Figure 6.17 shows the related event diagrams) is clarified by the following scenario:

A company worker, on duty driving the truck, wishes to access the GVNS from the terminal on the truck. D_2 is the AMPS domain.

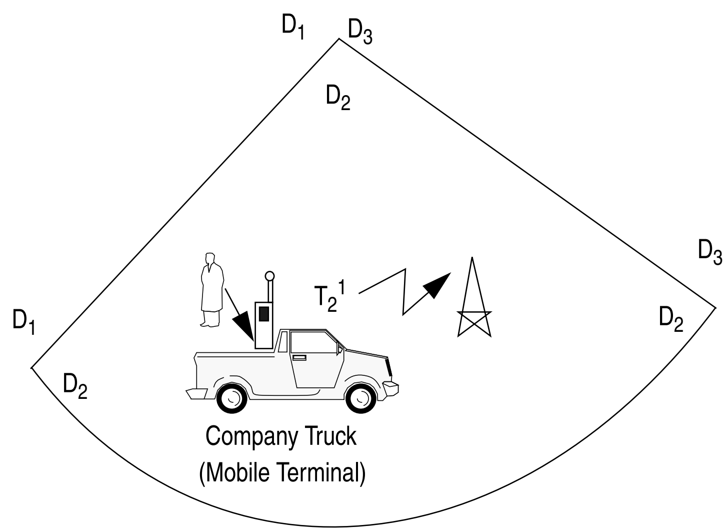


Figure 6.16 Local Authentication: Terminal SDP

1. The terminal is turned on, it attempts to register by sending a message with terminal identifier to the SSF (RegistrationReq(IMEI) in Figure 6.17).

2. The SSF sends the location identifier to the SLP within the SCF. (SendSDPID (Location). (SDPID: ServiceDeliveryPoint Identifier))

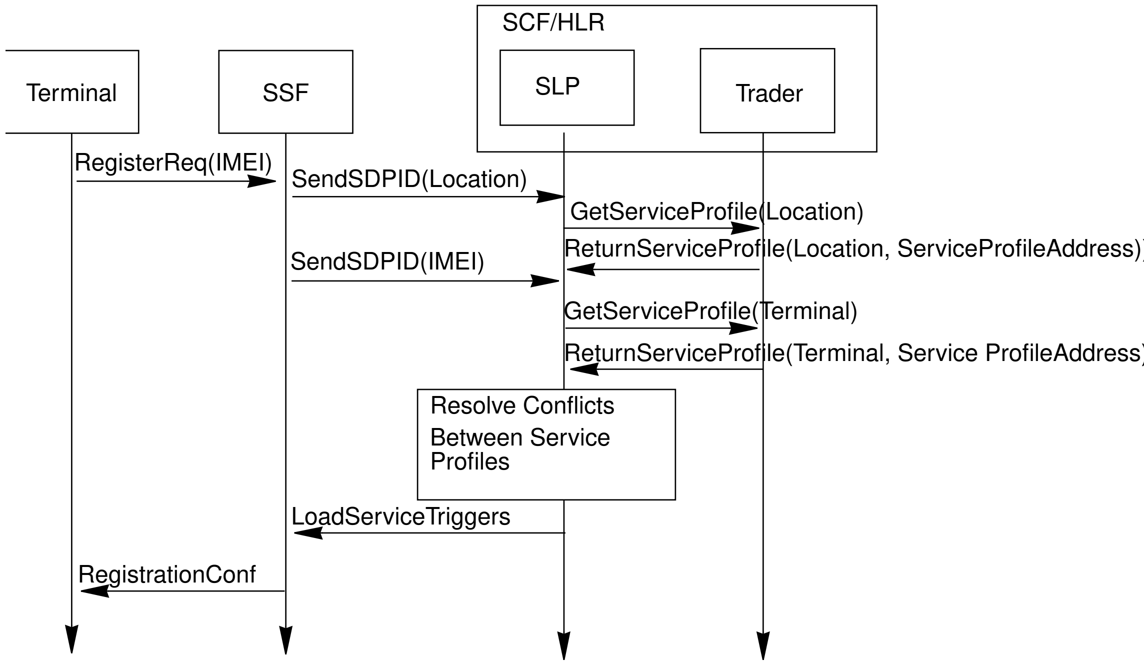


Figure 6.17 Event Diagram: Location Authentication of Terminal.

3. The SLP imports the location SDP service profile. (GetServiceProfile(Location))

4. The trader returns the service profile address. (ReturnServiceProfile(Location, Service ProfileAdress)).

5. The SSF sends the terminal identifier to the SCF. (Send SDPID (IMEI)).

6. The SLP imports the terminal service profile. (GetServiceProfile (IMEI))

7. The trader returns the terminal service profile address. (ReturnServiceProfile(IMEI, Service ProfileAddress)

8. The terminal and location profiles contain references to subscribed service in their respective service sets. The SLP resolves conflicts between the service profiles.

9. In the absence of conflicts, the appropriate data triggers are loaded (LoadServiceTriggers) into the SSF.

10. The SSF sends RegistrationConf message to the terminal.

The terminal is now successfully registered and is identified as an active part of the GVNS.

Local Authentication of On Net User.

A user attempts registration from a fixed terminal, or from a mobile terminal pre-registered according to the steps above in Figure 6.16 & Figure 6.17.

Consider the following scenario (Figure 6.18 & Figure 6.19) that is an example of local user authentication. The president of BetterMouseTraps attempts registration at a fixed terminal in $Domain_3$ using PCS-1800.

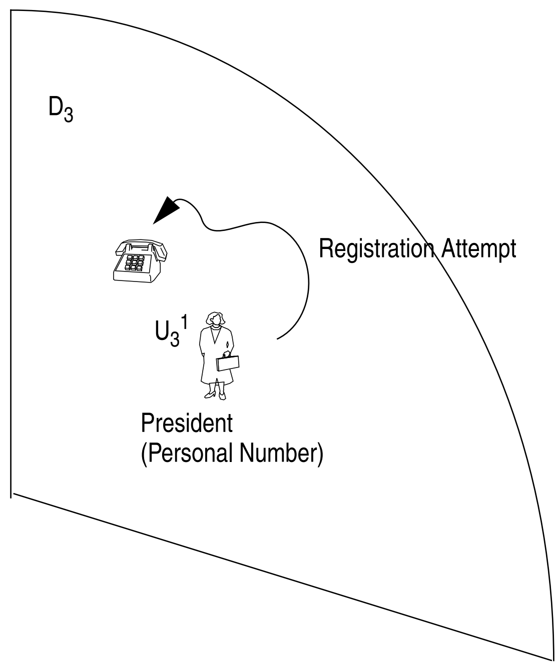


Figure 6.18 Local Authentication: UserSDP

1. The president enters IMUI.
2. The IMUI is sent to the SSF. (RegisterReq(IMUI))
2. The SSF sends the location identifier to the SLP. (SendSDPID(Location))
3. The SLP imports the service profile by sending a GetServiceProfile (Location) message to the trader.

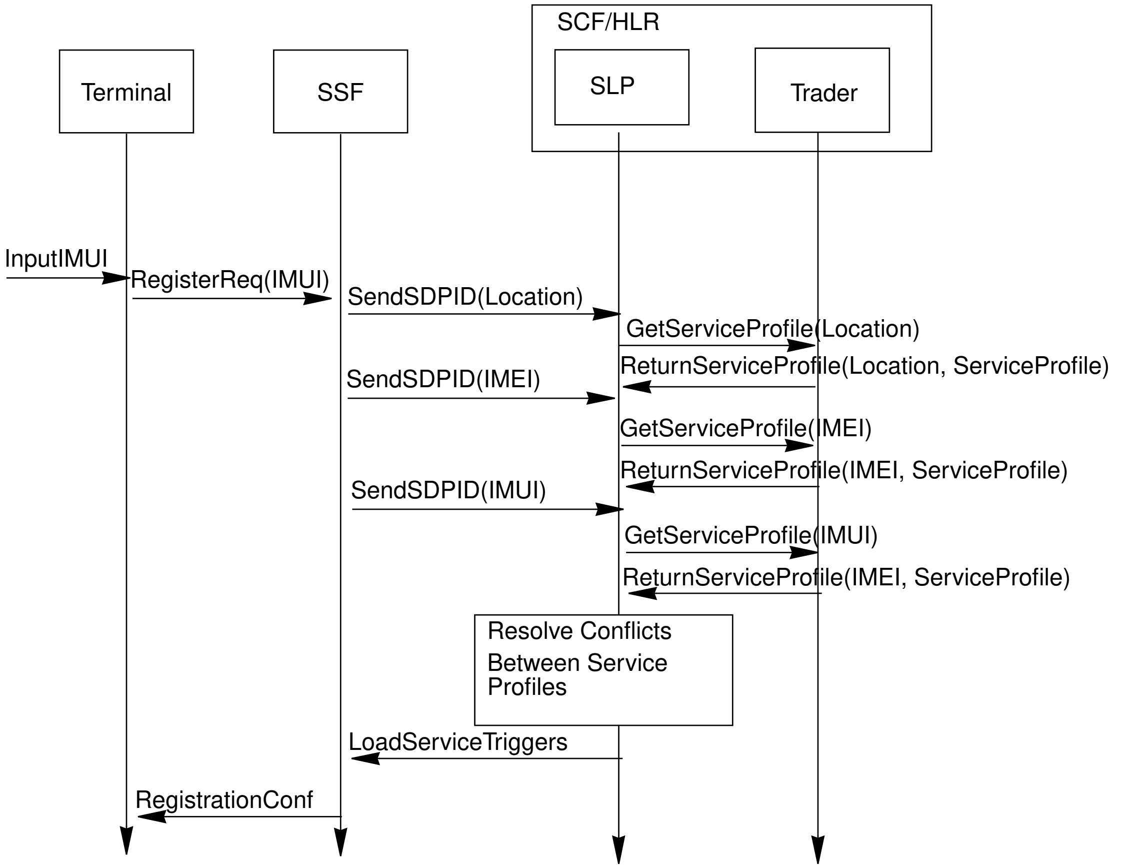


Figure 6.19 Event Diagrams: Local Authentication User.

4. The trader returns the service profile address. (Return Service Profile (Location, ServiceProfile)).

5. The SSF Sends the terminal identifier to the SLP. (SendSDPID(IMEI)).

6. The SLP imports the terminal service profile with (GetServiceProfile(IMEI)).

7. The Trader returns the terminal service profile address with (ReturnServiceProfile(IMEI, Service ProfileAddress)).

8. The SSF sends the user identifier to the SLP. (SendSDPID(IMUI))

9. The SLP imports the user service profile with GetServiceProfile(IMUI).

10. The trader returns user service profile address with ReturnServiceFile(IMUI, ServiceProfileAddress)

11. The user terminal and location profiles contain references to subscribed services in their respective service sets. The SLP resolves conflicts between the service profiles.

12. In the absence of conflicts, the appropriate data triggers are loaded (LoadServiceTriggers) into the SSF.

10. The SSF sends RegistrationConf message to the terminal.

The user is now successfully registered and is identified as an active part of the GVNS.

Visitor Authentication

Visitor authentication is of two types: for terminals and users. Authentication for terminals is shown in Figure 6.20 and Figure 6.21 and authentication for users is shown in Figure 6.22 and Figure 6.23.

To illustrate visitor terminal authentication consider the scenario in Figure 6.20. The truck with the terminal moves from $Domain_2$ (AMPS)to $Domain_3$ (PCS-1800).

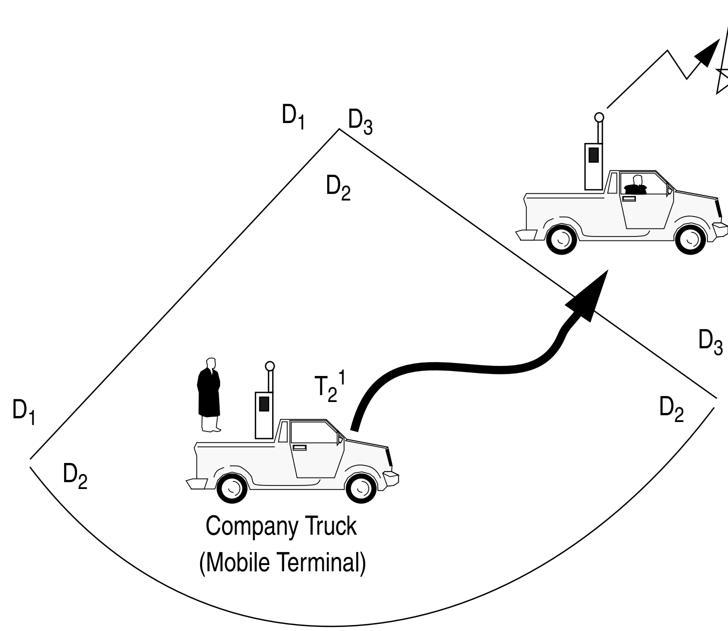


Figure 6.20 Visitor Authentication: Terminal

1. The terminal is turned on and it sends a RegisterReq(IMEI) message to the SSF.
2. The SSF sends the location identifier to the SLP within the SCF. (SendSDPID (Location).
3. The SLP imports the location SDP service profile. (GetServiceProfile(Location))
4. The trader returns the service profile address. (ReturnServiceProfile(Location, Service Pro-

fileAddress)).

5. The SSF send the terminal identifier to the SCF. (Send SDPID (IMEI)).

6. The SLP imports the terminal service profile. (GetServiceProfile (IMEI))

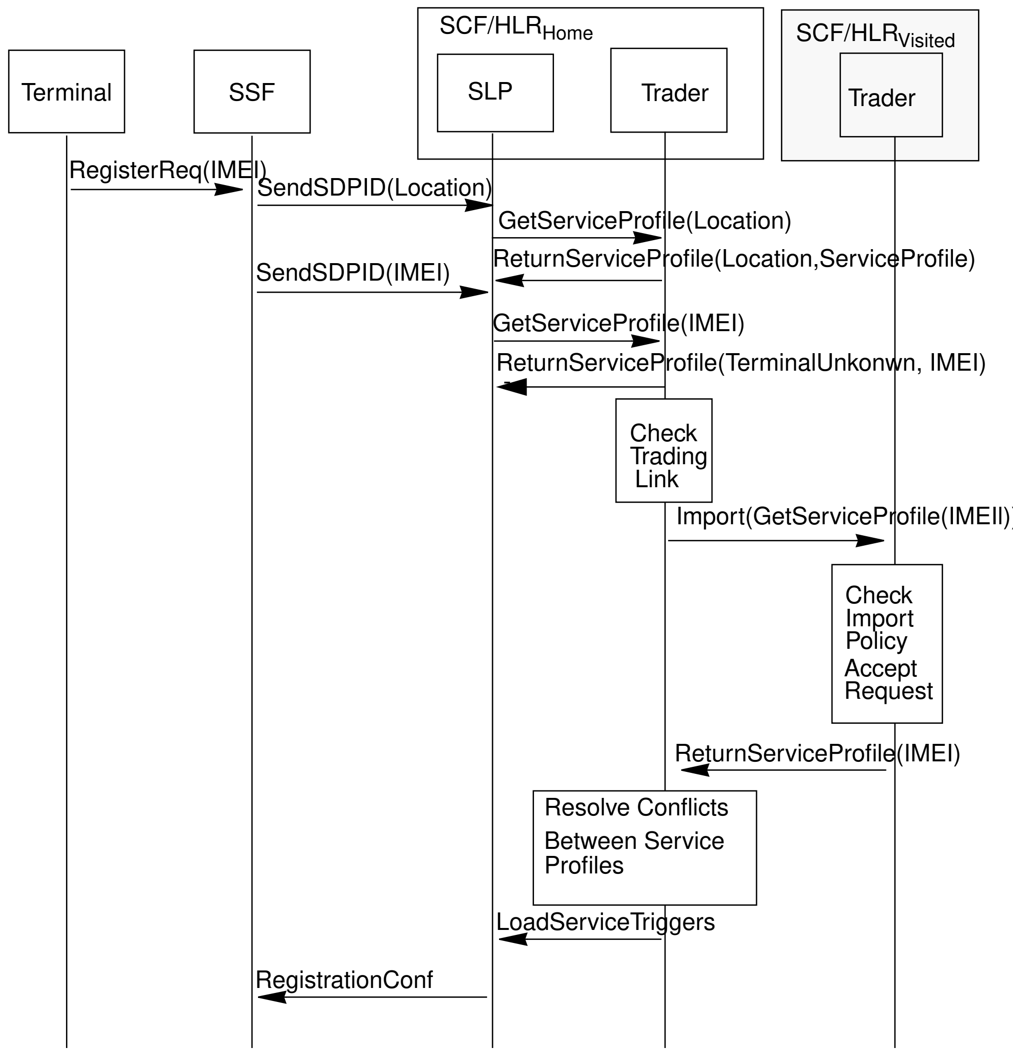


Figure 6.21 Event Diagrams: Visitor Authentication Terminal.

7. Since the terminal is a visitor, its profile is not available . The message ReturnServiceProfile(TerminalUnkonwn, IMEI) is returned to the SLP. This message is returned so that the SLP may need to invoke other services or update its own information regarding visiting SDPs.

8.The IMEI contains information of the terminals home domain. The trader interworks along with the trading graph link associated with the visitor terminal's home domain and places an import request to the remote trader (Import(GetServiceProfile(IMEI))). (If a link is not found in the trading graph, this implies that interworking agreements are not in place, and the authentication attempt fails.)

9. On receiving the interworking request, the remote trader checks import request policy. On validation of the interworking request, it accepts the request, searches for the terminal service profile and returns it. ReturnServiceProfile(IMEI). This is called Internetwork Service Profile Transfer (ISPT). In this and in a subsequent examples the engineering mechanisms for ISPT are not explained. Instead event representing ISPT is indicated.

10. The SLP reads the terminal service profile, checks for conflicts between terminal and location profiles. In the absence of conflicts, it sends a (LoadServiceTriggers) message to the SSF and appropriate data triggers are loaded.

11. The SSF sends RegistrationConf message to the terminal.

Visitor Authentication: User

Visitor user authentication is illustrated using an intricate case, shown in Figure 6.22. The mobile terminal T_2^1 from Domain $_2$ (AMPS) visits Domain $_3$ (PCS-1800). The user U_1^1 visits from Domain $_1$ (PSTN), and wishes to register on T_2^1. T_2^1 is already registered as an active part of the GVNS.

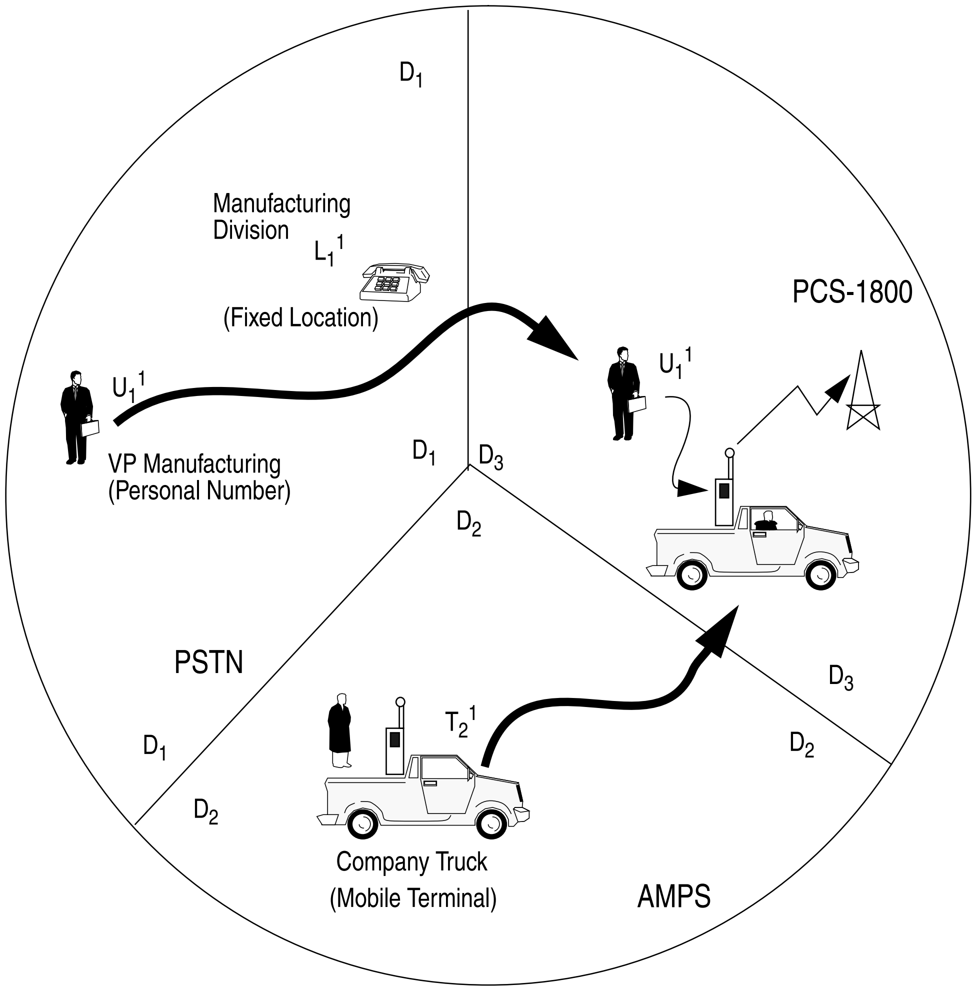


Figure 6.22 Visitor Authentication: User

1. The terminal is turned on and it sends a RegisterReq(IMEI) message to the SSF.

2. The SSF sends the location identifier to the SLP within the SCF. (SendSDPID (Location).

3. The SLP imports the location SDP service profile. (GetServiceProfile(Location))

4. The trader returns the service profile address. (ReturnServiceProfile(Location, Service ProfileAddress)).

5. The SSF send the terminal identifier to the SCF. (Send SDPID (IMEI)).

6. The SLP imports the terminal service profile. (GetServiceProfile (IMEI))

7. Since the terminal is a visitor, its profile is not available, and the ReturnServiceProfile(TerminalUnknown, IMEI) is returned to the SLP. This is returned incase the SLP is required to maintain information about visiting SDPs.

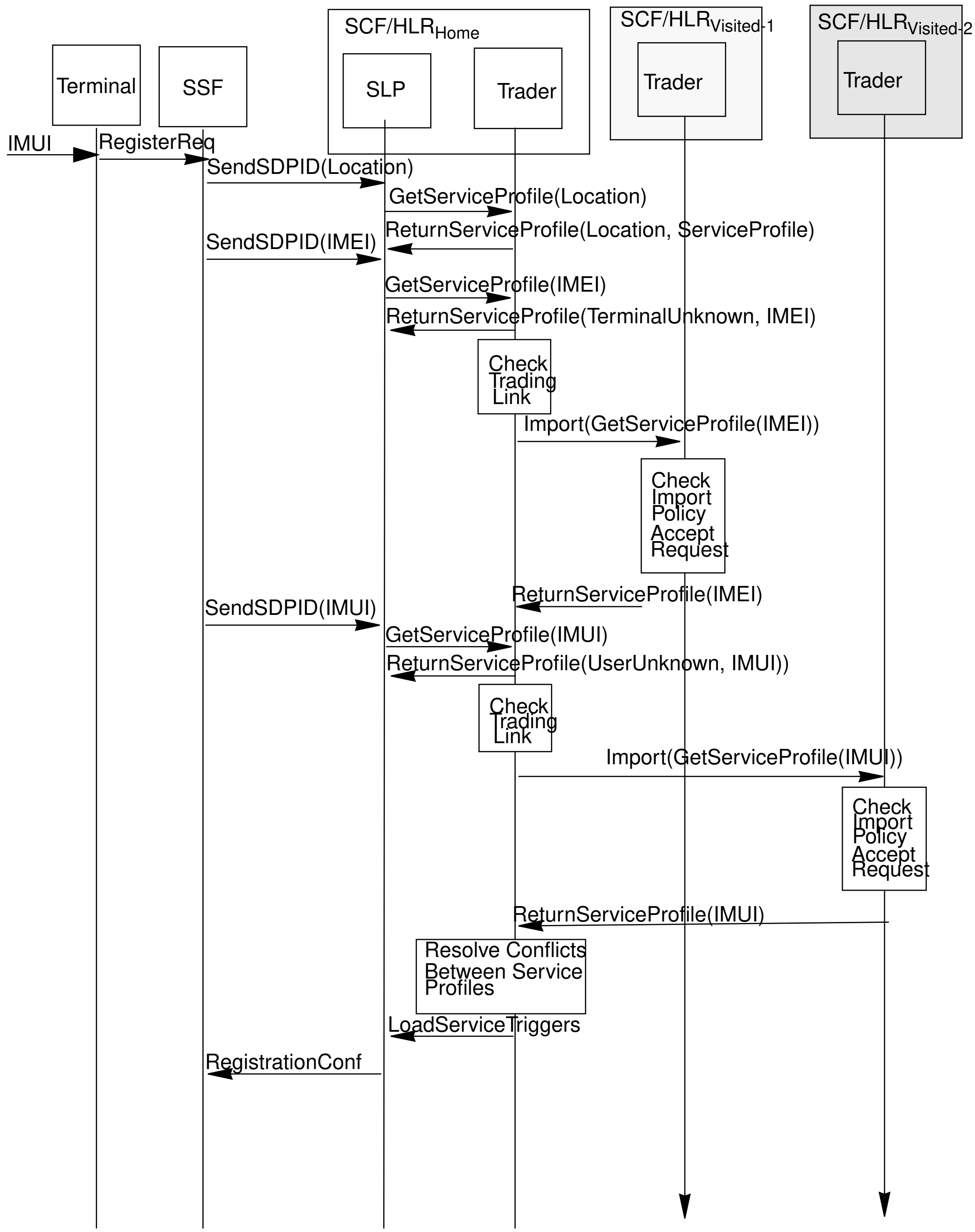


Figure 6.23 Event Diagrams: VIsitor Authentication User.

8. The IMEI contains information of the terminal's home domain. The trader

interworks along with the trading graph link associated with the visitor's home domain and places an import request to the remote trader (Import(Getserviceprofile(IMEI))). (If a link is not found in the trading graph, this implies that interworking agreements are not in place, and the authentication attempt fails.)

9. On receiving the interworking request, the remote trader checks import request policy. On validation of the interworking request, it accepts the request, searches for the terminal service profile and returns it. This is called Internetwork Service Profile transfer (ISPT)

10. The SSF send the user identifier to the SCF. (Send SDPID (IMUI)).

11. The SLP imports the user service profile. (GetServiceProfile (IMUI))

12. Since the user is a visitor, its profile is not available, and the ReturnServiceProfile(TerminalUnknown, IMUI) is returned to the SLP.

13.The IMUI contains information of the user's home domain. The trader interworks along with the trading graph link associated with the visitor's home domain and places an import request to the remote trader (Import(Getserviceprofile(IMUI))). (If a link is not found in the trading graph, this implies that interworking agreements are not in place, and the authentication attempt fails.)

14. On receiving the interworking request, the remote trader checks import request policy. On validation of the interworking request, it accepts the request, searches for the user service profile and returns it (ISPT).

15. The SLP checks for conflicts between user, terminal and location profiles. In the absence of conflicts, it sends a (LoadServiceTriggers) message to the SSF and appropriate data triggers are loaded.

16. The SSF sends RegistrationConf message to the terminal.

Call Setup:

Call setup is performed after the authentication procedures have successfully been executed. Call Setup is typified by two scenarios. The first, when both the called and caller SDPs are located in the same domain. The second, when the called SDP is not registered in the local domain.

The first scenario has no interworking requirements as service profiles for both SDPs are local. In the second scenario, the local SCF has to interwork, locate the called SDP, resolve conflicts of service profiles and establish connection.

To illustrate the first scenario consider the example shown in Figure 6.27. The president U_3^1 visits domain 1 from domain 3. After successfully registering at Domain 3, U_3^1 wishes to make a call to the manufacturing unit L_1^1.

The following steps are executed to place the call.

1. U_3^1 dials 18 the logical address of L_1^1 according to the Customer Defined Numbering Plan (CDNP). (The CDNP is shown in Figure 6.4)

2. The SSF sends a message InfoCollected(CallerSDPID, DialledDigits) to the SLP.

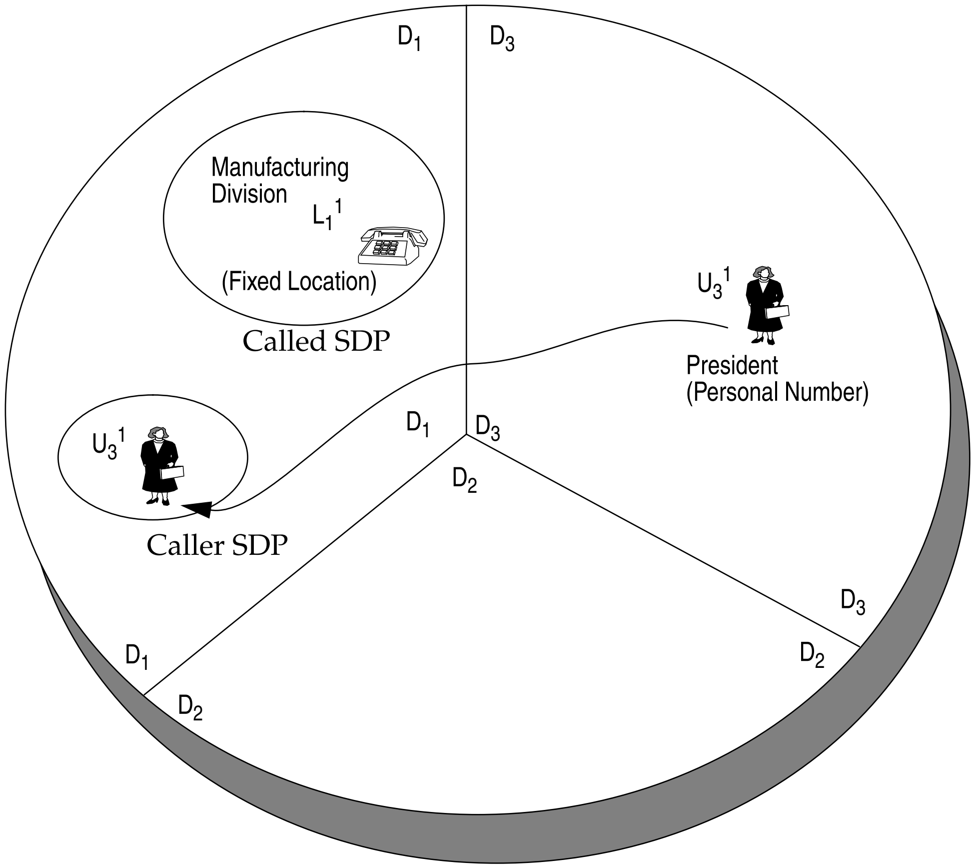


Figure 6.24 Call Setup

3. The SLP retrieves the service profile for the CallerSDPID U_3^1. GetServiceProfile(CallerSDPID). This service profile identifies that the 18 corresponds to SDPID L_1^1.

4. The SLP retrieves the service profile of L_1^1. GetServiceProfile(CalledSDP).

5. Service profiles are checked for previously active service that may result in modification of the call or other restrictions. In the absence of these, a call is setup.

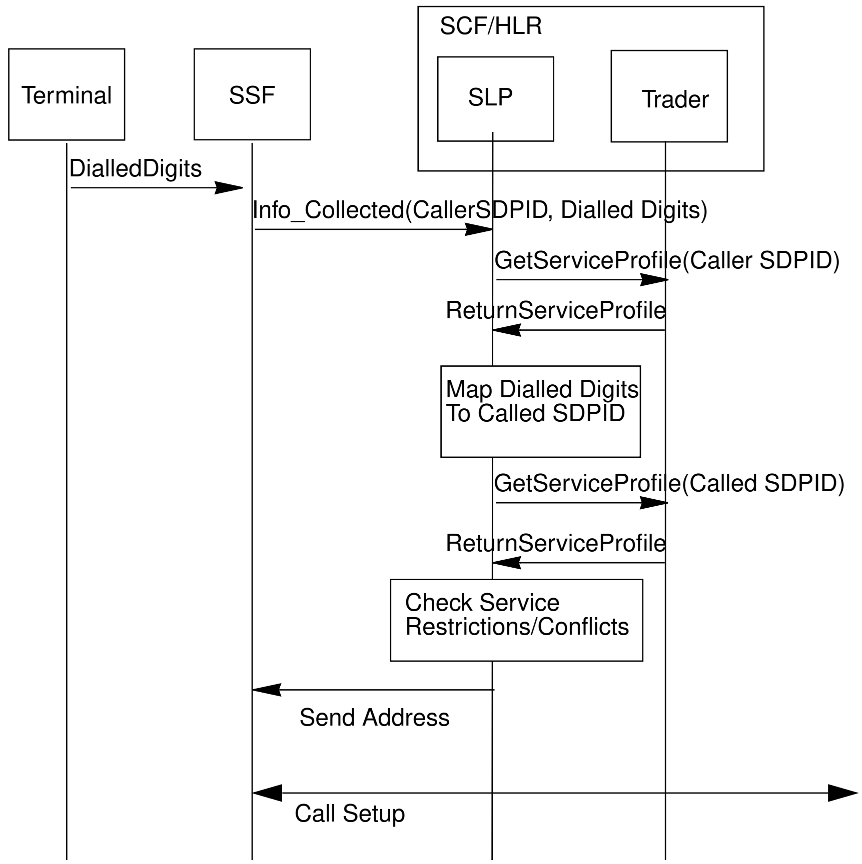


Figure 6.25 Event Diagrams: Call Setup. Caller and Callee in same domain

The second scenario is illustrated in Figure 6.26 and Figure 6.27. The user U_3^1 in $Domain_3$ wishes to establish a call with U_1^1 in $Domain_1$.

1. U_3^1 dials 14 the logical address of U_1^1 according to the Customer Defined Numbering Plan (CDNP). (The CDNP is shown in Figure 6.4)
2. The SSF sends a message InfoCollected(CallerSDPID, DialledDigits) to the SLP.

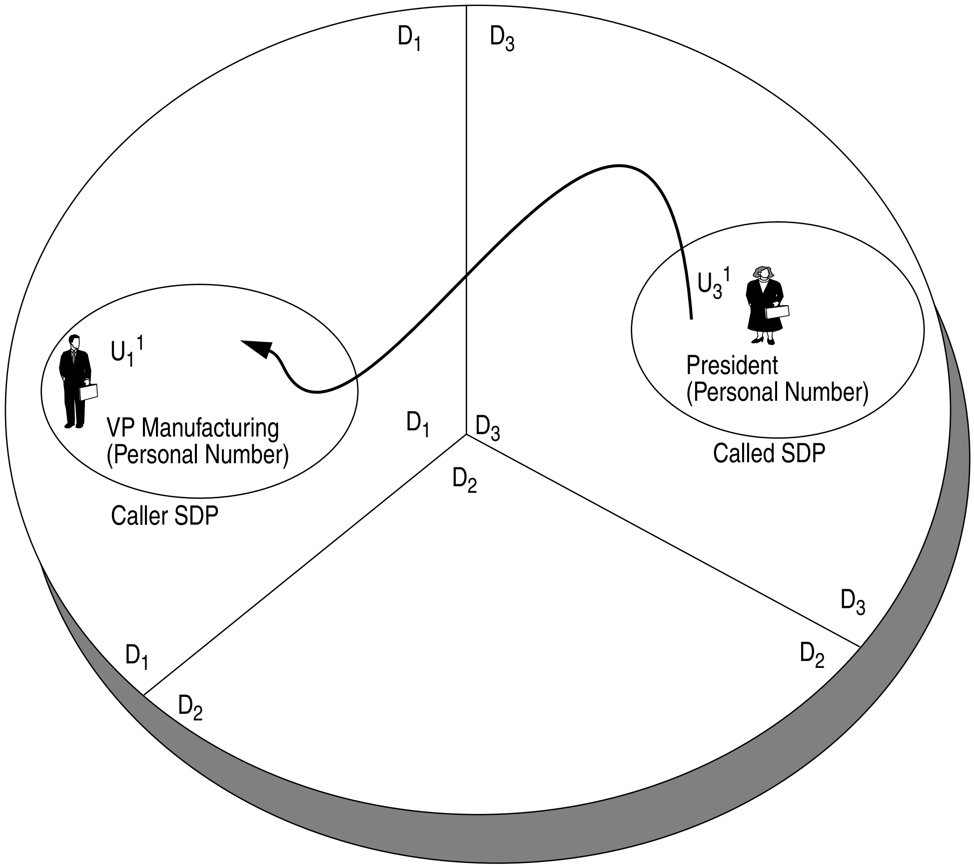


Figure 6.26 Call Setup

3. The SLP retrieves the service profile for the CallerSDPID U_3^1. GetServiceProfile(CallerSDPID). This service profile identifies that the 14 corresponds to SPDID U_1^1.

4. The SLP retrieves the service profile of U_1^1. GetServiceProfile(CalledSDP).

The service profile is not available. The trader interworks along a link in the trading graph to the called SDPs home domain.(Import(GetServiceProfile(CalledSDPID))

5. The remote trader returns the service profile address: ReturnServiceProfile(Address). In this case ISPT is not performed..

6. Service profiles are checked for previously active service that may result in modification of the call or other restrictions. In the absence of these, a call is setup.

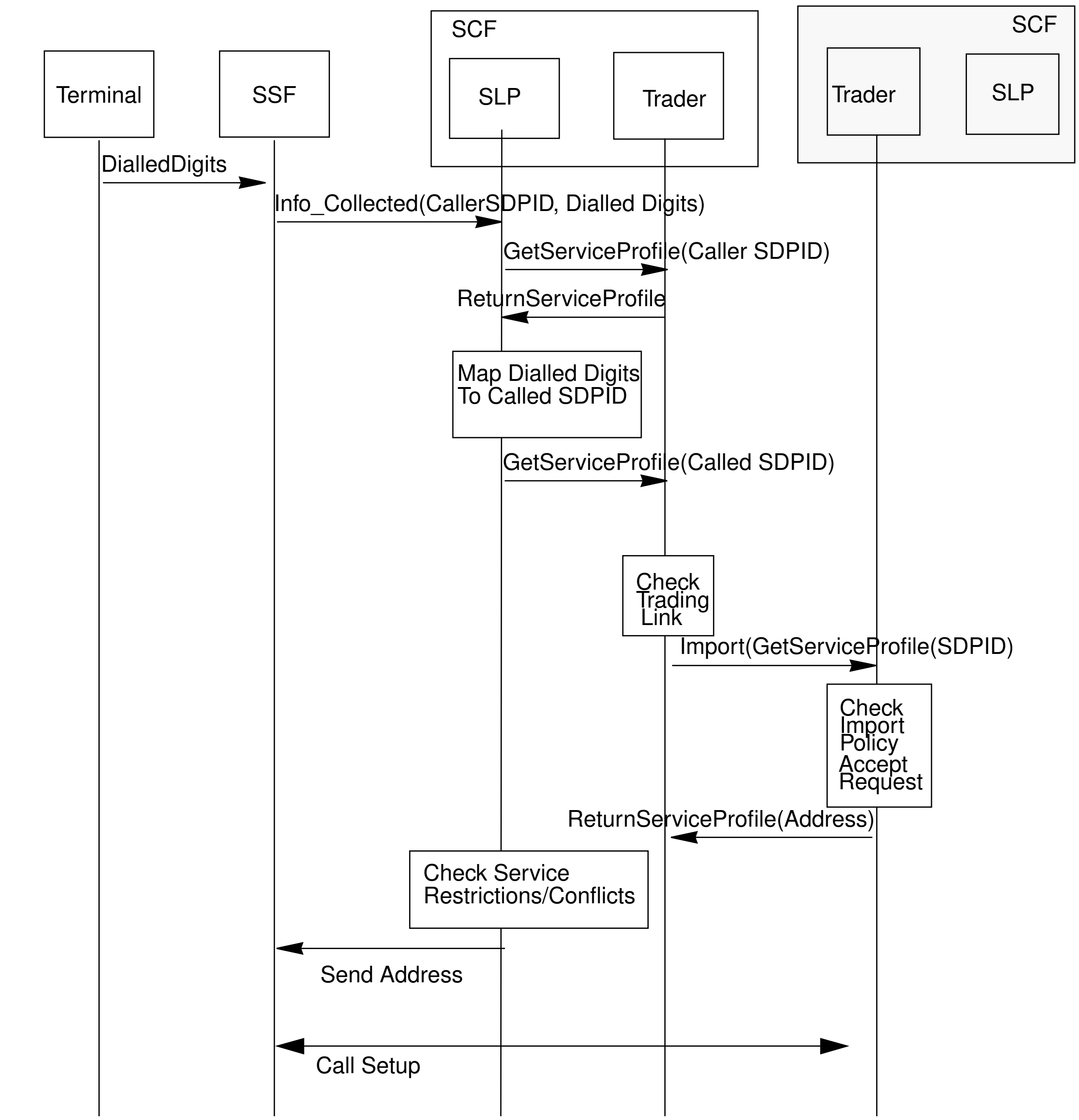


Figure 6.27 Event Diagrams: Call Setup. Caller and Callee in different domains.

Auxiliary Service Access

Auxiliary service access is explained by examination of the following scenario. A

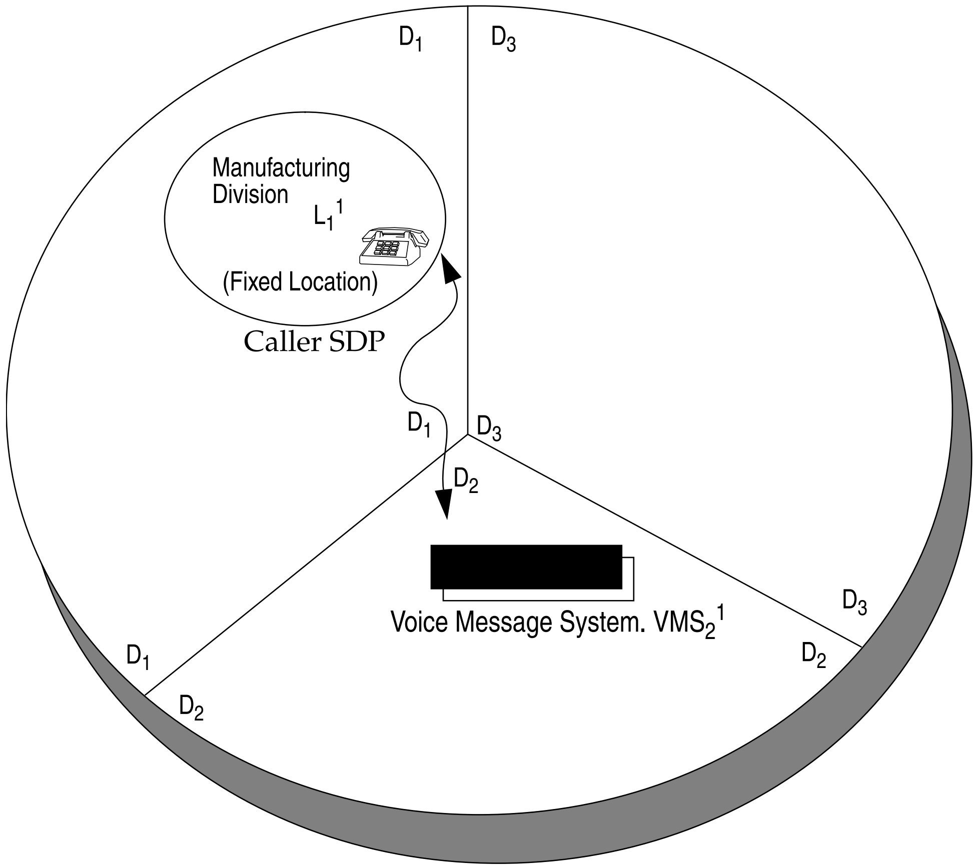


Figure 6.28 Auxiliary Service Access

user utilizes the fixed location L_1^1 to access the voice message system located in

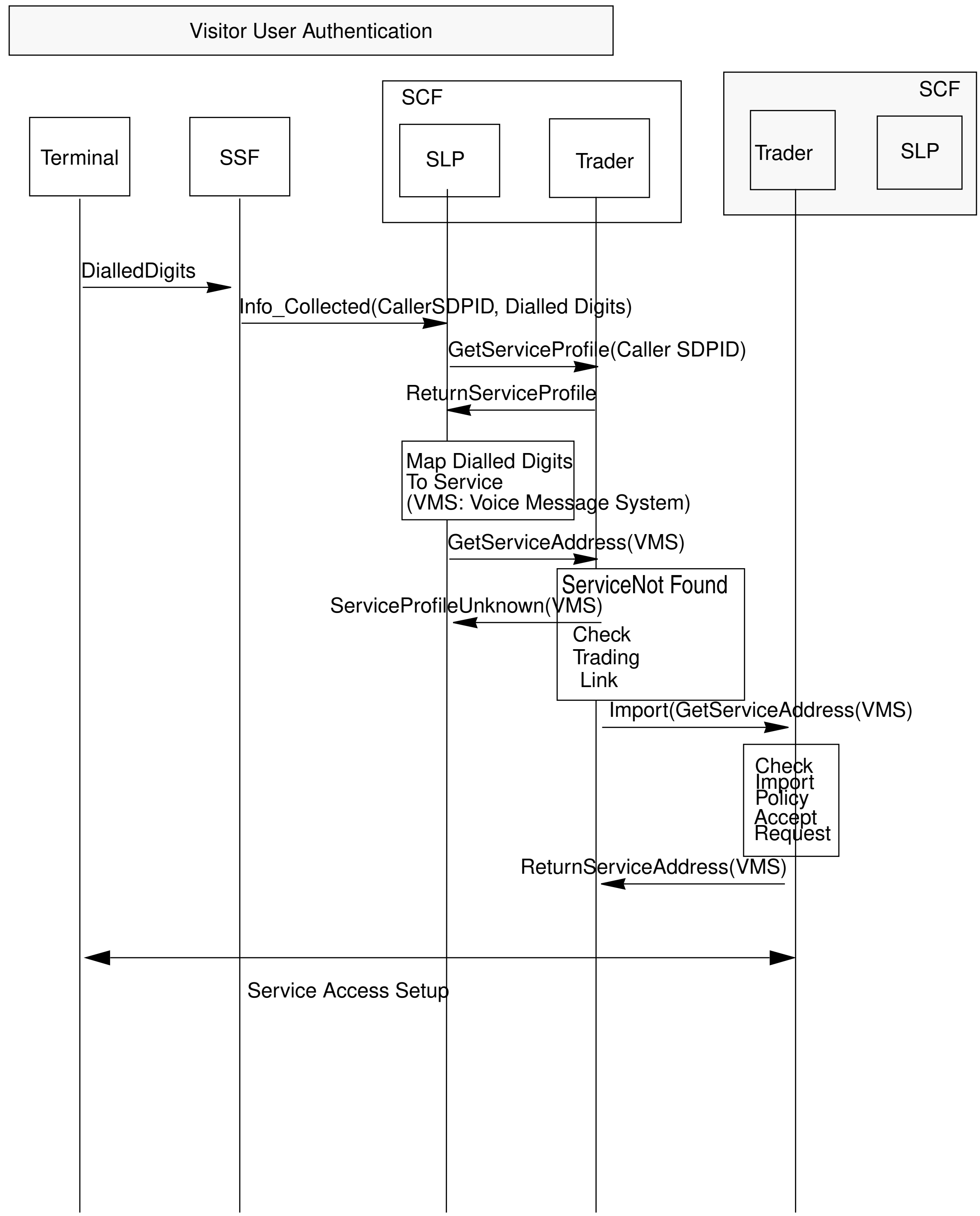


Figure 6.29 Auxiliary Service Access: Event Diagram

domain$_2$.The voice message system is given the number 20.

1. The dialled digits 20 are received by the SSF.

2. The SSF sends a message Info_Collected(CallerSDPID, Dialled Digits) to the SLP in the SCF.

3. The SLP retrieves the service profile of the caller SDP (i.e. of L_1^1). GetSerVicePro-file(CallerSDPID). This service profile identifies that 18 corresponds to SDPID VMS_2^1.

4. The SLP attempts to retrieve the address of the VMS.

5. The address is not found in the local domain, and the trader interworks along its trading links. It imports the service address using the Import(GetServiceAddress(VMS) message.

6.The remote trader checks the import request policy and returns the Service Address. ReturnServiceAddress(VMS).

7. The SSF connects the caller SDP with the Voice Message System.

6.2 Conclusion

This chapter showed the enterprise and information viewpoints of a GVNS using a case study. It showed that functionalities described in RM-ODP, i.e. service trading can be effectively applied to telecommunications service modeling. The trading graph allows a service designer to focus on interworking aspects. The Interworking State Model provides a uniform view of interworking activities. Such a model allows the SCFs to have vendor implementation interactions with each other.

The next chapter, chapter 7, discusses the conclusions of this work and points to potential future work.

Chapter 7
Conclusions and Further Work

7.1 Introduction

The primary intent of this work was to apply ODP distributed processing concepts for the development of IN services in a mobile and distributed environment.This chapter concludes by highlighting significant contributions and pointing to possible further research work.

This thesis presented an integrated modeling to enhance IN service development using RM-ODP concepts. The context of this work was explained by investigation of telecommunication service trends and the limitations of existing IN methodology (Chapter 2). Subsequent to elucidation of RM-ODP concepts in chapter 3, a service representation mechanism to model personal and terminal mobility in a distributed interworked-domain environment was proposed in chapter 4. Chapter 5 explored IN and ODP abstraction levels, and discussed their integration. Chapter 6 presented a case study of an integrated modeling of a mobile and distributed service - a GVNS.

7.2 Contributions

The primary contributions of this work are discussed in the sections below.

7.2.1 Service Behavior Abstractions for a Mobile Environment.

Service behavior has traditionally been understood in the fixed network context. When these concepts are extended to the mobility context ambiguities may occour. Such ambiguities were explained in Section 2.5 “Impact of Co-operative and Mobile Services on IN” on page 31. Section 4.2 “RM-ODP Models for Service Profile and Service Portability” on page 59 proposed service behavior abstractions that recognize and fit two key perspectives: evolving network structure and the human situation in terms of roles.

7.2.2 Service Profile Definition

To clarify interworking relationships in a distributed environment, a service profile definition was put forth in Section 4.2.1 “Specification of Service Profiles” on page 63. The service profile definition accounted for roles, multiple subscriptions, and service sets associated with each subscription. Prior to this proposal, service profiles had numerous implicit meanings. This work, with the service profile definition, clarified what information should be transferred in a service profile. As a consequence service configuration and management have been simplified.

7.2.3 Issues in integrated modeling of ODP and IN.

IN is based on telephony concepts while ODP is based on distributed computing. The real problem faced was that requirements and constraints are not only scarce and imprecise but that they belong to extremely different categories. An ad-hoc approach to using ODP in IN was deliberately avoided. Chapter 5 examined the issues in regard to integration of ODP and IN abstraction (Section 5.1.1 "Modeling Abstraction" on page 68) and co-operation (Section5.1.2 "Modeling Co-operation" on page 72) mechanisms. Related issues of consistency and system design trajectory were also discussed. Though a rigorous and complete model was not developed, useful introductory concepts were established.

7.2.4 ODP Enterprise and Information Model

Based on the integration concepts in chapter 5, the enterprise and information modeling was demonstrated using a GVNS case study. Two significant contributions were:

Use of Trading Graph

The use of ODP trading graphs (Section 6.1.2 "BetterMouseTraps GVNS Trading Graph" on page 95) as representations that allow a service designer to define information structure and valid information flows. This was to supplement a deficiency in IN, i.e., an absence of models that support the exchange of information between any two SCFs. The ODP trading graph was used to represent configuration of domains and the flow of control information across them. The graph isolates cross references between domains and highlights interworking aspects. Shared service logic and data are easily visible. Policies associated with co-operation and administrative constraints are made explicit.

Interworking State Machine.

An Interworking State Machine (ISM) was proposed, which is a representation of the process of interworking two SCFS and the points at which information exchange occurs (Section 6.1.3 "Interworking State Model" on page 98). When standardized by all domains, this ISM would provide a uniform view of distributed services processing

The concepts were illustrated by means of an example.

7.3 Further Work

Many topics in this thesis require further attention. A direction that could be pursued in the short term is development of ODP computational and engineering viewpoints. Refinement of the enterprise and information models could lead to further clarity. In this work for reasons of manageability, discussion was limited to enterprise and information viewpoints only. The computational and engineering models were researched, but the results could not be consolidated in a manageable unit. It is suggested that specification of computational and engineering viewpoints are immediate items for further work.

The computational and engineering aspects will now be explained.

7.3.1 Computational Viewpoint

A computational viewpoint defines the functional decomposition of an ODP system into objects which interact at interfaces. In the computational viewpoint, applications and ODP functions consist of configurations of interacting computational objects.

As shown in chapter 6, in a mobile GVNS service the interworking of ODP traders can be used to help locate objects. Trader domain management functions are used to manage and configure the exchange of information. In the computational viewpoint, interworking contracts are used to implement domain management functions. Interworking contracts may be asymmetric and are between exporting traders and importing traders. Exporting traders allow access to the offer spaces (possibly with some restrictions). Importing traders use an exporting traders trading space.

Contracts use policies. A policy is a prescriptive relationship between one or more objects and some reference behavior. A behavior is a collection of actions and in the computational viewpoint actions are: operations on objects, which are invoked at interfaces, returning some terminations with input and output parameters. Thus behavior can be specified by interactions and restrictions.

Therefore using computation level abstractions, policy specification could be developed in further detail.

7.3.2 Engineering Viewpoint

An engineering specification defines the mechanisms and functions required to support distributed interaction between computational objects in an ODP system.

The engineering model of the GVNS provides a machine independent execution environment for distributed applications. Unlike the enterprise, information and computational models which deal with the semantics of distributed applications, the engineering model is not concerned with the semantics of the distributed application, except to determine its requirements for distribution [KF93]. The engineering viewpoint provides an infrastructure for support of the computational model. The model is concerned with how an application, specified in the computational viewpoint, may be engineered onto the distributed platform. The selection of protocol objects and other support mechanisms tailored to application needs forms an important task.

It is the author's opinion that the computational interactions described above can be engineered using the X.500 directory standard. The underlying models of both X.500 and ODP trader are very similar. Both represent information as type value pairs. In both these type value pairs are collected into objects which can be manipulated as units. This similarity means that much of the complexity of the ODP trader can be avoided is X.500 is used to store the trader information.

Such work has been done in Designing a ODP trader Implementation using X.500 [WB95].

This exercise may prove to be very useful as it has been suggested in IN standards that interworking IN domains should be based on X.500.

References

[AB94] A. N. Brydon. The Leap into Third Generation Mobile Communication Systems. Proceedings of the IEEE Globecom 94.

[AH91] A.Herbert. The Challenge of ODP. ANSA Phase III Technical Report 30 pages. APM.1016.01. From http://www.ansa.co.uk.

[AN95]. K. Asatani and S. Nogami. Standardization of Network Technologies and Services. IEEE Communications Magazine. August 1995.

[BBV 95]A. Beitz, M. Bearman and A. Vogel. Service Location in an Open Distributed Environment. Proceedings of the Second International Workshop on Services in Distributed and Networked Environements, Whistler, Canada. June 1995.

[BBV95] A. Beitz, M. Bearman and A. Vogel. Service Location in an Open Distributed Environment . Proceedings of Second International Workshop on Services in Distributed and Networked Environments. IEEE Computer Society Press, Los Alamitos, CA, (1995) also available as: Technical Report #21, CRC for Distributed Systems Technology, Brisbane, 1995.

[BCMW95] E. Buitenwerf, G. Colombo, H. Mitts and P. Wright. UMTS: Fixed Network Issues and Design Options. IEEE Personal Communications. February 1995.

[BDS95] H. Bowman, J. Derrick and M. Steen.Some Results on Cross Viewpoints Consistency Checking. Proceedings of The International Conference on Open Distributed Processing (ICODP'95) 1995.

[BFP93] L. Barbetta, A. Francese, B. Papini. Interworking Issues in International UPT service. Proceedings of the ICC 93. Geneva. Switzerland. May 1993.

[BH9] H. Bisseling, J de Hartog. Intelligent Networks for Personal Communications. Intelligent Networks 1994.

[BIV95] W. Brookes, J. Indulska, A. Vogel Types and their Management in Open Distributed Systems. The DSTC Symposium. 1995.

[BKMN94]W. van den Broek, E. Kuisch, C. van Maastricht, T. Norp. Impact of UMTS on IN developments. Proceedings of ICIN-94.

[BR93]M. Bearman and K. Raymond, Federating Traders: An ODP Adventure,Proceedings of the IFIP TC6/WG6.4 International Workshop on Open Distributed Processing (ICODP'91), 1991.

[CB95]C. Burger, Cooperation Policies for Traders. Proceedings of the International Conference on Open Distributed Processing, (ICODP'95) 1995.

[CBG95]M. Chapman, H. Berndt and N. Gatti. Software Architecture for the Future Information Market. Computer Communications. Volume 18, No 11. November 1995.

[CBM 94] C.S. Cheung, Mark A. Beach and J. P. McGeehan. Network Planning for Third generaion Mobile Radio Systems. IEEE Communications Magazine. November 1994.

[CFO 93] S. Chen, M. Fujioka and G. O'Reilly. Intelligent Networking for the Gloabl Marketplace. IEEE Communications Magazine. March 1993.

[CH94] F.Cuervo and D. Helmy. Applying the IN Service Cycle Process to UPT.

IEEE Network. January/February 1994.

[DF95] J.S. Dasilva and B. E. Fernandez. The European Research Program for Advanced Mobile Systems. IEEE Personal Communications. February 1995

[DP 95] P. Dupuis. A European View on the Transition Path Toward Advanced Mobile Systems. IEEE Personal Communications. February 1995.

[DRV95] K. Duddy, K. Raymond and A. Vogel, "Trader Down Under: Upside Down and Inside Out" Technical Report #25, CRC for Distributed Systems Technology, Brisbane, 1995.

[ET94] G.P. eleftheriadis and M.E. Theologou. User Profile Identification in Future Mobile Telecommunications Systesm. IEEE Network September/October 1994.

[F.16] ITU Draft Recommendation F.16. Global Virtual Network Service.

[FG95] F. F. Faraci and G. Gatti. ETSI and Information Society - The State of the Art 1995.

[FZ93] G. H. Forman and J. Zahorajan. The Challenges of Mobile Computing Tech Report #93-11-03. From ftp.cs.wasington.edu

[GH 95] M.P. Gervais, T. Hua, A Proposal of an InterWorking Functional Entity 4th IEEE International Workshop on Intelligent Networks (IN'95), Ottawa, Canada, May 1995.

[GHY96] [D. Graham, H Hetz, and J. Young. Evolution of Intelligent Networks: Object Oriented Relationships for Holistic Communications Services. Proceedings of IEEE Intelligent Networks Workshop IN'96.

[GY93] G.Young. Universal Personal telecommunications Capability Set 1 Issues. Proceeding of IEEE Intelligent Networks Workshop. IN’93.

[HC94] Y. Hoffner and B.Crawford. Federation and Interoperability. ANSA Phase III .Architecture Report APM.1514.01.From http://www.ansa.co.uk.

[HM96] S.S. Husain and J.A. Marocchi. Intelligent Network: A Key Platform for PCS Interworking and Interoperability. IEEE Communications Magazine 1996.

[JPD] J.P. Deschrevel. The ANSA Model for Trading and Federation. ANSA Phase III (approved) Architectuere APM.1005.01. From http://www.ansa.co.uk.

[JR95] J. Rapeli UMTS: Targets, System Concept and Standardization in a Global framework. IEEE Personal Communications. February 1995.

[JR95b] J. Rapeli. Standardization for Global Mobile Communication in the 21st Century. European Telecommunications Standards Institute. ETSI - State of the Art. 1995.

[KF96a] K. Farooqui. Architecture for Open Distributed Software Systems. Parallel and Distributed Computing Handbook, Ed. Albert Y. Zomaya. McGraw Hill Publishers. 1996.

[KF96b] K. Farooqui. ODP-based Distribution Platform: Policy Driven Engineering Support for Mobile and Group-Oriented Distributed Computing. Proceedings of the International Conference on Distributed Platforms. ICDP-96. Germany.

[KJ95] M. Kockelmans and E. de Jong. Overview of IN and TMN harmonization. IEEE Communications Magazine. March 1995.

[KO95] : Kristensen and Osterbyte. Conceptual Modeling and Programming Languages.

[KSB94] J.P. Katoen, A. Saidi & I. Baccaro. A UMTS Network Architecture. Proceedings of the 2nd RACE Mobile Telecommunications Workshop. Amsterdam 1994.

[LC89] L. Cardelli. Typeful Programming. Formal Description of Programming Con cepts. E.J. Neuhold, M. Paul EDs. Springer-Verlag, 1991.

[LC97] Luca Cardelli. Section of Mobile Object Systems. Towards the Programmable Internet. Jan Vitek and Christian Tschudin (Eds.) Second International Workshop, MOS'96. Linz, Austria, July 1996.

[LC96] Y. Lin and I. Chlamtac. Heterogenous Personal Communications Services: Integration of PCS Systems. IEEE Communications Magazine. September 1996.

[LR95] M. Laitinen and J. Rantala. Integration of Intelligent Network Services in Future GSM Networks. IEEE Communications Magazine. June 1995.

[MB95]M.Bearman, Trading in Open Distributed Environments, Proceedings of the International Conference on Open Distributed Processing, February 1995.

[MC 94] M. H. Callendar. Future Public Land Mobile Telecommunication Systems. IEEE Personal Communications. Fourth Quarter 1994

[MPG 94a] M.P. Gervais , Cooperation: a Keyword for the Global IN Achievement Proceedings of the 3rd International Conference on Intelligence in Networks (ICIN'94) Bordeaux, France, October 1994.

[MPG 94b] M.P. Gervais, The UPT Service in a Multi-Operator Environment Proceedings of the 2nd IEEE International Workshop on Advanced Telesevices and High-Speed Communication Architectures (IWACA'94), Heidelberg, Allemagne, September 1994 .Lecture Notes in Computer Science 868, Springer Verlag.

[PH95] P. Haywood. The Dawn of the New VPN Era. Data Communication. September 95.

[PM93] C. Popien and B.Meyer. On the Role of an Administrator in a Federation of ODP Traders. Proceedings of Workshop on Parallel and Distributed Processing, Sofia, May 1993.

[PM94] C. Popien and B. Meyer, A Service Request Description Language, Proceedings of FORTE'94, Bern, Switzerland, October, 1994.

[PZ95] P. Zave. Secrets of Call Forwarding: A Specification Case Study. FORTE'95. Montreal, Canada.

[PZ96] P. Zave. Formal Methods are Research, not Development; in An Invitation to Formal Methods. IEEE Computer, April 1996.

[Q.12xx] ITU Intelligent Networks Draft Recommentations.]

[Q.1223. TD 4/11]. Open Issues on the Global Function Plane. IN Draft Recommendations Q.1223.

[Q.85] Recommendation Q.85. Stage 2 Description of Global Virtual Network Service based on Intellignet Network CS-1 (1995) capabilities.

[RB91] T. Rodden and G. Blair. CSCW and Distributed Systems. The Problem of Control. ECSCW'91. Amstredam, September 1991.

[RH95] A. Richman and D. Hoang, Accomplishing Distributed Traders Utilizing the X.500 Directory, due to appear in MICC'95 2nd IEEE Malaysia International Conference on Communications, Novermber 1995.

[RP95] R.Pandya. Emerging Mobile and Personal Communication Systems. IEEE Communications Magazine. June 1995.

[RS94] Raymond Steel. The Evolution of Personal Communications. IEEE Personal Communications Magazine. Vol 1. No 1.

[TE96]][T. Ely. The Service Control Point as a Cross Network Integrator. in IEEE Intelligent Networks Workshop IN'96 5.1.2] .

[TM96] T. Magedanz. Integration and Evolution of Existing Mobile Telecommunications Systems toward UMTS. IEEE Communications Magazine. Sept 1996.

[VBI95] A. Vogel, A. Beitz, R. Ianella. Discovery and Access of Services in Globally Distributed Systems . The DSTC Symposium 1995.

[WB95]A.Waugh and M.Bearman, Designing an ODP Trader Implementation using X.500, Proceedings of the International Conference on Open Distributed Processing, February 1995.

[X.900] ISO/IEC Draft International Standard DIS 10746. Reference Model for Open Distributed Processing.

Index